


# Understanding Legacy Data Structures

This appendix contains information about data structures supported by eStreamer at previous versions of Secure Firewall System products.

If your client uses event stream requests with bits set to request data in older version formats, you can use the information in this appendix to identify the data structures of the data messages you receive.

Note that prior to version 5.0, separate detection engines were assigned IDs. For version 5.0, devices are assigned IDs. Based on the version, data structures reflect this.

**Note** This appendix describes only data structures from version 4.9 or later of the Secure Firewall System. If you require documentation for structures from earlier data structure versions, contact Cisco Customer Support.

See the following sections for more information:


- Legacy Intrusion Data Structures, page B-1
- Legacy Malware Event Data Structures, page B-68
- Legacy Discovery Data Structures, page B-121
- Legacy Connection Data Structures, page B-158
- Legacy File Event Data Structures, page B-290
- Legacy Correlation Event Data Structures, page B-331
- Legacy Host Data Structures, page B-346


# Legacy Intrusion Data Structures


- Intrusion Event (IPv4) Record 5.0.x - 5.1, page B-2
- Intrusion Event (IPv6) Record 5.0.x - 5.1, page B-6
- Intrusion Event Record 5.2.x, page B-12
- Intrusion Event Record 5.3, page B-17
- Intrusion Event Record 5.1.1.x, page B-23
- Intrusion Event Record 5.3.1, page B-29
- Intrusion Event Record 5.4.x, page B-36
- Intrusion Event Record 6.x, page B-44

- Intrusion Event Record 7.0, page B-53
- Intrusion Impact Alert Data, page B-62
- Intrusion Event Extra Data Record, page B-65
- Intrusion Event Extra Data Metadata, page B-66


# Intrusion Event (IPv4) Record 5.0.x - 5.1

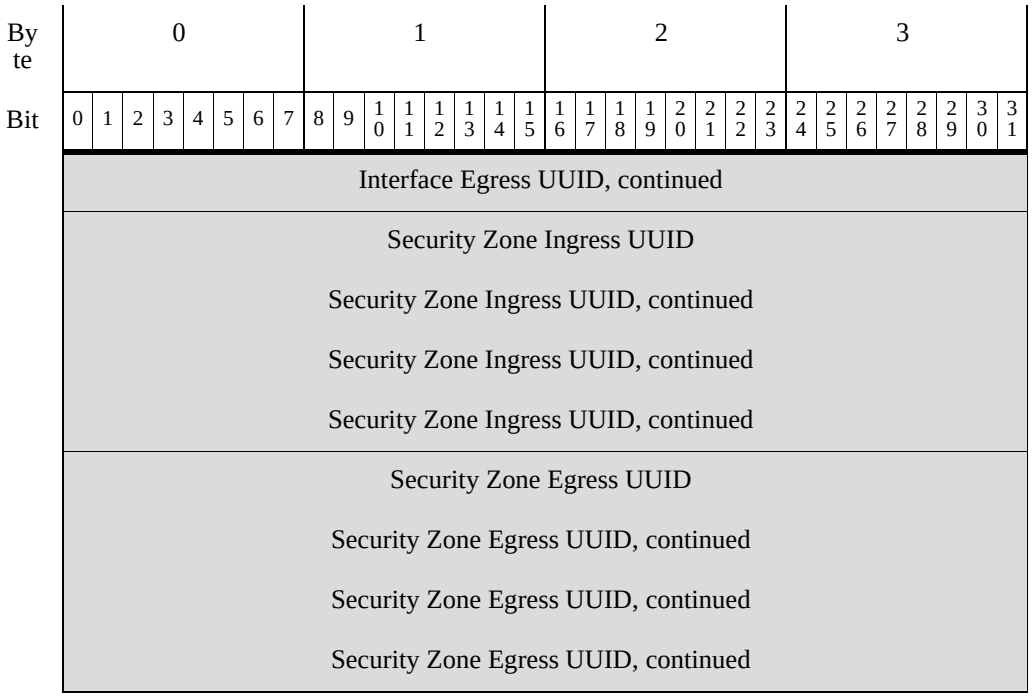
The fields in the intrusion event (IPv4) record are shaded in the following graphic. The record type is 207.

You request intrusion event records by setting the intrusion event flag or the extended requests flag in the request message. See Request Flags, page 2-13 and Submitting Extended Requests, page 2-4.

For version 5.0.x - 5.1 intrusion events, the event ID, the managed device ID, and the event second form a unique identifier.

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Header Version (1) | | | | | | | | | | | | | | | | | Message Type (4) | | | | | | | | | | | | | | | |
| Message Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Netmap ID | | | | | | | | | | | | | | | | | Record Type (207) | | | | | | | | | | | | | | | |
| Record Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| eStreamer Server Timestamp (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reserved for Future Use (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Second | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Microsecond | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID (Signature ID) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Generator ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Classification ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Priority ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IPv4 Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Destination IPv4 Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source Port | | | | | | | | | | | | | | | | | Destination Port | | | | | | | | | | | | | | | |
| IP Protocol ID | | | | | | | | | Impact Flags | | | | | | | | Impact | | | | | | | | Blocked | | | | | | | |
| MPLS Label | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VLAN ID | | | | | | | | | | | | | | | | | Pad | | | | | | | | | | | | | | | |
| Policy UUID<br>Policy UUID, continued<br>Policy UUID, continued<br>Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Web Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Application Protocol ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID<br>Access Control Policy UUID, continued<br>Access Control Policy UUID, continued<br>Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID<br>Interface Ingress UUID, continued<br>Interface Ingress UUID, continued<br>Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID<br>Interface Egress UUID, continued<br>Interface Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |




The following table describes each intrusion event record data field.

***Table B-1 Intrusion Event (IPv4) Record Fields***

| Field | Data Type | Description |
|---|---|---|
| Device ID | unit32 | Contains the identification number of the detecting managed device. You can obtain the managed device name by requesting Version 3 or 4 metadata. See Managed Device Record Metadata, page 3-33 for more information. |
| Event ID | uint32 | Event identification number. |
| Event Second | uint32 | UNIX timestamp (seconds since 01/01/1970) of the event's detection. |
| Event Microsecond | uint32 | Microsecond (one millionth of a second) increment of the timestamp of the event's detection. |
| Rule ID (Signature ID) | uint32 | Rule identification number that corresponds with the event. |
| Generator ID | uint32 | Identification number of the Secure Firewall System preprocessor that generated the event. |
| Rule Revision | uint32 | Rule revision number. |
| Classification ID | uint32 | Identification number of the event classification message. |
| Priority ID | uint32 | Identification number of the priority associated with the event. |
| Source IPv4 Address | uint8[4] | Source IPv4 address used in the event, in address octets. |
| Destination IPv4 Address | uint8[4] | Destination IPv4 address used in the event, in address octets. |

*Table B-1 Intrusion Event (IPv4) Record Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Source Port | uint16 | The source port number if the event protocol type is TCP or UDP. |
| Destination Port | uint16 | The destination port number if the event protocol type is TCP or UDP. |
| IP Protocol Number | uint8 | IANA-specified protocol number. For example:<br>• `0` — IP<br>• `1` — ICMP<br>• `6` — TCP<br>• `17` — UDP |
| Impact Flags | bits[8] | Impact flag value of the event. The low-order eight bits indicate the impact level. Values are:<br>• `0x01` (bit 0) — Source or destination host is in a network monitored by the system.<br>• `0x02` (bit 1) — Source or destination host exists in the network map.<br>• `0x04` (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol.<br>• `0x08` (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event.<br>• `0x10` (bit 4) — There is a vulnerability mapped to the server detected in the event.<br>• `0x20` (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface.<br>• `0x40` (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software.<br>• `0x80` (bit 7) — There is a vulnerability mapped to the client detected in the event.<br>The following impact level values map to specific priorities on the Defense Center. An `X` indicates the value can be `0` or `1`:<br>• (0, unknown): `00X00000`<br>• red (1, vulnerable): `XXXX1XXX, XXX1XXXX, X1XXXXXX, 1XXXXXXX`<br>• orange (2, potentially vulnerable): `00X00111`<br>• yellow (3, currently not vulnerable): `00X00011`<br>• blue (4, unknown target): `00X00001` |

*Table B-1 Intrusion Event (IPv4) Record Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Impact | uint8 | Impact flag value of the event. Values are:<br>• 1 — Red (vulnerable)<br>• 2 — Orange (potentially vulnerable)<br>• 3 — Yellow (currently not vulnerable)<br>• 4 — Blue (unknown target)<br>• 5 — (unknown impact) |
| Blocked | uint8 | Value indicating whether the event was blocked.<br>• 0 — Not blocked<br>• 1 — Blocked<br>• 2 — Would be blocked (but not permitted by configuration) |
| MPLS Label | uint32 | MPLS label. |
| VLAN ID | uint16 | Indicates the ID of the VLAN where the packet originated. |
| Pad | uint16 | Reserved for future use. |
| Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the intrusion policy. |
| User ID | uint32 | The internal identification number for the user, if applicable. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |
| Application Protocol ID | uint32 | The internal identification number for the application protocol, if applicable. |
| Access Control Rule ID | uint32 | A rule ID number that acts as a unique identifier for the access control rule. |
| Access Control Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the access control policy. |
| Ingress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the ingress interface. |
| Egress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the egress interface. |
| Ingress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the ingress security zone. |
| Egress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the egress security zone. |

# Intrusion Event (IPv6) Record 5.0.x - 5.1

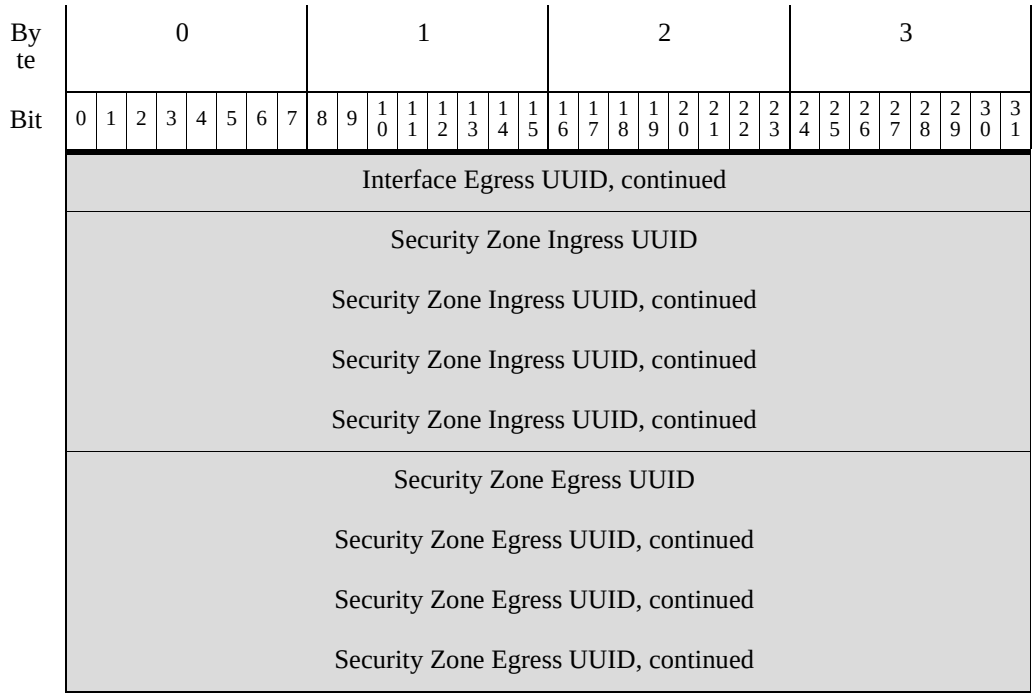
The fields in the intrusion event (IPv6) record are shaded in the following graphic. The record type is 208.

You request intrusion event records by setting the intrusion event flag or the extended requests flag in the request message. See Request Flags, page 2-13 and Submitting Extended Requests, page 2-4.

For version 5.0.x - 5.1 intrusion events, the event ID, the managed device ID, and the event second form a unique identifier.

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Header Version (1) | | | | | | | | | | | | | | | | | Message Type (4) | | | | | | | | | | | | | | | |
| Message Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Netmap ID | | | | | | | | | | | | | | | | | Record Type (208) | | | | | | | | | | | | | | | |
| Record Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| eStreamer Server Timestamp (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reserved for Future Use (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Second | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Microsecond | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID (Signature ID) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Generator ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Classification ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Priority ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IPv6 Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IPv6 Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Destination IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source Port/ICMP Type | | | | | | | | | | | | | | | | | Destination Port/ICMP Code | | | | | | | | | | | | | | | |
| IP Protocol ID | | | | | | | | | Impact Flags | | | | | | | | Impact | | | | | | | | Blocked | | | | | | | |
| MPLS Label | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VLAN ID | | | | | | | | | | | | | | | | | Pad | | | | | | | | | | | | | | | |
| Policy UUID<br>Policy UUID, continued<br>Policy UUID, continued<br>Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Web Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Application Protocol ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID<br>Access Control Policy UUID, continued<br>Access Control Policy UUID, continued<br>Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID<br>Interface Ingress UUID, continued<br>Interface Ingress UUID, continued<br>Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID<br>Interface Egress UUID, continued<br>Interface Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |




The following table describes each intrusion event record data field.

***Table B-2 Intrusion Event (IPv6) Record Fields***

| Field | Data Type | Description |
|---|---|---|
| Device ID | unit32 | Contains the identification number of the detecting device. You can obtain the managed device name by requesting Version 3 or 4 metadata. See Managed Device Record Metadata, page 3-33 for more information. |
| Event ID | uint32 | Event identification number. |
| Event Second | uint32 | UNIX timestamp (seconds since 01/01/1970) of the event's detection. |
| Event Microsecond | uint32 | Microsecond (one millionth of a second) increment of the timestamp of the event's detection. |
| Rule ID (Signature ID) | uint32 | Rule identification number that corresponds with the event. |
| Generator ID | uint32 | Identification number of the Secure Firewall System preprocessor that generated the event. |
| Rule Revision | uint32 | Rule revision number. |
| Classification ID | uint32 | Identification number of the event classification message. |
| Priority ID | uint32 | Identification number of the priority associated with the event. |
| Source IPv6 Address | uint8[16] | Source IPv6 address used in the event, in address octets. |
| Destination IPv6 Address | uint8[16] | Destination IPv6 address used in the event, in address octets. |

*Table B-2 Intrusion Event (IPv6) Record Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Source Port/ICMP Type | uint16 | The source port number if the event protocol type is TCP or UDP. If the protocol type is ICMP, this indicates the ICMP type. |
| Destination Port/ICMP Code | uint16 | The destination port number if the event protocol type is TCP or UDP. If the protocol type is ICMP, this indicates the ICMP code. |
| IP Protocol Number | uint8 | IANA-specified protocol number. For example:<br>• `0` — IP<br>• `1` — ICMP<br>• `6` — TCP<br>• `17` — UDP |
| Impact Flags | bits[8] | Impact flag value of the event. The low-order eight bits indicate the impact level. Values are:<br>• `0x01` (bit 0) — Source or destination host is in a network monitored by the system.<br>• `0x02` (bit 1) — Source or destination host exists in the network map.<br>• `0x04` (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol.<br>• `0x08` (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event.<br>• `0x10` (bit 4) — There is a vulnerability mapped to the server detected in the event.<br>• `0x20` (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface.<br>• `0x40` (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software.<br>• `0x80` (bit 7) — There is a vulnerability mapped to the client detected in the event.<br>The following impact level values map to specific priorities on the Defense Center. An `X` indicates the value can be `0` or `1`:<br>• (0, unknown): `00X00000`<br>• red (1, vulnerable): `XXXX1XXX, XXX1XXXX, X1XXXXXX, 1XXXXXXX`<br>• orange (2, potentially vulnerable): `00X00111`<br>• yellow (3, currently not vulnerable): `00X00011`<br>• blue (4, unknown target): `00X00001` |

*Table B-2 Intrusion Event (IPv6) Record Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Impact | uint8 | Impact flag value of the event. Values are:<br>• 1 — Red (vulnerable)<br>• 2 — Orange (potentially vulnerable)<br>• 3 — Yellow (currently not vulnerable)<br>• 4 — Blue (unknown target)<br>• 5 — (unknown impact) |
| Blocked | uint8 | Value indicating whether the event was blocked.<br>• 0 — Not blocked<br>• 1 — Blocked<br>• 2 — Would be blocked (but not permitted by configuration) |
| MPLS Label | uint32 | MPLS label. (Applies to 4.9+ events only.) |
| VLAN ID | uint16 | Indicates the ID of the VLAN where the packet originated. (Applies to 4.9+ events only.) |
| Pad | uint16 | Reserved for future use. |
| Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the intrusion policy. |
| User ID | uint32 | The internal identification number for the user, if applicable. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |
| Application Protocol ID | uint32 | The internal identification number for the application protocol, if applicable. |
| Access Control Rule ID | uint32 | A rule ID number that acts as a unique identifier for the access control rule. |
| Access Control Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the access control policy. |
| Ingress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the ingress interface. |
| Egress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the egress interface. |
| Ingress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the ingress security zone. |
| Egress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the egress security zone. |

# Intrusion Event Record 5.2.x

The fields in the intrusion event record are shaded in the following graphic. The record type is 400 and the block type is 34 in the series 2 set of data blocks.

You can request 5.2.x intrusion events from eStreamer only by extended request, for which you request event type code 12 and version code 5 in the Stream Request message (see Submitting Extended Requests, page 2-4 for information about submitting extended requests).

For version 5.2.x intrusion events, the event ID, the managed device ID, and the event second form a unique identifier. The connection second, connection instance, and connection counter together form a unique identifier for the connection event associated with the intrusion event.

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Header Version (1) | | | | | | | | | | | | | | | | | Message Type (4) | | | | | | | | | | | | | | | |
| Message Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Netmap ID | | | | | | | | | | | | | | | | | Record Type (400) | | | | | | | | | | | | | | | |
| Record Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| eStreamer Server Timestamp (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reserved for Future Use (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Type (34) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Second | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Microsecond | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID (Signature ID) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Generator ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Classification ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Priority ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Source IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source Port or ICMP Type | | | | | | | | | | | | | | | | | Destination Port or ICMP Code | | | | | | | | | | | | | | | |
| IP Protocol ID | | | | | | | | | Impact Flags | | | | | | | | Impact | | | | | | | | Blocked | | | | | | | |
| MPLS Label | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VLAN ID | | | | | | | | | | | | | | | | | Pad | | | | | | | | | | | | | | | |
| Policy UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Web Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Application Protocol ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Ingress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Egress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Connection Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Connection Instance ID | | | | | | | | | | | | | | | | | Connection Counter | | | | | | | | | | | | | | | |
| Source Country | | | | | | | | | | | | | | | | | Destination Country | | | | | | | | | | | | | | | |

The following table describes each intrusion event record data field.

***Table B-3 Intrusion Event Record 5.2.x Fields***

| Field | Data Type | Description |
|---|---|---|
| Block Type | unint32 | Initiates an Intrusion Event data block. This value is always `34`. |
| Block Length | unint32 | Total number of bytes in the Intrusion Event data block, including eight bytes for the Intrusion Event block type and length fields, plus the number of bytes of data that follows. |
| Device ID | unit32 | Contains the identification number of the detecting managed device. You can obtain the managed device name by requesting Version 3 or 4 metadata. See Managed Device Record Metadata, page 3-33 for more information. |

*Table B-3 Intrusion Event Record 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Event ID | uint32 | Event identification number. |
| Event Second | uint32 | UNIX timestamp (seconds since 01/01/1970) of the event's detection. |
| Event Microsecond | uint32 | Microsecond (one millionth of a second) increment of the timestamp of the event's detection. |
| Rule ID (Signature ID) | uint32 | Rule identification number that corresponds with the event. |
| Generator ID | uint32 | Identification number of the Secure Firewall System preprocessor that generated the event. |
| Rule Revision | uint32 | Rule revision number. |
| Classification ID | uint32 | Identification number of the event classification message. |
| Priority ID | uint32 | Identification number of the priority associated with the event. |
| Source IP Address | uint8[16] | Source IPv4 or IPv6 address used in the event. |
| Destination IP Address | uint8[16] | Destination IPv4 or IPv6 address used in the event. |
| Source Port or ICMP Type | uint16 | The source port number if the event protocol type is TCP or UDP, or the ICMP type if the event is caused by ICMP traffic. |
| Destination Port or ICMP Code | uint16 | The destination port number if the event protocol type is TCP or UDP, or the ICMP code if the event is caused by ICMP traffic. |
| IP Protocol Number | uint8 | IANA-specified protocol number. For example:<br>• `0` — IP<br>• `1` — ICMP<br>• `6` — TCP<br>• `17` — UDP |

*Table B-3 Intrusion Event Record 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Impact Flags | bits[8] | Impact flag value of the event. The low-order eight bits indicate the impact level. Values are:<br>• `0x01` (bit 0) — Source or destination host is in a network monitored by the system.<br>• `0x02` (bit 1) — Source or destination host exists in the network map.<br>• `0x04` (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol.<br>• `0x08` (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event.<br>• `0x10` (bit 4) — There is a vulnerability mapped to the server detected in the event.<br>• `0x20` (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface.<br>• `0x40` (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software.<br>• `0x80` (bit 7) — There is a vulnerability mapped to the client detected in the event. (version 5.0+ only)<br>The following impact level values map to specific priorities on the Defense Center. An `x` indicates the value can be 0 or 1:<br>• (0, unknown): `00X00000`<br>• red (1, vulnerable): `XXXX1XXX, XXX1XXXX, X1XXXXXX, 1XXXXXXX` (version 5.0+ only)<br>• orange (2, potentially vulnerable): `00X0011X`<br>• yellow (3, currently not vulnerable): `00X0001X`<br>• blue (4, unknown target): `00X00001` |
| Impact | uint8 | Impact flag value of the event. Values are:<br>• `1` — Red (vulnerable)<br>• `2` — Orange (potentially vulnerable)<br>• `3` — Yellow (currently not vulnerable)<br>• `4` — Blue (unknown target)<br>• `5` — (unknown impact) |
| Blocked | uint8 | Value indicating whether the event was blocked.<br>• `0` — Not blocked<br>• `1` — Blocked<br>• `2` — Would be blocked (but not permitted by configuration) |

*Table B-3 Intrusion Event Record 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| MPLS Label | uint32 | MPLS label. |
| VLAN ID | uint16 | Indicates the ID of the VLAN where the packet originated. |
| Pad | uint16 | Reserved for future use. |
| Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the intrusion policy. |
| User ID | uint32 | The internal identification number for the user, if applicable. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |
| Application Protocol ID | uint32 | The internal identification number for the application protocol, if applicable. |
| Access Control Rule ID | uint32 | A rule ID number that acts as a unique identifier for the access control rule. |
| Access Control Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the access control policy. |
| Ingress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the ingress interface. |
| Egress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the egress interface. |
| Ingress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the ingress security zone. |
| Egress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the egress security zone. |
| Connection Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the connection event associated with the intrusion event. |
| Connection Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the connection event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint 16 | Code for the country of the destination host. |

# Intrusion Event Record 5.3

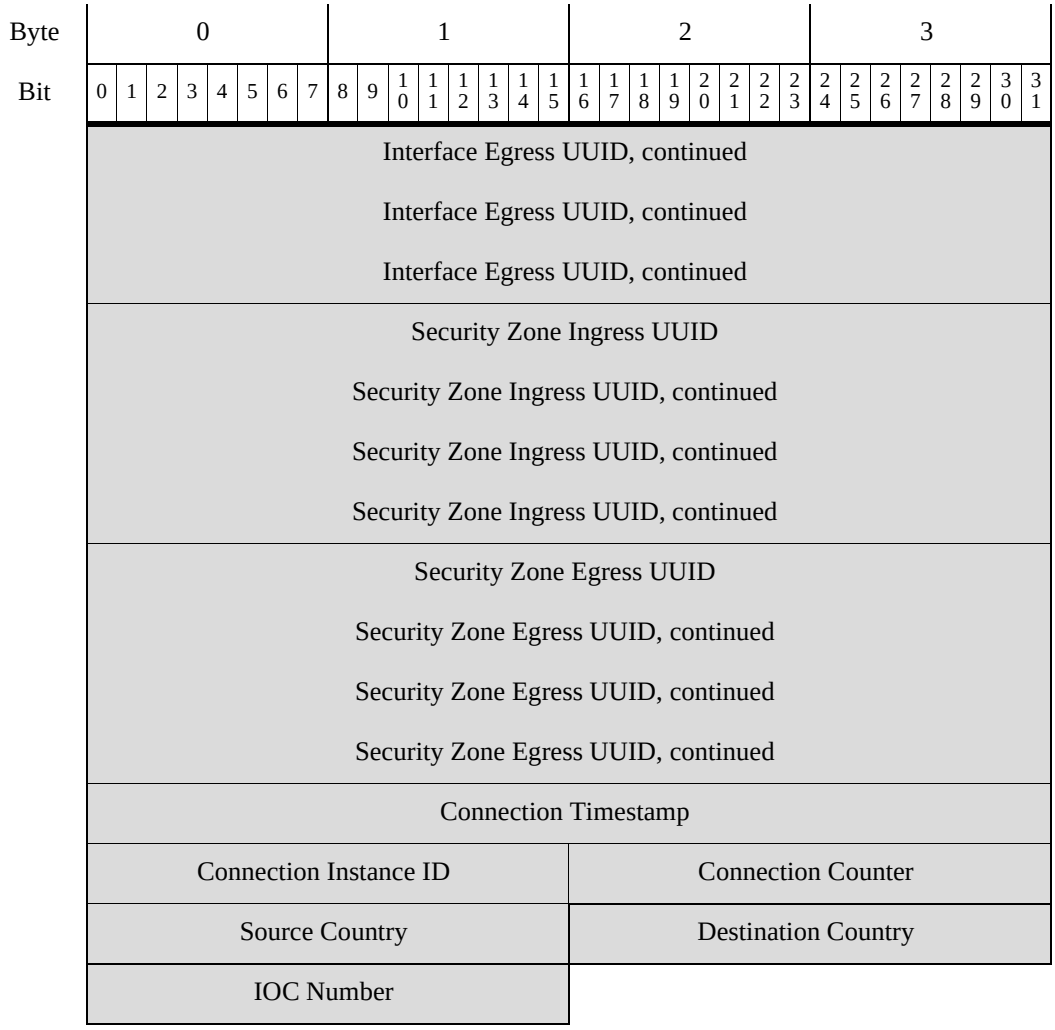
The fields in the intrusion event record are shaded in the following graphic. The record type is 400 and the block type is 41 in the series 2 set of data blocks.

You can request 5.3 intrusion events from eStreamer only by extended request, for which you request event type code 12 and version code 6 in the Stream Request message (see Submitting Extended Requests, page 2-4 for information about submitting extended requests).

For version 5.3 intrusion events, the event ID, the managed device ID, and the event second form a unique identifier. The connection second, connection instance, and connection counter together form a unique identifier for the connection event associated with the intrusion event.

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Header Version (1) | | | | | | | | | | | | | | | | | Message Type (4) | | | | | | | | | | | | | | | |
| Message Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Netmap ID | | | | | | | | | | | | | | | | | Record Type (400) | | | | | | | | | | | | | | | |
| Record Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| eStreamer Server Timestamp (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reserved for Future Use (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Type (41) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Second | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Microsecond | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID (Signature ID) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Generator ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Classification ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Priority ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address<br>Source IP Address, continued<br>Source IP Address, continued<br>Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Destination IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source Port or ICMP Type | | | | | | | | | | | | | | | | | Destination Port or ICMP Code | | | | | | | | | | | | | | | |
| IP Protocol ID | | | | | | | | | Impact Flags | | | | | | | | Impact | | | | | | | | Blocked | | | | | | | |
| MPLS Label | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VLAN ID | | | | | | | | | | | | | | | | | Pad | | | | | | | | | | | | | | | |
| Policy UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Web Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Application Protocol ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |




The following table describes each intrusion event record data field.

***Table B-4 Intrusion Event Record 5.3 Fields***

| Field | Data Type | Description |
|---|---|---|
| Block Type | unint32 | Initiates an Intrusion Event data block. This value is always `34`. |
| Block Length | unint32 | Total number of bytes in the Intrusion Event data block, including eight bytes for the Intrusion Event block type and length fields, plus the number of bytes of data that follows. |
| Device ID | unit32 | Contains the identification number of the detecting managed device. You can obtain the managed device name by requesting Version 3 or 4 metadata. See Managed Device Record Metadata, page 3-33 for more information. |
| Event ID | uint32 | Event identification number. |
| Event Second | uint32 | UNIX timestamp (seconds since 01/01/1970) of the event's detection. |
| Event Microsecond | uint32 | Microsecond (one millionth of a second) increment of the timestamp of the event's detection. |

*Table B-4 Intrusion Event Record 5.3 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Rule ID (Signature ID) | uint32 | Rule identification number that corresponds with the event. |
| Generator ID | uint32 | Identification number of the Secure Firewall System preprocessor that generated the event. |
| Rule Revision | uint32 | Rule revision number. |
| Classification ID | uint32 | Identification number of the event classification message. |
| Priority ID | uint32 | Identification number of the priority associated with the event. |
| Source IP Address | uint8[16] | Source IPv4 or IPv6 address used in the event. |
| Destination IP Address | uint8[16] | Destination IPv4 or IPv6 address used in the event. |
| Source Port or ICMP Type | uint16 | The source port number if the event protocol type is TCP or UDP, or the ICMP type if the event is caused by ICMP traffic. |
| Destination Port or ICMP Code | uint16 | The destination port number if the event protocol type is TCP or UDP, or the ICMP code if the event is caused by ICMP traffic. |
| IP Protocol Number | uint8 | IANA-specified protocol number. For example:<br>• `0` — IP<br>• `1` — ICMP<br>• `6` — TCP<br>• `17` — UDP |

*Table B-4 Intrusion Event Record 5.3 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Impact Flags | bits[8] | Impact flag value of the event. The low-order eight bits indicate the impact level. Values are:<br>• `0x01` (bit 0) — Source or destination host is in a network monitored by the system.<br>• `0x02` (bit 1) — Source or destination host exists in the network map.<br>• `0x04` (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol.<br>• `0x08` (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event.<br>• `0x10` (bit 4) — There is a vulnerability mapped to the server detected in the event.<br>• `0x20` (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface.<br>• `0x40` (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software.<br>• `0x80` (bit 7) — There is a vulnerability mapped to the client detected in the event. (version 5.0+ only)<br>The following impact level values map to specific priorities on the Defense Center. An `X` indicates the value can be `0` or `1`:<br>• (0, unknown): `00X00000`<br>• red (1, vulnerable): `XXXX1XXX, XXX1XXXX, X1XXXXXX, 1XXXXXXX` (version 5.0+ only)<br>• orange (2, potentially vulnerable): `00X0011X`<br>• yellow (3, currently not vulnerable): `00X0001X`<br>• blue (4, unknown target): `00X00001` |
| Impact | uint8 | Impact flag value of the event. Values are:<br>• `1` — Red (vulnerable)<br>• `2` — Orange (potentially vulnerable)<br>• `3` — Yellow (currently not vulnerable)<br>• `4` — Blue (unknown target)<br>• `5` — (unknown impact) |
| Blocked | uint8 | Value indicating whether the event was blocked.<br>• `0` — Not blocked<br>• `1` — Blocked<br>• `2` — Would be blocked (but not permitted by configuration) |

*Table B-4 Intrusion Event Record 5.3 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| MPLS Label | uint32 | MPLS label. |
| VLAN ID | uint16 | Indicates the ID of the VLAN where the packet originated. |
| Pad | uint16 | Reserved for future use. |
| Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the intrusion policy. |
| User ID | uint32 | The internal identification number for the user, if applicable. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |
| Application Protocol ID | uint32 | The internal identification number for the application protocol, if applicable. |
| Access Control Rule ID | uint32 | A rule ID number that acts as a unique identifier for the access control rule. |
| Access Control Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the access control policy. |
| Ingress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the ingress interface. |
| Egress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the egress interface. |
| Ingress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the ingress security zone. |
| Egress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the egress security zone. |
| Connection Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the connection event associated with the intrusion event. |
| Connection Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the connection event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint 16 | Code for the country of the destination host. |
| IOC Number | uint16 | ID Number of the compromise associated with this event. |

# Intrusion Event Record 5.1.1.x

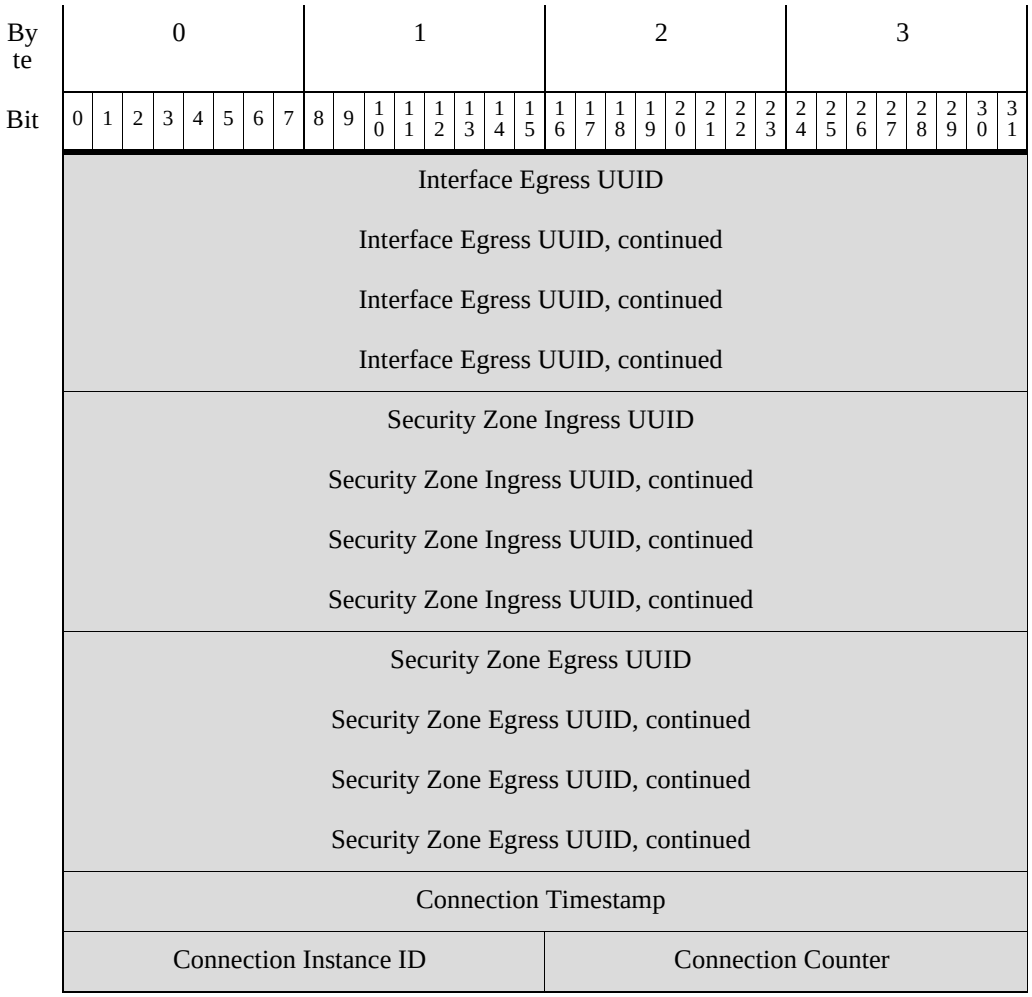
The fields in the intrusion event record are shaded in the following graphic. The record type is 400 and the block type is 25.

You can request 5.1.1.x intrusion events from eStreamer only by extended request, for which you request event type code 12 and version code 4 in the Stream Request message (see Submitting Extended Requests, page 2-4 for information about submitting extended requests).

For version 5.1.1.x intrusion events, the event ID, the managed device ID, and the event second form a unique identifier. The connection second, connection instance, and connection counter together form a unique identifier for the connection event associated with the intrusion event.

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Header Version (1) | | | | | | | | | | | | | | | | | Message Type (4) | | | | | | | | | | | | | | | |
| Message Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Netmap ID | | | | | | | | | | | | | | | | | Record Type (400) | | | | | | | | | | | | | | | |
| Record Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| eStreamer Server Timestamp (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reserved for Future Use (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Type (25) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Second | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Microsecond | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID (Signature ID) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Generator ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Classification ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Priority ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address<br>Source IP Address, continued<br>Source IP Address, continued<br>Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Destination IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source Port/ICMP Type | | | | | | | | | | | | | | | | | Destination Port/ICMP Code | | | | | | | | | | | | | | | |
| IP Protocol ID | | | | | | | | | Impact Flags | | | | | | | | Impact | | | | | | | | Blocked | | | | | | | |
| MPLS Label | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VLAN ID | | | | | | | | | | | | | | | | | Pad | | | | | | | | | | | | | | | |
| Policy UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Web Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Application Protocol ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |




The following table describes each intrusion event record data field.

***Table B-5 Intrusion Event Record 5.1.1 Fields***

| Field | Data Type | Description |
|---|---|---|
| Block Type | unint32 | Initiates an Intrusion Event data block. This value is always `25`. |
| Block Length | unint32 | Total number of bytes in the Intrusion Event data block, including eight bytes for the Intrusion Event block type and length fields, plus the number of bytes of data that follows. |
| Device ID | unit32 | Contains the identification number of the detecting managed device. You can obtain the managed device name by requesting Version 3 or 4 metadata. See Managed Device Record Metadata, page 3-33 for more information. |
| Event ID | uint32 | Event identification number. |
| Event Second | uint32 | UNIX timestamp (seconds since 01/01/1970) of the event's detection. |
| Event Microsecond | uint32 | Microsecond (one millionth of a second) increment of the timestamp of the event's detection. |

*Table B-5 Intrusion Event Record 5.1.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Rule ID (Signature ID) | uint32 | Rule identification number that corresponds with the event. |
| Generator ID | uint32 | Identification number of the Secure Firewall System preprocessor that generated the event. |
| Rule Revision | uint32 | Rule revision number. |
| Classification ID | uint32 | Identification number of the event classification message. |
| Priority ID | uint32 | Identification number of the priority associated with the event. |
| Source IP Address | uint8[16] | Source IPv4 or IPv6 address used in the event. |
| Destination IP Address | uint8[16] | Destination IPv4 or IPv6 address used in the event. |
| Source Port/ICMP Type | uint16 | The source port number if the event protocol type is TCP or UDP, or the ICMP type if the event is caused by ICMP traffic. |
| Destination Port/ICMP Code | uint16 | The destination port number if the event protocol type is TCP or UDP, or the ICMP code if the event is caused by ICMP traffic. |
| IP Protocol Number | uint8 | IANA-specified protocol number. For example:<br>• `0` — IP<br>• `1` — ICMP<br>• `6` — TCP<br>• `17` — UDP |

*Table B-5 Intrusion Event Record 5.1.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Impact Flags | bits[8] | Impact flag value of the event. The low-order eight bits indicate the impact level. Values are:<br>• `0x01` (bit 0) — Source or destination host is in a network monitored by the system.<br>• `0x02` (bit 1) — Source or destination host exists in the network map.<br>• `0x04` (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol.<br>• `0x08` (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event.<br>• `0x10` (bit 4) — There is a vulnerability mapped to the server detected in the event.<br>• `0x20` (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface.<br>• `0x40` (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software.<br>• `0x80` (bit 7) — There is a vulnerability mapped to the client detected in the event.<br>The following impact level values map to specific priorities on the Defense Center. An `x` indicates the value can be `0` or `1`:<br>• (0, unknown): `00X00000`<br>• red (1, vulnerable): `XXXX1XXX, XXX1XXXX, X1XXXXXX, 1XXXXXXX`<br>• orange (2, potentially vulnerable): `00X00111`<br>• yellow (3, currently not vulnerable): `00X00011`<br>• blue (4, unknown target): `00X00001` |
| Impact | uint8 | Impact flag value of the event. Values are:<br>• `1` — Red (vulnerable)<br>• `2` — Orange (potentially vulnerable)<br>• `3` — Yellow (currently not vulnerable)<br>• `4` — Blue (unknown target)<br>• `5` — (unknown impact) |
| Blocked | uint8 | Value indicating whether the event was blocked.<br>• `0` — Not blocked<br>• `1` — Blocked<br>• `2` — Would be blocked (but not permitted by configuration) |

*Table B-5 Intrusion Event Record 5.1.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| MPLS Label | uint32 | MPLS label. |
| VLAN ID | uint16 | Indicates the ID of the VLAN where the packet originated. |
| Pad | uint16 | Reserved for future use. |
| Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the intrusion policy. |
| User ID | uint32 | The internal identification number for the user, if applicable. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |
| Application Protocol ID | uint32 | The internal identification number for the application protocol, if applicable. |
| Access Control Rule ID | uint32 | A rule ID number that acts as a unique identifier for the access control rule. |
| Access Control Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the access control policy. |
| Ingress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the ingress interface. |
| Egress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the egress interface. |
| Ingress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the ingress security zone. |
| Egress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the egress security zone. |
| Connection Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the connection event associated with the intrusion event. |
| Connection Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the connection event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |

# Intrusion Event Record 5.3.1

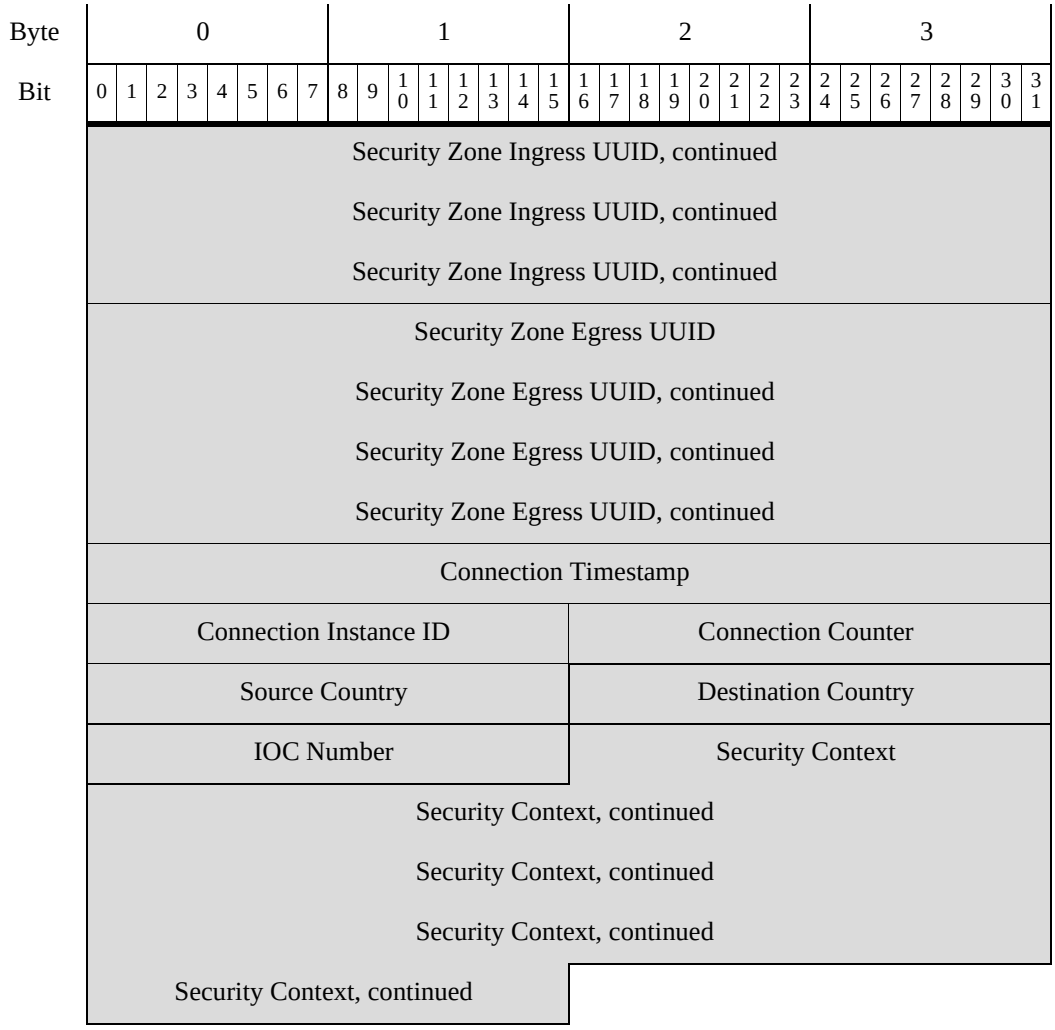
The fields in the intrusion event record are shaded in the following graphic. The record type is 400 and the block type is 42 in the series 2 set of data blocks.

You can request 5.3.1 intrusion events from eStreamer only by extended request, for which you request event type code 12 and version code 7 in the Stream Request message (see Submitting Extended Requests, page 2-4 for information about submitting extended requests).

For version 5.3.1 intrusion events, the event ID, the managed device ID, and the event second form a unique identifier. The connection second, connection instance, and connection counter together form a unique identifier for the connection event associated with the intrusion event.

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Header Version (1) | | | | | | | | | | | | | | | | | Message Type (4) | | | | | | | | | | | | | | | |
| Message Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Netmap ID | | | | | | | | | | | | | | | | | Record Type (400) | | | | | | | | | | | | | | | |
| Record Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| eStreamer Server Timestamp (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reserved for Future Use (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Type (42) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Second | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Microsecond | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID (Signature ID) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Generator ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Classification ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Priority ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address<br>Source IP Address, continued<br>Source IP Address, continued<br>Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address<br>Destination IP Address, continued<br>Destination IP Address, continued<br>Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Source Port or ICMP Type | | | | | | | | | | | | | | | | | Destination Port or ICMP Code | | | | | | | | | | | | | | | |
| IP Protocol ID | | | | | | | | | Impact Flags | | | | | | | | Impact | | | | | | | | Blocked | | | | | | | |
| MPLS Label | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VLAN ID | | | | | | | | | | | | | | | | | Pad | | | | | | | | | | | | | | | |
| Policy UUID<br>Policy UUID, continued<br>Policy UUID, continued<br>Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Web Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Application Protocol ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID<br>Access Control Policy UUID, continued<br>Access Control Policy UUID, continued<br>Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID<br>Interface Ingress UUID, continued<br>Interface Ingress UUID, continued<br>Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID<br>Interface Egress UUID, continued<br>Interface Egress UUID, continued<br>Interface Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Ingress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |




The following table describes each intrusion event record data field.

***Table B-6 Intrusion Event Record 5.3.1 Fields***

| Field | Data Type | Description |
|---|---|---|
| Block Type | unint32 | Initiates an Intrusion Event data block. This value is always 42. |
| Block Length | unint32 | Total number of bytes in the Intrusion Event data block, including eight bytes for the Intrusion Event block type and length fields, plus the number of bytes of data that follows. |
| Device ID | unit32 | Contains the identification number of the detecting managed device. You can obtain the managed device name by requesting Version 3 or 4 metadata. See Managed Device Record Metadata, page 3-33 for more information. |
| Event ID | uint32 | Event identification number. |
| Event Second | uint32 | UNIX timestamp (seconds since 01/01/1970) of the event's detection. |
| Event Microsecond | uint32 | Microsecond (one millionth of a second) increment of the timestamp of the event's detection. |

*Table B-6 Intrusion Event Record 5.3.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Rule ID (Signature ID) | uint32 | Rule identification number that corresponds with the event. |
| Generator ID | uint32 | Identification number of the Secure Firewall System preprocessor that generated the event. |
| Rule Revision | uint32 | Rule revision number. |
| Classification ID | uint32 | Identification number of the event classification message. |
| Priority ID | uint32 | Identification number of the priority associated with the event. |
| Source IP Address | uint8[16] | Source IPv4 or IPv6 address used in the event. |
| Destination IP Address | uint8[16] | Destination IPv4 or IPv6 address used in the event. |
| Source Port or ICMP Type | uint16 | The source port number if the event protocol type is TCP or UDP, or the ICMP type if the event is caused by ICMP traffic. |
| Destination Port or ICMP Code | uint16 | The destination port number if the event protocol type is TCP or UDP, or the ICMP code if the event is caused by ICMP traffic. |
| IP Protocol Number | uint8 | IANA-specified protocol number. For example:<br>• 0 — IP<br>• 1 — ICMP<br>• 6 — TCP<br>• 17 — UDP |

*Table B-6 Intrusion Event Record 5.3.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Impact Flags | bits[8] | Impact flag value of the event. The low-order eight bits indicate the impact level. Values are:<br>• 0x01 (bit 0) — Source or destination host is in a network monitored by the system.<br>• 0x02 (bit 1) — Source or destination host exists in the network map.<br>• 0x04 (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol.<br>• 0x08 (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event.<br>• 0x10 (bit 4) — There is a vulnerability mapped to the server detected in the event.<br>• 0x20 (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface.<br>• 0x40 (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software.<br>• 0x80 (bit 7) — There is a vulnerability mapped to the client detected in the event. (version 5.0+ only)<br>The following impact level values map to specific priorities on the Defense Center. An `x` indicates the value can be 0 or 1:<br>• (0, unknown): `00X00000`<br>• red (1, vulnerable): `XXXX1XXX`, `XXX1XXXX`, `X1XXXXXX`, `1XXXXXXX` (version 5.0+ only)<br>• orange (2, potentially vulnerable): `00X0011X`<br>• yellow (3, currently not vulnerable): `00X0001X`<br>• blue (4, unknown target): `00X00001` |
| Impact | uint8 | Impact flag value of the event. Values are:<br>• `1` — Red (vulnerable)<br>• `2` — Orange (potentially vulnerable)<br>• `3` — Yellow (currently not vulnerable)<br>• `4` — Blue (unknown target)<br>• `5` — (unknown impact) |
| Blocked | uint8 | Value indicating whether the event was blocked.<br>• `0` — Not blocked<br>• `1` — Blocked<br>• `2` — Would be blocked (but not permitted by configuration) |

*Table B-6 Intrusion Event Record 5.3.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| MPLS Label | uint32 | MPLS label. |
| VLAN ID | uint16 | Indicates the ID of the VLAN where the packet originated. |
| Pad | uint16 | Reserved for future use. |
| Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the intrusion policy. |
| User ID | uint32 | The internal identification number for the user, if applicable. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |
| Application Protocol ID | uint32 | The internal identification number for the application protocol, if applicable. |
| Access Control Rule ID | uint32 | A rule ID number that acts as a unique identifier for the access control rule. |
| Access Control Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the access control policy. |
| Ingress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the ingress interface. |
| Egress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the egress interface. |
| Ingress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the ingress security zone. |
| Egress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the egress security zone. |
| Connection Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the connection event associated with the intrusion event. |
| Connection Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the connection event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint 16 | Code for the country of the destination host. |
| IOC Number | uint16 | ID number of the compromise associated with this event. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |

# Intrusion Event Record 5.4.x

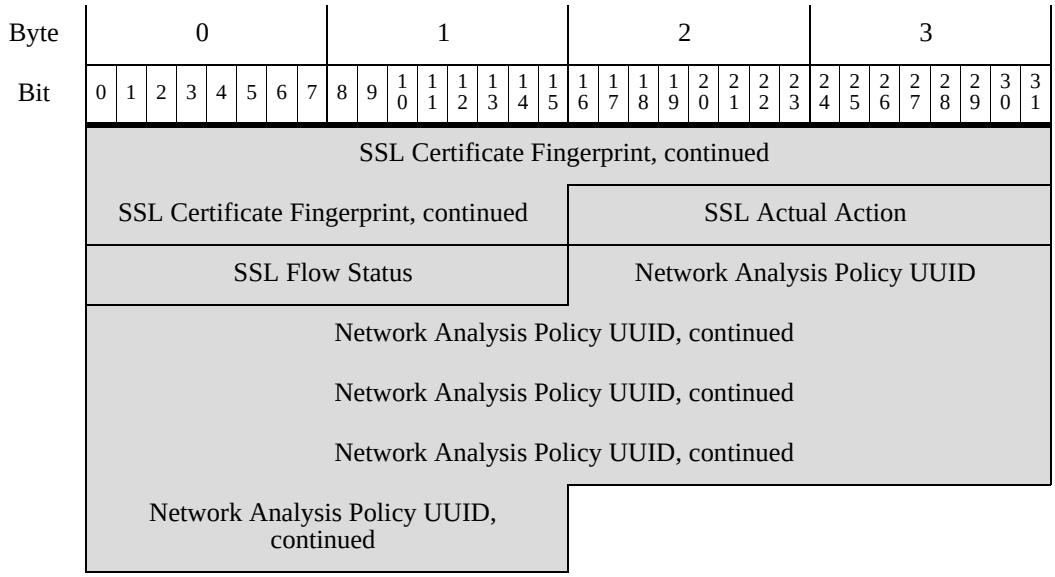
The fields in the intrusion event record are shaded in the following graphic. The record type is 400 and the block type is 45 in the series 2 set of data blocks. It supersedes block type 42, and is superseded by block type 60. Fields for SSL support and Network Analysis Policy have been added.

You can request 5.4.x intrusion events from eStreamer only by extended request, for which you request event type code 12 and version code 8 in the Stream Request message (see Submitting Extended Requests, page 2-4 for information about submitting extended requests).

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Header Version (1) | | | | | | | | | | | | | | | | | Message Type (4) | | | | | | | | | | | | | | | |
| Message Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Netmap ID | | | | | | | | | | | | | | | | | Record Type (400) | | | | | | | | | | | | | | | |
| Record Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| eStreamer Server Timestamp (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reserved for Future Use (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Type (45) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Second | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Microsecond | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID (Signature ID) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Generator ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Classification ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Priority ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Source IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source Port or ICMP Type | | | | | | | | | | | | | | | | | Destination Port or ICMP Code | | | | | | | | | | | | | | | |
| IP Protocol ID | | | | | | | | | Impact Flags | | | | | | | | Impact | | | | | | | | Blocked | | | | | | | |
| MPLS Label | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VLAN ID | | | | | | | | | | | | | | | | | Pad | | | | | | | | | | | | | | | |
| Policy UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Web Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Application Protocol ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Ingress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Egress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Connection Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Connection Instance ID | | | | | | | | | | | | | | | | | Connection Counter | | | | | | | | | | | | | | | |
| Source Country | | | | | | | | | | | | | | | | | Destination Country | | | | | | | | | | | | | | | |
| IOC Number | | | | | | | | | | | | | | | | | Security Context | | | | | | | | | | | | | | | |
| Security Context, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Context, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Context, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Context, continued | | | | | | | | | | | | | | | | | SSL Certificate Fingerprint | | | | | | | | | | | | | | | |
| SSL Certificate Fingerprint, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| SSL Certificate Fingerprint, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| SSL Certificate Fingerprint, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |




The following table describes each intrusion event record data field.

***Table B-7 Intrusion Event Record 5.4.x Fields***

| Field | Data Type | Description |
|---|---|---|
| Block Type | unint32 | Initiates an Intrusion Event data block. This value is always 45. |
| Block Length | unint32 | Total number of bytes in the Intrusion Event data block, including eight bytes for the Intrusion Event block type and length fields, plus the number of bytes of data that follows. |
| Device ID | unit32 | Contains the identification number of the detecting managed device. You can obtain the managed device name by requesting Version 3 or 4 metadata. See Managed Device Record Metadata, page 3-33 for more information. |
| Event ID | uint32 | Event identification number. |
| Event Second | uint32 | UNIX timestamp (seconds since 01/01/1970) of the event's detection. |
| Event Microsecond | uint32 | Microsecond (one millionth of a second) increment of the timestamp of the event's detection. |
| Rule ID (Signature ID) | uint32 | Rule identification number that corresponds with the event. |
| Generator ID | uint32 | Identification number of the Secure Firewall System preprocessor that generated the event. |
| Rule Revision | uint32 | Rule revision number. |
| Classification ID | uint32 | Identification number of the event classification message. |
| Priority ID | uint32 | Identification number of the priority associated with the event. |
| Source IP Address | uint8[16] | Source IPv4 or IPv6 address used in the event. |
| Destination IP Address | uint8[16] | Destination IPv4 or IPv6 address used in the event. |

*Table B-7 Intrusion Event Record 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Source Port or ICMP Type | uint16 | The source port number if the event protocol type is TCP or UDP, or the ICMP type if the event is caused by ICMP traffic. |
| Destination Port or ICMP Code | uint16 | The destination port number if the event protocol type is TCP or UDP, or the ICMP code if the event is caused by ICMP traffic. |
| IP Protocol Number | uint8 | IANA-specified protocol number. For example:<br>• 0 — IP<br>• 1 — ICMP<br>• 6 — TCP<br>• 17 — UDP |
| Impact Flags | bits[8] | Impact flag value of the event. The low-order eight bits indicate the impact level. Values are:<br>• 0x01 (bit 0) — Source or destination host is in a network monitored by the system.<br>• 0x02 (bit 1) — Source or destination host exists in the network map.<br>• 0x04 (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol.<br>• 0x08 (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event.<br>• 0x10 (bit 4) — There is a vulnerability mapped to the server detected in the event.<br>• 0x20 (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface.<br>• 0x40 (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software.<br>• 0x80 (bit 7) — There is a vulnerability mapped to the client detected in the event. (version 5.0+ only)<br>The following impact level values map to specific priorities on the Defense Center. An `x` indicates the value can be 0 or 1:<br>• gray (0, unknown): `00X00000`<br>• red (1, vulnerable): `XXXX1XXX, XXX1XXXX, X1XXXXXX, 1XXXXXXX` (version 5.0+ only)<br>• orange (2, potentially vulnerable): `00X0011X`<br>• yellow (3, currently not vulnerable): `00X0001X`<br>• blue (4, unknown target): `00X00001` |

*Table B-7 Intrusion Event Record 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Impact | uint8 | Impact flag value of the event. Values are:<br>• 1 — Red (vulnerable)<br>• 2 — Orange (potentially vulnerable)<br>• 3 — Yellow (currently not vulnerable)<br>• 4 — Blue (unknown target)<br>• 5 — Gray (unknown impact) |
| Blocked | uint8 | Value indicating whether the event was blocked.<br>• 0 — Not blocked<br>• 1 — Blocked<br>• 2 — Would be blocked (but not permitted by configuration) |
| MPLS Label | uint32 | MPLS label. |
| VLAN ID | uint16 | Indicates the ID of the VLAN where the packet originated. |
| Pad | uint16 | Reserved for future use. |
| Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the intrusion policy. |
| User ID | uint32 | The internal identification number for the user, if applicable. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |
| Application Protocol ID | uint32 | The internal identification number for the application protocol, if applicable. |
| Access Control Rule ID | uint32 | A rule ID number that acts as a unique identifier for the access control rule. |
| Access Control Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the access control policy. |
| Ingress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the ingress interface. |
| Egress Interface UUID | uint8[16] | An interface ID number that acts as a unique identifier for the egress interface. |
| Ingress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the ingress security zone. |
| Egress Security Zone UUID | uint8[16] | A zone ID number that acts as a unique identifier for the egress security zone. |
| Connection Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the connection event associated with the intrusion event. |
| Connection Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the connection event. |

*Table B-7 Intrusion Event Record 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint 16 | Code for the country of the destination host. |
| IOC Number | uint16 | ID number of the compromise associated with this event. |
| Security Context | uint8[16] | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |
| SSL Certificate Fingerprint | uint8[20] | SHA1 hash of the SSL Server certificate. |
| SSL Actual Action | uint16 | The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |

*Table B-7 Intrusion Event Record 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Status | uint16 | Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'No Match'<br>• 2 — 'Success'<br>• 3 — 'Uncached Session'<br>• 4 — 'Unknown Cipher Suite'<br>• 5 — 'Unsupported Cipher Suite'<br>• 6 — 'Unsupported SSL Version'<br>• 7 — 'SSL Compression Used'<br>• 8 — 'Session Undecryptable in Passive Mode'<br>• 9 — 'Handshake Error'<br>• 10 — 'Decryption Error'<br>• 11 — 'Pending Server Name Category Lookup'<br>• 12 — 'Pending Common Name Category Lookup'<br>• 13 — 'Internal Error'<br>• 14 — 'Network Parameters Unavailable'<br>• 15 — 'Invalid Server Certificate Handle'<br>• 16 — 'Server Certificate Fingerprint Unavailable'<br>• 17 — 'Cannot Cache Subject DN'<br>• 18 — 'Cannot Cache Issuer DN'<br>• 19 — 'Unknown SSL Version'<br>• 20 — 'External Certificate List Unavailable'<br>• 21 — 'External Certificate Fingerprint Unavailable'<br>• 22 — 'Internal Certificate List Invalid'<br>• 23 — 'Internal Certificate List Unavailable'<br>• 24 — 'Internal Certificate Unavailable'<br>• 25 — 'Internal Certificate Fingerprint Unavailable'<br>• 26 — 'Server Certificate Validation Unavailable'<br>• 27 — 'Server Certificate Validation Failure'<br>• 28 — 'Invalid Action' |
| Network Analysis Policy UUID | uint8[16] | The UUID of the Network Analysis Policy that created the intrusion event. |

# Intrusion Event Record 6.x

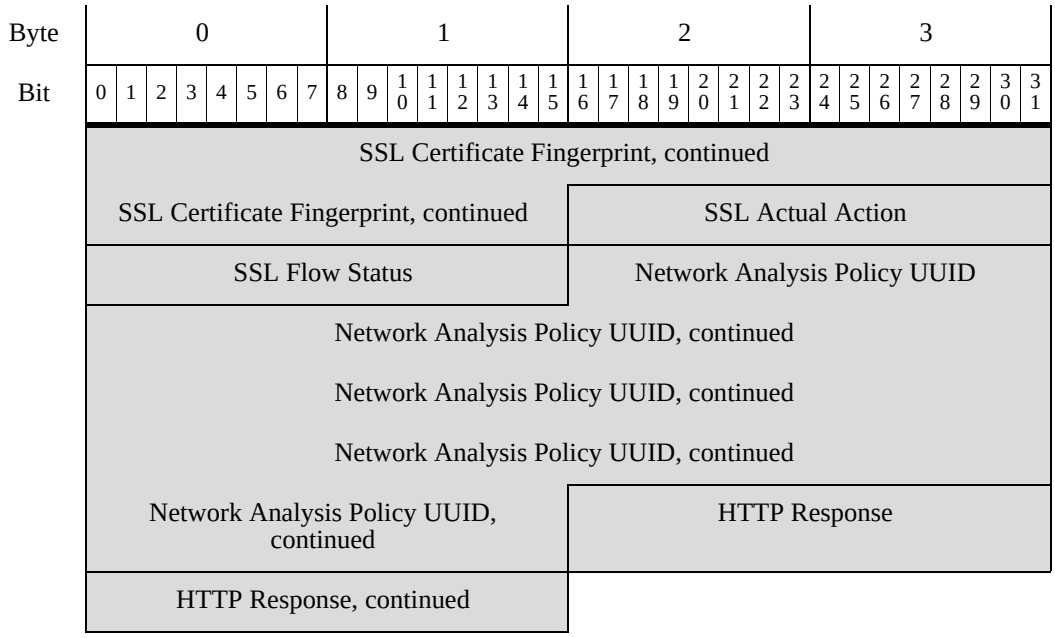
The fields in the intrusion event record are shaded in the following graphic. The record type is 400 and the block type is 60 in the series 2 set of data blocks. It supersedes block type 45, and is superseded by block type 81 in 7.0. An HTTP Response field has been added.

You can request 6.x intrusion events from eStreamer only by extended request, for which you request event type code 12 and version code 9 in the Stream Request message (see Submitting Extended Requests, page 2-4 for information about submitting extended requests).

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Header Version (1) | | | | | | | | | | | | | | | | | Message Type (4) | | | | | | | | | | | | | | | |
| Message Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Netmap ID | | | | | | | | | | | | | | | | | Record Type (400) | | | | | | | | | | | | | | | |
| Record Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| eStreamer Server Timestamp (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reserved for Future Use (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Type (60) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Second | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Microsecond | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID (Signature ID) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Generator ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Classification ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Priority ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Source IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source Port or ICMP Type | | | | | | | | | | | | | | | | | Destination Port or ICMP Code | | | | | | | | | | | | | | | |
| IP Protocol ID | | | | | | | | | Impact Flags | | | | | | | | Impact | | | | | | | | Blocked | | | | | | | |
| MPLS Label | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VLAN ID | | | | | | | | | | | | | | | | | Pad | | | | | | | | | | | | | | | |
| Policy UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Web Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Application Protocol ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Interface Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Ingress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Ingress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Egress UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Zone Egress UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Connection Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Connection Instance ID | | | | | | | | | | | | | | | | | Connection Counter | | | | | | | | | | | | | | | |
| Source Country | | | | | | | | | | | | | | | | | Destination Country | | | | | | | | | | | | | | | |
| IOC Number | | | | | | | | | | | | | | | | | Security Context | | | | | | | | | | | | | | | |
| Security Context, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Context, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Context, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Security Context, continued | | | | | | | | | | | | | | | | | SSL Certificate Fingerprint | | | | | | | | | | | | | | | |
| SSL Certificate Fingerprint, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| SSL Certificate Fingerprint, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| SSL Certificate Fingerprint, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |




The following table describes each intrusion event record data field.

***Table B-8 Intrusion Event Record 6.x Fields***

| Field | Data Type | Description |
|---|---|---|
| Block Type | unint32 | Initiates an Intrusion Event data block. This value is always 60. |
| Block Length | unint32 | Total number of bytes in the Intrusion Event data block, including eight bytes for the Intrusion Event block type and length fields, plus the number of bytes of data that follows. |
| Device ID | unit32 | Contains the identification number of the detecting managed device. You can obtain the managed device name by requesting Version 3 or 4 metadata. See Managed Device Record Metadata, page 3-33 for more information. |
| Event ID | uint32 | Event identification number. |
| Event Second | uint32 | UNIX timestamp (seconds since 01/01/1970) of the event's detection. |
| Event Microsecond | uint32 | Microsecond (one millionth of a second) increment of the timestamp of the event's detection. |
| Rule ID (Signature ID) | uint32 | Rule identification number that corresponds with the event. |
| Generator ID | uint32 | Identification number of the Secure Firewall System preprocessor that generated the event. |
| Rule Revision | uint32 | Rule revision number. |
| Classification ID | uint32 | Identification number of the event classification message. |
| Priority ID | uint32 | Identification number of the priority associated with the event. |
| Source IP Address | uint8[16] | Source IPv4 or IPv6 address used in the event. |

*Table B-8 Intrusion Event Record 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Destination IP Address | uint8[16] | Destination IPv4 or IPv6 address used in the event. |
| Source Port or ICMP Type | uint16 | The source port number if the event protocol type is TCP or UDP, or the ICMP type if the event is caused by ICMP traffic. |
| Destination Port or ICMP Code | uint16 | The destination port number if the event protocol type is TCP or UDP, or the ICMP code if the event is caused by ICMP traffic. |
| IP Protocol ID | uint8 | IANA-specified protocol number. For example:<br>• `0` — IP<br>• `1` — ICMP<br>• `6` — TCP<br>• `17` — UDP |

*Table B-8 Intrusion Event Record 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Impact Flags | bits[8] | Impact flag value of the event. The low-order eight bits indicate the impact level. Values are:<br>• `0x01` (bit 0) — Source or destination host is in a network monitored by the system.<br>• `0x02` (bit 1) — Source or destination host exists in the network map.<br>• `0x04` (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol.<br>• `0x08` (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event.<br>• `0x10` (bit 4) — There is a vulnerability mapped to the server detected in the event.<br>• `0x20` (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface.<br>• `0x40` (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software.<br>• `0x80` (bit 7) — There is a vulnerability mapped to the client detected in the event. (version 5.0+ only)<br>The following impact level values map to specific priorities on the Management Center. An `x` indicates the value can be 0 or 1:<br>• gray (0, unknown): `00X00000`<br>• red (1, vulnerable): `XXXX1XXX`, `XXX1XXXX`, `X1XXXXXX`, `1XXXXXXX` (version 5.0+ only)<br>• orange (2, potentially vulnerable): `00X0011X`<br>• yellow (3, currently not vulnerable): `00X0001X`<br>• blue (4, unknown target): `00X00001` |
| Impact | uint8 | Impact flag value of the event. Values are:<br>• `1` — Red (vulnerable)<br>• `2` — Orange (potentially vulnerable)<br>• `3` — Yellow (currently not vulnerable)<br>• `4` — Blue (unknown target)<br>• `5` — Gray (unknown impact) |
| Blocked | uint8 | Value indicating whether the event was blocked.<br>• `0` — Not blocked<br>• `1` — Blocked<br>• `2` — Would be blocked (but not permitted by configuration) |

*Table B-8 Intrusion Event Record 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| MPLS Label | uint32 | MPLS label. |
| VLAN ID | uint16 | Indicates the ID of the VLAN where the packet originated. |
| Pad | uint16 | Reserved for future use. |
| Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the intrusion policy. |
| User ID | uint32 | The internal identification number for the user, if applicable. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |
| Application Protocol ID | uint32 | The internal identification number for the application protocol, if applicable. |
| Access Control Rule ID | uint32 | A rule ID number that acts as a unique identifier for the access control rule. |
| Access Control Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the access control policy. |
| Interface Ingress UUID | uint8[16] | An interface ID number that acts as a unique identifier for the ingress interface. |
| Interface Egress UUID | uint8[16] | An interface ID number that acts as a unique identifier for the egress interface. |
| Security Zone Ingress UUID | uint8[16] | A zone ID number that acts as a unique identifier for the ingress security zone. |
| Security Zone Egress UUID | uint8[16] | A zone ID number that acts as a unique identifier for the egress security zone. |
| Connection Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the connection event associated with the intrusion event. |
| Connection Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the connection event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint 16 | Code for the country of the destination host. |
| IOC Number | uint16 | ID number of the compromise associated with this event. |
| Security Context | uint8[16] | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |
| SSL Certificate Fingerprint | uint8[20] | SHA1 hash of the SSL Server certificate. |

*Table B-8 Intrusion Event Record 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Actual Action | uint16 | The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |

*Table B-8 Intrusion Event Record 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Status | uint16 | Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'No Match'<br>• 2 — 'Success'<br>• 3 — 'Uncached Session'<br>• 4 — 'Unknown Cipher Suite'<br>• 5 — 'Unsupported Cipher Suite'<br>• 6 — 'Unsupported SSL Version'<br>• 7 — 'SSL Compression Used'<br>• 8 — 'Session Undecryptable in Passive Mode'<br>• 9 — 'Handshake Error'<br>• 10 — 'Decryption Error'<br>• 11 — 'Pending Server Name Category Lookup'<br>• 12 — 'Pending Common Name Category Lookup'<br>• 13 — 'Internal Error'<br>• 14 — 'Network Parameters Unavailable'<br>• 15 — 'Invalid Server Certificate Handle'<br>• 16 — 'Server Certificate Fingerprint Unavailable'<br>• 17 — 'Cannot Cache Subject DN'<br>• 18 — 'Cannot Cache Issuer DN'<br>• 19 — 'Unknown SSL Version'<br>• 20 — 'External Certificate List Unavailable'<br>• 21 — 'External Certificate Fingerprint Unavailable'<br>• 22 — 'Internal Certificate List Invalid'<br>• 23 — 'Internal Certificate List Unavailable'<br>• 24 — 'Internal Certificate Unavailable'<br>• 25 — 'Internal Certificate Fingerprint Unavailable'<br>• 26 — 'Server Certificate Validation Unavailable'<br>• 27 — 'Server Certificate Validation Failure'<br>• 28 — 'Invalid Action' |
| Network Analysis Policy UUID | uint8[16] | The UUID of the Network Analysis Policy that created the intrusion event. |
| HTTP Response | uint32 | Response code of the HTTP Request. |

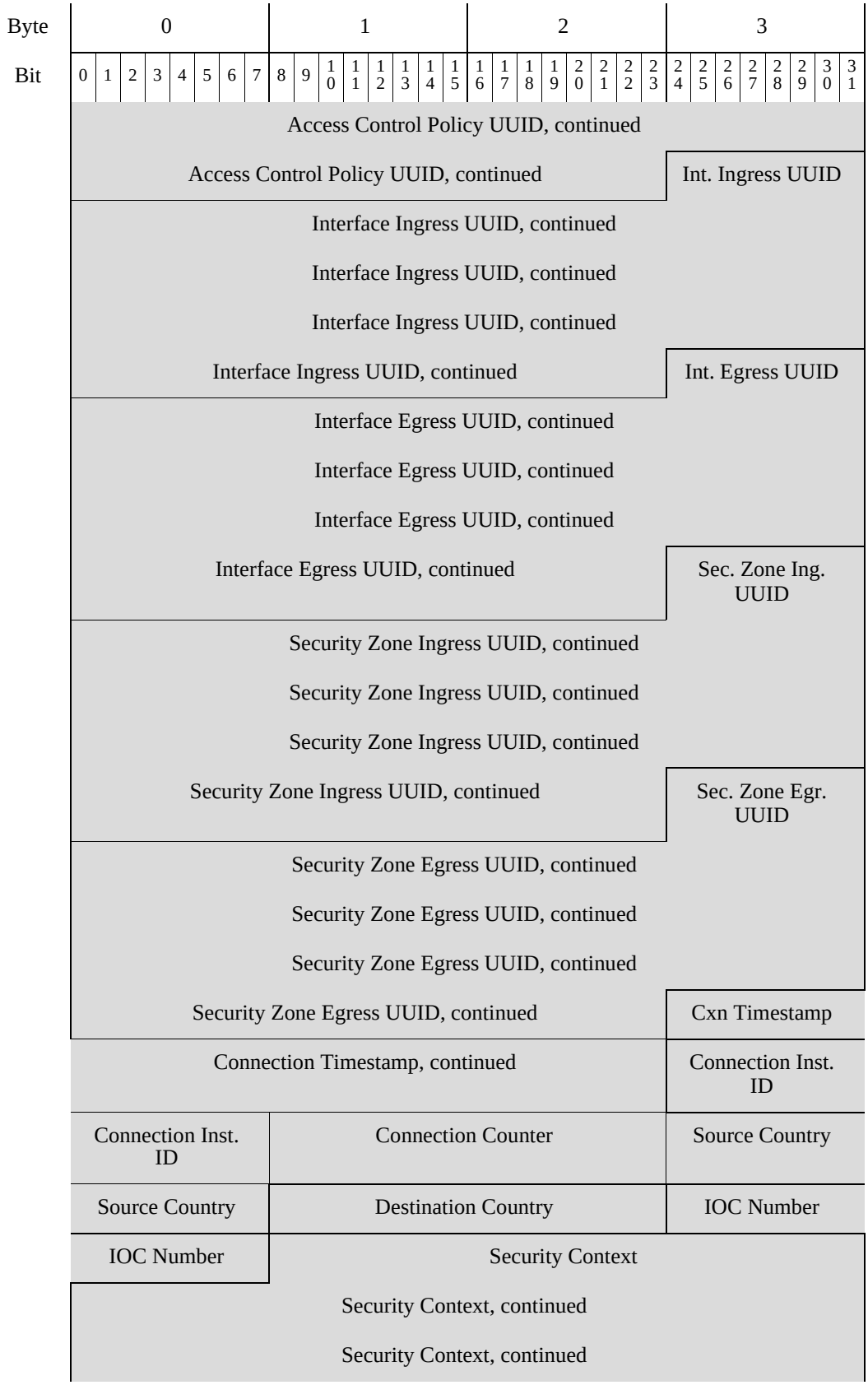
# Intrusion Event Record 7.0

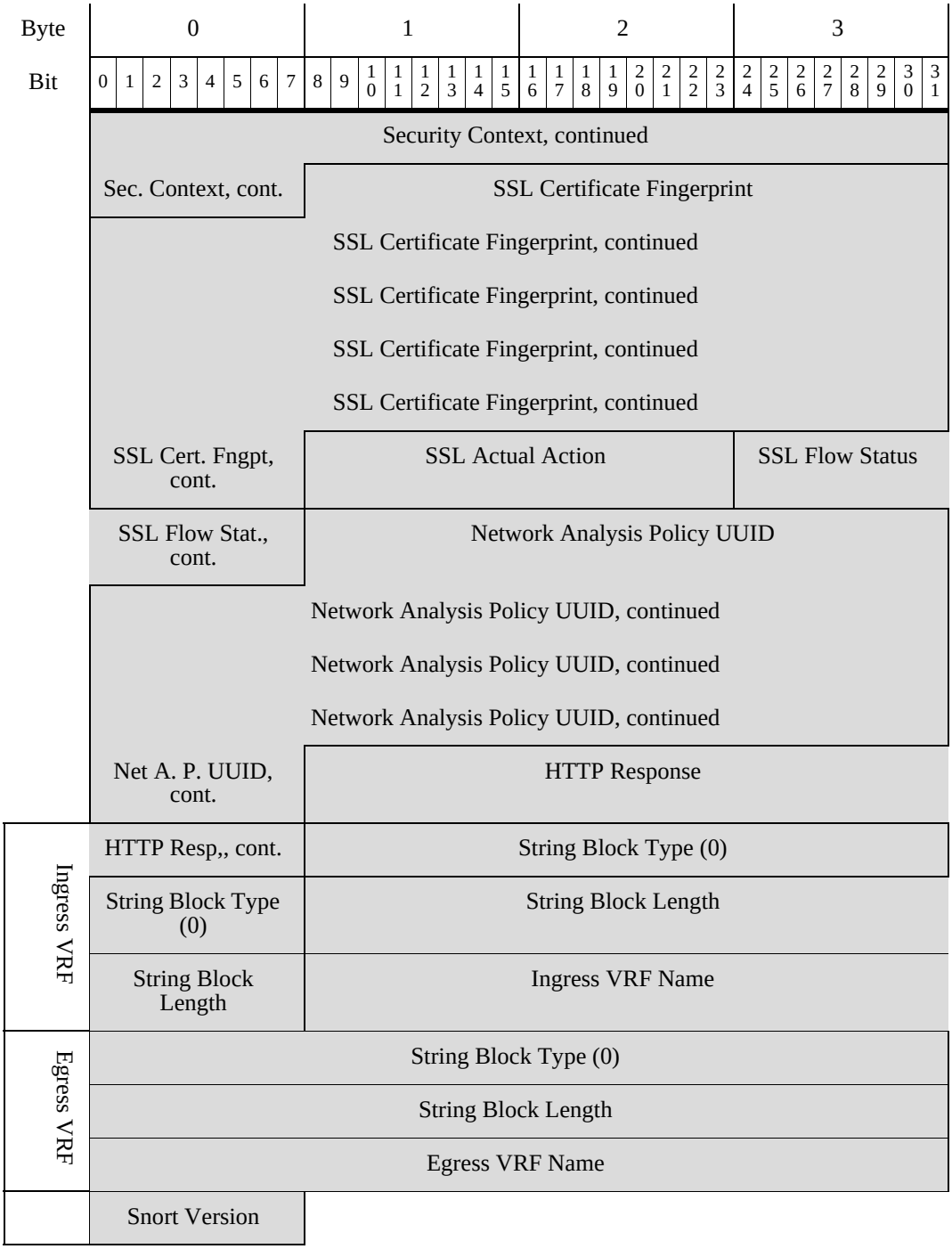
The fields in the intrusion event record are shaded in the following graphic. The record type is 400 and the block type is 81 in the series 2 set of data blocks. It supersedes block type 60, and is superseded by block type 85. Inline Result Reason, Ingress and Egress Virtual Route Forwarding, and Snort Version fields have been added. The Blocked field has been renamed Inline Result.

You can request 7.0 intrusion events from eStreamer only by extended request, for which you request event type code 12 and version code 10in the Stream Request message (see Submitting Extended Requests, page 2-4 for information about submitting extended requests).

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Header Version (1) | | | | | | | | | | | | | | | | | Message Type (4) | | | | | | | | | | | | | | | |
| Message Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Netmap ID | | | | | | | | | | | | | | | | | Record Type (400) | | | | | | | | | | | | | | | |
| Record Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| eStreamer Server Timestamp (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reserved for Future Use (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Type (81) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Second | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Microsecond | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID (Signature ID) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Generator ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Classification ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Priority ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Source IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source Port or ICMP Type | | | | | | | | | | | | | | | | | Destination Port or ICMP Code | | | | | | | | | | | | | | | |
| IP Protocol ID | | | | | | | | | Impact Flags | | | | | | | | Impact | | | | | | | | Inline Result | | | | | | | |
| Inline Result Reason | | | | | | | | | MPLS Label | | | | | | | | | | | | | | | | | | | | | | | |
| MPLS Label, cont. | | | | | | | | | VLAN ID | | | | | | | | | | | | | | | | Pad | | | | | | | |
| Pad, Cont. | | | | | | | | | Policy UUID | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User ID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Web Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Web Application ID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| App. Prot. ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Application Protocol ID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Ctrl Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Rule ID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Acc. Ctrl Policy UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Access Control Policy UUID, continued
Access Control Policy UUID, continued
Int. Ingress UUID
Interface Ingress UUID, continued
Interface Ingress UUID, continued
Interface Ingress UUID, continued
Interface Ingress UUID, continued
Int. Egress UUID
Interface Egress UUID, continued
Interface Egress UUID, continued
Interface Egress UUID, continued
Interface Egress UUID, continued
Sec. Zone Ing. UUID
Security Zone Ingress UUID, continued
Security Zone Ingress UUID, continued
Security Zone Ingress UUID, continued
Security Zone Ingress UUID, continued
Sec. Zone Egr. UUID
Security Zone Egress UUID, continued
Security Zone Egress UUID, continued
Security Zone Egress UUID, continued
Security Zone Egress UUID, continued
Cxn Timestamp
Connection Timestamp, continued
Connection Inst. ID
Connection Inst. ID
Connection Counter
Source Country
Source Country
Destination Country
IOC Number
IOC Number
Security Context
Security Context, continued
Security Context, continued




The following table describes each intrusion event record data field.

*Table B-9 Intrusion Event Record 7.0 Fields*

| Field | Data Type | Description |
|---|---|---|
| Block Type | unint32 | Initiates an Intrusion Event data block. This value is always 81. |
| Block Length | unint32 | Total number of bytes in the Intrusion Event data block, including eight bytes for the Intrusion Event block type and length fields, plus the number of bytes of data that follows. |
| Device ID | unit32 | Contains the identification number of the detecting managed device. You can obtain the managed device name by requesting Version 3 or 4 metadata. See Managed Device Record Metadata, page 3-33 for more information. |
| Event ID | uint32 | Event identification number. |
| Event Second | uint32 | UNIX timestamp (seconds since 01/01/1970) of the event's detection. |
| Event Microsecond | uint32 | Microsecond (one millionth of a second) increment of the timestamp of the event's detection. |
| Rule ID (Signature ID) | uint32 | Rule identification number that corresponds with the event. |
| Generator ID | uint32 | Identification number of the Secure Firewall System preprocessor that generated the event. |
| Rule Revision | uint32 | Rule revision number. |
| Classification ID | uint32 | Identification number of the event classification message. |
| Priority ID | uint32 | Identification number of the priority associated with the event. |
| Source IP Address | uint8[16] | Source IPv4 or IPv6 address used in the event. |
| Destination IP Address | uint8[16] | Destination IPv4 or IPv6 address used in the event. |
| Source Port or ICMP Type | uint16 | The source port number if the event protocol type is TCP or UDP, or the ICMP type if the event is caused by ICMP traffic. |
| Destination Port or ICMP Code | uint16 | The destination port number if the event protocol type is TCP or UDP, or the ICMP code if the event is caused by ICMP traffic. |
| IP Protocol ID | uint8 | IANA-specified protocol number. For example:<br>• `0` — IP<br>• `1` — ICMP<br>• `6` — TCP<br>• `17` — UDP |

*Table B-9 Intrusion Event Record 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Impact Flags | bits[8] | Impact flag value of the event. The low-order eight bits indicate the impact level. Values are:<br>• `0x01` (bit 0) — Source or destination host is in a network monitored by the system.<br>• `0x02` (bit 1) — Source or destination host exists in the network map.<br>• `0x04` (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol.<br>• `0x08` (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event.<br>• `0x10` (bit 4) — There is a vulnerability mapped to the server detected in the event.<br>• `0x20` (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface.<br>• `0x40` (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software.<br>• `0x80` (bit 7) — There is a vulnerability mapped to the client detected in the event. (version 5.0+ only)<br>The following impact level values map to specific priorities on the Management Center. An `x` indicates the value can be 0 or 1:<br>• gray (0, unknown): `00X00000`<br>• red (1, vulnerable): `XXXX1XXX, XXX1XXXX, X1XXXXXX, 1XXXXXXX` (version 5.0+ only)<br>• orange (2, potentially vulnerable): `00X0011X`<br>• yellow (3, currently not vulnerable): `00X0001X`<br>• blue (4, unknown target): `00X00001` |
| Impact | uint8 | Impact flag value of the event. Values are:<br>• `1` — Red (vulnerable)<br>• `2` — Orange (potentially vulnerable)<br>• `3` — Yellow (currently not vulnerable)<br>• `4` — Blue (unknown target)<br>• `5` — Gray (unknown impact) |

*Table B-9 Intrusion Event Record 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Inline Result | uint8 | Value indicating the inline result.<br>• 0 — Pass<br>• 1 — Dropped<br>• 2 — Would be dropped (but not permitted by configuration)<br>• 3— Partially dropped |
| Inline Result Reason | uint8 | Value indicating the inline result reason.<br>• 1— Interface in Passive or Tap mode<br>• 2— Intrusion Policy in "Detection" inspection mode<br>• 3— Network Analysis Policy in "Detection" inspection mode<br>• 4— Connection timed out<br>• 5— Connection Closed (internal use)<br>• 6— Connection Closed (internal use)<br>• 7— Connection Closed (internal use) |
| MPLS Label | uint32 | MPLS label. |
| VLAN ID | uint16 | Indicates the ID of the VLAN where the packet originated. |
| Pad | uint16 | Reserved for future use. |
| Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the intrusion policy. |
| User ID | uint32 | The internal identification number for the user, if applicable. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |
| Application Protocol ID | uint32 | The internal identification number for the application protocol, if applicable. |
| Access Control Rule ID | uint32 | A rule ID number that acts as a unique identifier for the access control rule. |
| Access Control Policy UUID | uint8[16] | A policy ID number that acts as a unique identifier for the access control policy. |
| Interface Ingress UUID | uint8[16] | An interface ID number that acts as a unique identifier for the ingress interface. |
| Interface Egress UUID | uint8[16] | An interface ID number that acts as a unique identifier for the egress interface. |
| Security Zone Ingress UUID | uint8[16] | A zone ID number that acts as a unique identifier for the ingress security zone. |
| Security Zone Egress UUID | uint8[16] | A zone ID number that acts as a unique identifier for the egress security zone. |
| Connection Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the connection event associated with the intrusion event. |

*Table B-9 Intrusion Event Record 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Connection Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the connection event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint 16 | Code for the country of the destination host. |
| IOC Number | uint16 | ID number of the compromise associated with this event. |
| Security Context | uint8[16] | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |
| SSL Certificate Fingerprint | uint8[20] | SHA1 hash of the SSL Server certificate. |
| SSL Actual Action | uint16 | The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |

*Table B-9 Intrusion Event Record 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Status | uint16 | Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'No Match'<br>• 2 — 'Success'<br>• 3 — 'Uncached Session'<br>• 4 — 'Unknown Cipher Suite'<br>• 5 — 'Unsupported Cipher Suite'<br>• 6 — 'Unsupported SSL Version'<br>• 7 — 'SSL Compression Used'<br>• 8 — 'Session Undecryptable in Passive Mode'<br>• 9 — 'Handshake Error'<br>• 10 — 'Decryption Error'<br>• 11 — 'Pending Server Name Category Lookup'<br>• 12 — 'Pending Common Name Category Lookup'<br>• 13 — 'Internal Error'<br>• 14 — 'Network Parameters Unavailable'<br>• 15 — 'Invalid Server Certificate Handle'<br>• 16 — 'Server Certificate Fingerprint Unavailable'<br>• 17 — 'Cannot Cache Subject DN'<br>• 18 — 'Cannot Cache Issuer DN'<br>• 19 — 'Unknown SSL Version'<br>• 20 — 'External Certificate List Unavailable'<br>• 21 — 'External Certificate Fingerprint Unavailable'<br>• 22 — 'Internal Certificate List Invalid'<br>• 23 — 'Internal Certificate List Unavailable'<br>• 24 — 'Internal Certificate Unavailable'<br>• 25 — 'Internal Certificate Fingerprint Unavailable'<br>• 26 — 'Server Certificate Validation Unavailable'<br>• 27 — 'Server Certificate Validation Failure'<br>• 28 — 'Invalid Action' |
| Network Analysis Policy UUID | uint8[16] | The UUID of the Network Analysis Policy that created the intrusion event. |
| HTTP Response | uint32 | Response code of the HTTP Request. |

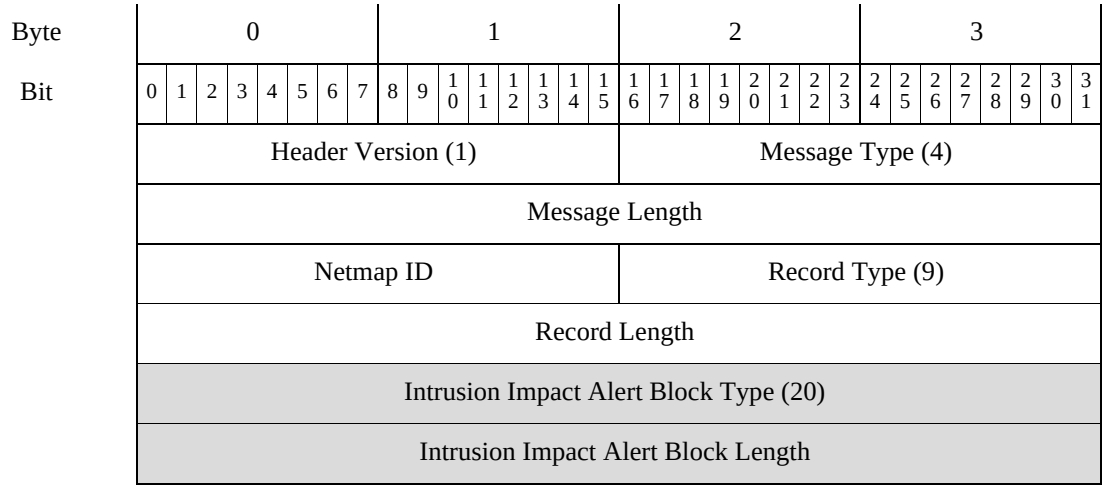
*Table B-9 Intrusion Event Record 7.0 Fields (continued)*

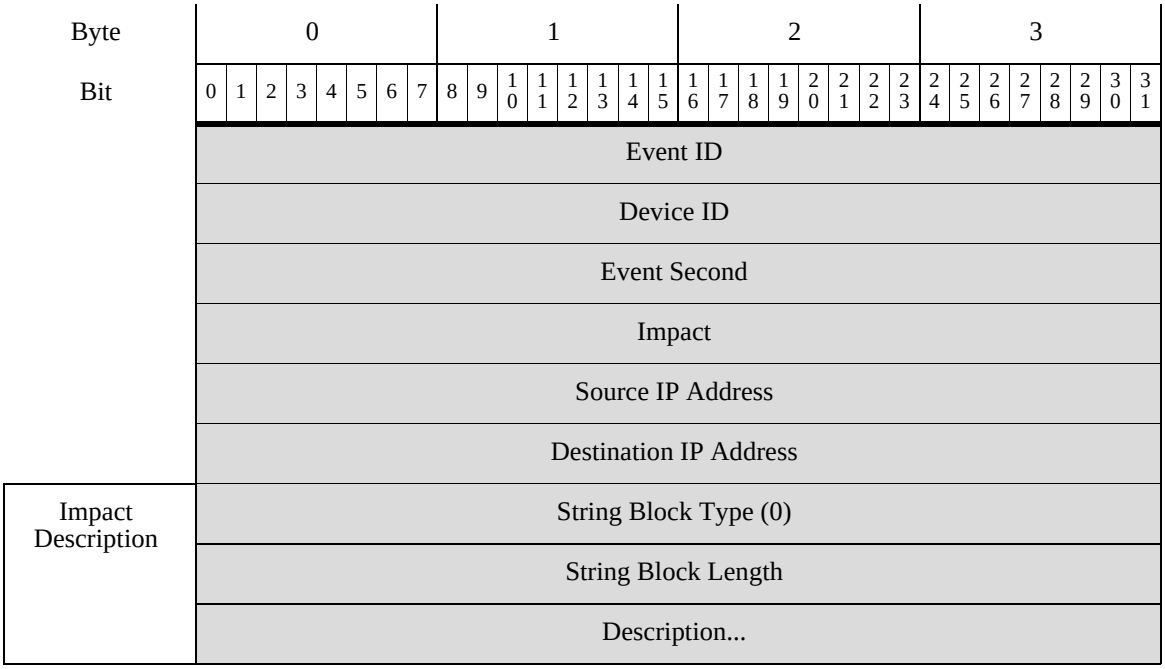
| Field | Data Type | Description |
|---|---|---|
| String Block Type | uint32 | Initiates a String data block containing the name of the ingress VRF. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Ingress VRF name field. |
| Ingress VRF Name | string | The virtual router through which traffic entered the network. |
| String Block Type | uint32 | Initiates a String data block containing the name of the egress VRF. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Egress VRF name field. |
| Egress VRF Name | string | The name of the virtual router through which traffic exited the network. |
| Snort Version | uint8 | Snort version number. |

# Intrusion Impact Alert Data

The Intrusion Impact Alert event contains information about impact events. It is transmitted when an intrusion event is compared to the system network map data and the impact is determined. It uses the standard record header with a record type of 9, followed by an Intrusion Impact Alert data block with a data block type of 20 in the series 1 group of blocks. (The Impact Alert data block is a type of series 1 data block. For more information about series 1 data blocks, see Understanding Discovery (Series 1) Blocks, page 4-62.)

You can request that eStreamer only transmit intrusion impact events by setting bit 5 in the Flags field of the request message. See Event Stream Request Message Format, page 2-12 for more information about request messages. Version 1 of these alerts only handles IPv4. Version 2, introduced in 5.3, handles IPv6 events in addition to IPv4.






The following table describes each data field in an impact event.

***Table B-10 Impact Event Data Fields***

| Field | Data Type | Description |
|---|---|---|
| Intrusion Impact Alert Block Type | uint32 | Indicates that an intrusion impact alert data block follows. This field will always have a value of `20`. See Intrusion Event and Metadata Record Types, page 3-1. |
| Intrusion Impact Alert Block Length | uint32 | Indicates the length of the intrusion impact alert data block, including all data that follows and 8 bytes for the intrusion impact alert block type and length. |
| Event ID | uint32 | Indicates the event identification number. |
| Device ID | uint32 | Indicates the managed device identification number. |
| Event Second | uint32 | Indicates the second (from 01/01/1970) that the event was detected. |

*Table B-10 Impact Event Data Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Impact | bits[8] | Impact flag value of the event. The low-order eight bits indicate the impact level. Values are:<br>• `0x01` (bit 0) — Source or destination host is in a network monitored by the system.<br>• `0x02` (bit 1) — Source or destination host exists in the network map.<br>• `0x04` (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol.<br>• `0x08` (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event.<br>• `0x10` (bit 4) — There is a vulnerability mapped to the server detected in the event.<br>• `0x20` (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface.<br>• `0x40` (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software.<br>• `0x80` (bit 7) — There is a vulnerability mapped to the client detected in the event. (version 5.0+ only)<br>The following impact level values map to specific priorities on the Defense Center. An `X` indicates the value can be `0` or `1`:<br>• (0, unknown): `00X00000`<br>• red (1, vulnerable): `XXXX1XXX`, `XXX1XXXX`, `X1XXXXXX`, `1XXXXXXX` (version 5.0+ only)<br>• orange (2, potentially vulnerable): `00X0011X`<br>• yellow (3, currently not vulnerable): `00X0001X`<br>• blue (4, unknown target): `00X00001` |
| Source IP Address | uint8[4] | IP address of the host associated with the impact event, in IP address octets. |
| Destination IP Address | uint8[4] | IP address of the destination IP address associated with the impact event (if applicable), in IP address octets. This value is `0` if there is no destination IP address. |
| String Block Type | uint32 | Initiates a string data block that contains the impact name. This value is always set to `0`. For more information about string blocks, see String Data Block, page 4-71. |

*Table B-10 Impact Event Data Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | Number of bytes in the event description string block. This includes the four bytes for the string block type, the four bytes for the string block length, and the number of bytes in the description. |
| Description | string | Description of the impact event. |

# Intrusion Event Extra Data Record

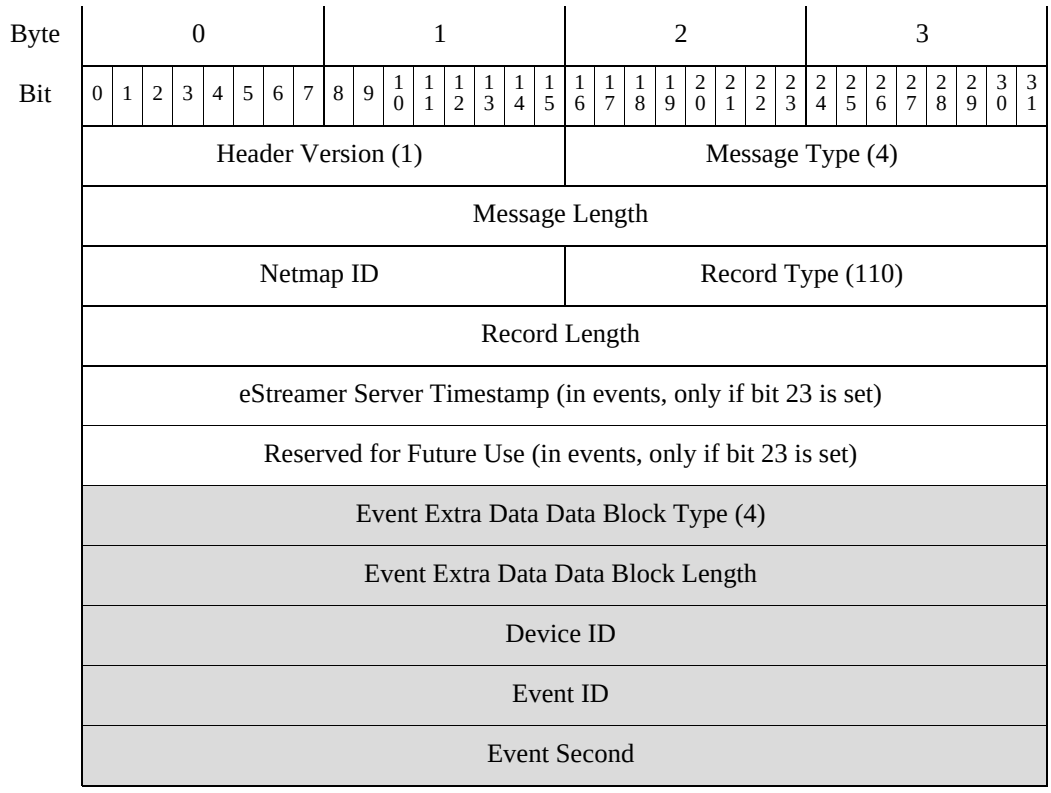
The eStreamer service transmits the event extra data associated with an intrusion event in the Intrusion Event Extra Data record. The record type is always `110`.

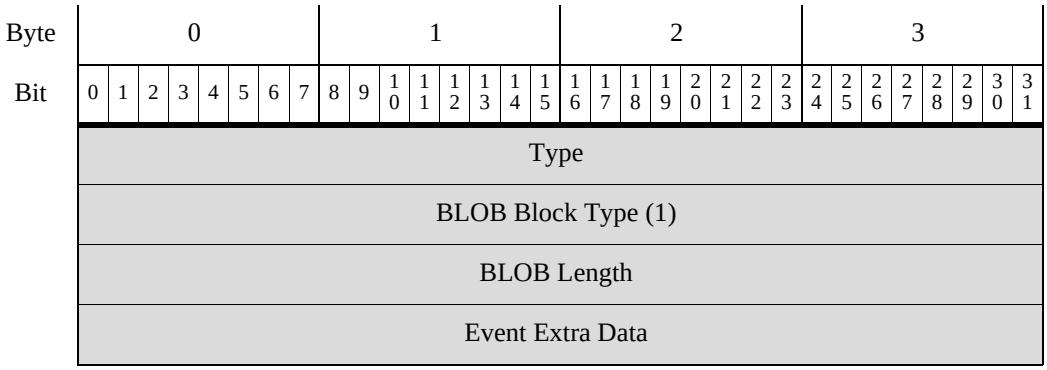
This record is deprecated in version 7.1. While it can still be requested no records will be generated.

The event extra data appears in an encapsulated Event Extra Data data block, which always has a data block type value of `4`. (The Event Extra Data data block is a series 2 data block. For more information about series 2 data blocks, see Understanding Series 2 Data Blocks, page 3-53.)

The supported types of extra data include IPv6 source and destination addresses, as well as the originating IP addresses (v4 or v6) of clients connecting to a web server through an HTTP proxy or load balancer. The graphic below shows the format of the Intrusion Event Extra Data record.

If bit 27 is set in the Request Flags field of the request message, you receive the event extra data for each intrusion event. If you set bit 20, you also receive the event extra data metadata described in Intrusion Event Extra Data Metadata, page B-66. If you enable bit 23, eStreamer will include the extended event header. See Request Flags, page 2-13 for information on setting request flags.






Note that the Event Extra Data block structure includes a BLOB block type, which is one of several variable length data structures introduced in Version 4.10 of the Secure Firewall System.

The following table describes the fields in the Intrusion Event Extra Data record.

***Table B-11 Intrusion Event Extra Data Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| Event Extra Data Data Block Type | uint32 | Initiates an Event Extra Data data block. This value is always `4`. The block type is a series 2 block; for information see Understanding Series 2 Data Blocks, page 3-53. |
| Event Extra Data Data Block Length | uint32 | Length of the data block. Includes the number of bytes of data plus the 8 bytes in the two data block header fields. |
| Device ID | uint32 | The managed device identification number. |
| Event ID | uint32 | The event identification number. |
| Event Second | uint32 | UNIX timestamp of the event (seconds since 01/01/1970). |
| Type | uint32 | Identifier for the type of extra data; for example:<br>• `2` — XFF client (IPv6)<br>• `9` — HTTP URI |
| BLOB Block Type | uint32 | Initiates a BLOB data block containing extra data. This value is always `1`. The block type is a series 2 block. |
| Length | uint32 | Total number of bytes in the BLOB data block. |
| Extra Data | variable | The content of the extra data. The data type is indicated in the Type field. |

# Intrusion Event Extra Data Metadata

The eStreamer service transmits the event extra data metadata associated with intrusion event extra data records in the Intrusion Event Extra Data Metadata record. The record type is always `111`.

This record is deprecated in version 7.1. While it can still be requested no records will be generated.

The event extra data metadata appears in an encapsulated Event Extra Data Metadata data block, which always has a data block type value of `5`. The Event Extra Data data block is a series 2 data block.

If bit 20 is set in the Request Flags field of a request message, you receive the event extra data metadata. If you want to receive both intrusion events and event extra data metadata, you must set bit 2 as well. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Header Version (1) | | | | | | | | | | | | | | | | | Message Type (4) | | | | | | | | | | | | | | | |
| Message Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Netmap ID | | | | | | | | | | | | | | | | | Record Type (111) | | | | | | | | | | | | | | | |
| Record Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| eStreamer Server Timestamp (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reserved for Future Use (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Extra Data Metadata Data Block Type (5) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Data Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Type | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Encoding | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Note that the block structure includes encapsulated String block types, one of several series 2 variable length data structures introduced in Version 4.10 of the Secure Firewall System.

The following table describes the fields in the Event Extra Data Metadata record.

***Table B-12 Event Extra Data Metadata Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| Event Extra Data Metadata Data Block Type | uint32 | Initiates an Event Extra Data Metadata data block. This value is always 5. This block type is a series 2 block. |
| Event Extra Data Metadata Data Block Length | uint32 | Length of the data block. Includes the number of bytes of data plus the 8 bytes in the two data block header fields. |

*Table B-12 Event Extra Data Metadata Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Type | uint32 | The type of extra data. Matches the Type field in the associated Event Extra Data record. This field is the unique key for this record. |
| String Block Type | uint32 | Initiates a String data block for the client application version. This value is always 0. This block type is a series 2 block. |
| String Block Length | uint32 | Number of bytes in the client application version String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the version string. |
| Name | string | Name of the type of event extra data, for example, XFF client (IPv6), and HTTP URI. |
| String Block Type | uint32 | Initiates a string data block for the client application URL. This value is always 0. This block type is a series 2 block. |
| String Block Length | uint32 | Number of bytes in the client application URL String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the URL string. |
| Encoding | string | Encoding used for the event extra data, for example, IPv4, IPv6, or string. |

# Legacy Malware Event Data Structures

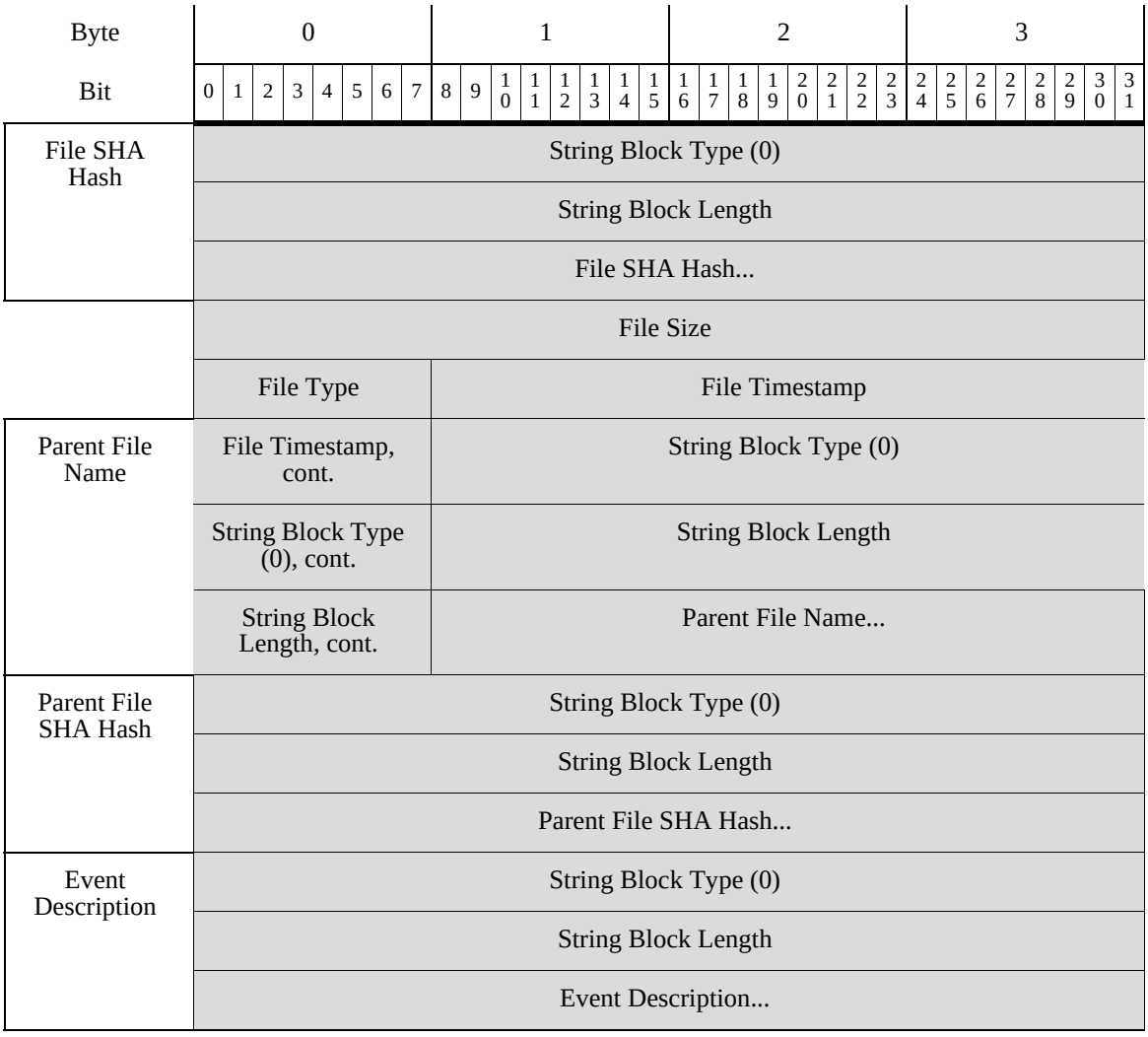

- Malware Event Data Block 5.1, page B-68
- Malware Event Data Block 5.1.1.x, page B-72
- Malware Event Data Block 5.2.x, page B-78
- Malware Event Data Block 5.3, page B-85
- Malware Event Data Block 5.3.1, page B-92
- Malware Event Data Block 5.4.x, page B-99
- Malware Event Data Block 6.x, page B-110


## Malware Event Data Block 5.1

The eStreamer service uses the malware event data block to store information on malware events. These events contain information on malware detected or quarantined within a cloud, the detection method, and hosts and users affected by the malware. The malware event data block has a block type of 16 in the series 2 group of blocks. You request the event as part of the malware event record by setting the malware event flag—bit 30 in the request flags field—in the request message with an event version of 1 and an event code of 101.

The following graphic shows the structure of the malware event data block:

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Malware Event Block Type (16) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Malware Event Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Agent UUID<br>Agent UUID, continued<br>Agent UUID, continued<br>Agent UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID<br>Cloud UUID, continued<br>Cloud UUID, continued<br>Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Type ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Subtype ID | | | | | | | | Host IP Address | | | | | | | | | | | | | | | | | | | | | | | |
| Detection Name | Host IP Address, cont. | | | | | | | | Detector ID | | | | | | | | String Block Type (0) | | | | | | | | | | | | | | | |
| | String Block Type (0), cont. | | | | | | | | | | | | | | | | String Block Length | | | | | | | | | | | | | | | |
| | String Block Length, cont. | | | | | | | | | | | | | | | | Detection Name... | | | | | | | | | | | | | | | |
| User | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | User... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Path | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Path... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| File SHA Hash | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File SHA Hash... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Size | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Type | | | | | | | | File Timestamp | | | | | | | | | | | | | | | | | | | | | | | |
| Parent File Name | File Timestamp, cont. | | | | | | | | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Type (0), cont. | | | | | | | | String Block Length | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length, cont. | | | | | | | | Parent File Name... | | | | | | | | | | | | | | | | | | | | | | | |
| Parent File SHA Hash | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Parent File SHA Hash... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Description | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Description... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

The following table describes the fields in the malware event data block.

***Table B-13 Malware Event Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| Malware Event Block Type | uint32 | Initiates a malware event data block. This value is always `16`. |
| Malware Event Block Length | uint32 | Total number of bytes in the malware event data block, including eight bytes for the malware event block type and length fields, plus the number of bytes of data that follows. |
| Agent UUID | uint8[16] | The internal unique ID of the AMP for Endpoints agent reporting the malware event. |
| Cloud UUID | uint8[16] | The internal unique ID of the malware awareness network from which the malware event originated. |
| Timestamp | uint32 | The malware event generation timestamp. |
| Event Type ID | uint32 | The internal ID of the malware event type. |

*Table B-13 Malware Event Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Event Subtype ID | uint8 | The internal ID of the action that led to malware detection. |
| Host IP Address | uint32 | The host IP address associated with the malware event. |
| Detector ID | uint8 | The internal ID of the detection technology that detected the malware. |
| String Block Type | uint32 | Initiates a String data block containing the detection name. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the Detection Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Detection Name field. |
| Detection Name | string | The name of the detected or quarantined malware. |
| String Block Type | uint32 | Initiates a String data block containing the username. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the User String data block, including eight bytes for the block type and header fields plus the number of bytes in the User field. |
| User | string | The user of the computer where the Cisco Agent is installed and where the malware event occurred. Note that these users are not tied to user discovery. |
| String Block Type | uint32 | Initiates a String data block containing the file name. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Name field. |
| File Name | string | The name of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file path. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Path String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Path field. |
| File Path | string | The file path, not including the file name, of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file SHA hash. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the File SHA Hash field. |
| File SHA Hash | string | The SHA-256 hash value of the detected or quarantined file. |
| File Size | uint32 | The size in bytes of the detected or quarantined file. |
| File Type | uint8 | The file type of the detected or quarantined file. |
| File Timestamp | uint32 | The creation timestamp of the detected or quarantined file. |

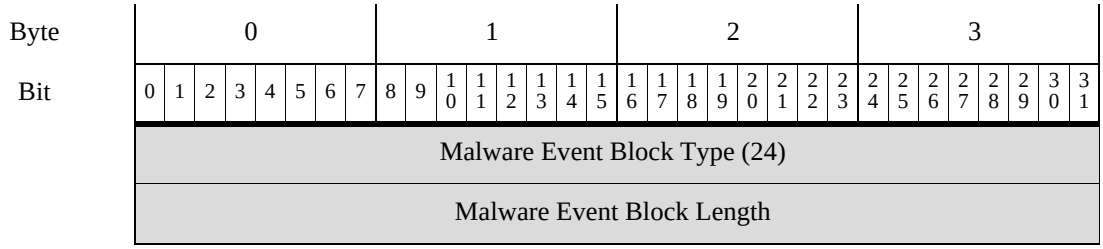
*Table B-13 Malware Event Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Type | uint32 | Initiates a String data block containing the parent file name. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the Parent File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File Name field. |
| Parent File Name | string | The name of the file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the parent file SHA hash. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the Parent File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File SHA Hash field. |
| Parent File SHA Hash | string | The SHA-256 hash value of the parent file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the event description. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the Event Description String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Description field. |
| Event Description | string | The additional event information associated with the event type. |

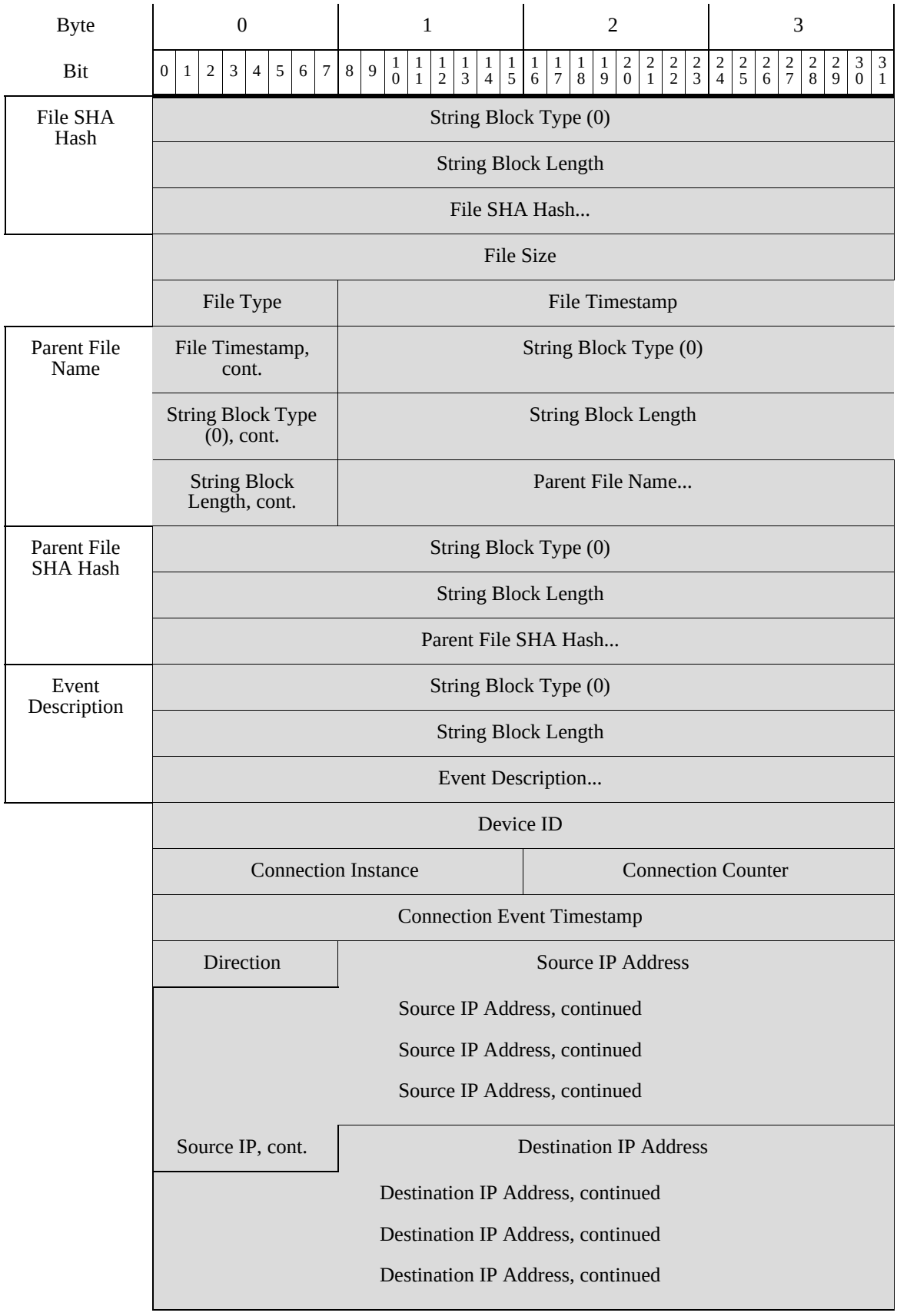
# Malware Event Data Block 5.1.1.x

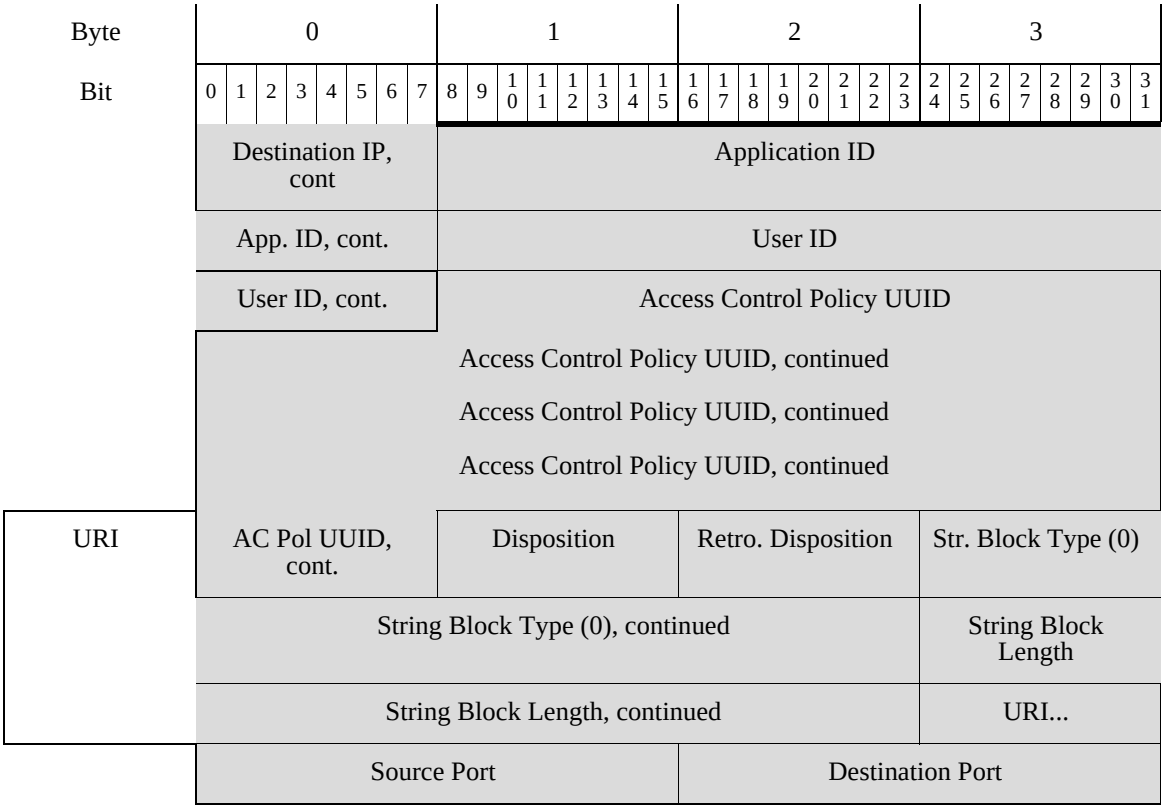
The eStreamer service uses the malware event data block to store information on malware events. These events contain information on malware detected or quarantined within a cloud, the detection method, and hosts and users affected by the malware. The malware event data block has a block type of 24 in the series 2 group of blocks. You request the event as part of the malware event record by setting the malware event flag—bit 30 in the request flags field—in the request message with an event version of 2 and an event code of 101.

The following graphic shows the structure of the malware event data block:



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Agent UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Agent UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Agent UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Agent UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Malware Event Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Type ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Subtype ID | | | | | | | | Host IP Address | | | | | | | | | | | | | | | | | | | | | | | |
| Detection Name | Host IP Address, cont. | | | | | | | | Detector ID | | | | | | | | String Block Type (0) | | | | | | | | | | | | | | | |
| | String Block Type (0), cont. | | | | | | | | | | | | | | | | String Block Length | | | | | | | | | | | | | | | |
| | String Block Length, cont. | | | | | | | | | | | | | | | | Detection Name... | | | | | | | | | | | | | | | |
| User | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | User... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Path | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Path... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
File SHA Hash
String Block Type (0)
String Block Length
File SHA Hash...
File Size
File Type
File Timestamp
Parent File Name
File Timestamp, cont.
String Block Type (0)
String Block Type (0), cont.
String Block Length
String Block Length, cont.
Parent File Name...
Parent File SHA Hash
String Block Type (0)
String Block Length
Parent File SHA Hash...
Event Description
String Block Type (0)
String Block Length
Event Description...
Device ID
Connection Instance
Connection Counter
Connection Event Timestamp
Direction
Source IP Address
Source IP Address, continued
Source IP Address, continued
Source IP Address, continued
Source IP, cont.
Destination IP Address
Destination IP Address, continued
Destination IP Address, continued
Destination IP Address, continued




The following table describes the fields in the malware event data block.

***Table B-14 Malware Event Data Block for 5.1.1.x Fields***

| Field | Data Type | Description |
|---|---|---|
| Malware Event Block Type | uint32 | Initiates a malware event data block. This value is always `24`. |
| Malware Event Block Length | uint32 | Total number of bytes in the malware event data block, including eight bytes for the malware event block type and length fields, plus the number of bytes of data that follows. |
| Agent UUID | uint8[16] | The internal unique ID of the AMP for Endpoints agent reporting the malware event. |
| Cloud UUID | uint8[16] | The internal unique ID of the malware awareness network from which the malware event originated. |
| Malware Event Timestamp | uint32 | The malware event generation timestamp. |
| Event Type ID | uint32 | The internal ID of the malware event type. |
| Event Subtype ID | uint8 | The internal ID of the action that led to malware detection. |
| Host IP Address | uint32 | The host IP address associated with the malware event. |
| Detector ID | uint8 | The internal ID of the detection technology that detected the malware. |
| String Block Type | uint32 | Initiates a String data block containing the detection name. This value is always `0`. |

*Table B-14 Malware Event Data Block for 5.1.1.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the Detection Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Detection Name field. |
| Detection Name | string | The name of the detected or quarantined malware. |
| String Block Type | uint32 | Initiates a String data block containing the username. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the User String data block, including eight bytes for the block type and header fields plus the number of bytes in the User field. |
| User | string | The user of the computer where the Cisco Agent is installed and where the malware event occurred. Note that these users are not tied to user discovery. |
| String Block Type | uint32 | Initiates a String data block containing the file name. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Name field. |
| File Name | string | The name of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file path. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Path String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Path field. |
| File Path | string | The file path, not including the file name, of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file SHA hash. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the File SHA Hash field. |
| File SHA Hash | string | The rendered string of the SHA-256 hash value of the detected or quarantined file. |
| File Size | uint32 | The size in bytes of the detected or quarantined file. |
| File Type | uint8 | The file type of the detected or quarantined file. |
| File Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the creation of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the parent file name. This value is always 0. |

*Table B-14 Malware Event Data Block for 5.1.1.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the Parent File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File Name field. |
| Parent File Name | string | The name of the file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the parent file SHA hash. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Parent File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File SHA Hash field. |
| Parent File SHA Hash | string | The SHA-256 hash value of the parent file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the event description. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Event Description String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Description field. |
| Event Description | string | The additional event information associated with the event type. |
| Device ID | uint32 | ID for the device that generated the event. |
| Connection Instance | uint16 | Snort instance on the device that generated the event. Used to link the event with a connection or IDS event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Connection Event Timestamp | uint32 | Timestamp of the connection event. |
| Direction | uint8 | Indicates whether the file was uploaded or downloaded. Can have the following values:<br>• `1` — Download<br>• `2` — Upload<br>Currently the value depends on the protocol (for example, if the connection is HTTP it is a download). |
| Source IP Address | uint8[16] | IPv4 or IPv6 address for the source of the connection. |
| Destination IP Address | uint8[16] | IPv4 or IPv6 address for the destination of the connection. |
| Application ID | uint32 | ID number that maps to the application using the file transfer. |
| User ID | uint32 | Identification number for the user logged into the destination host, as identified by the system. |

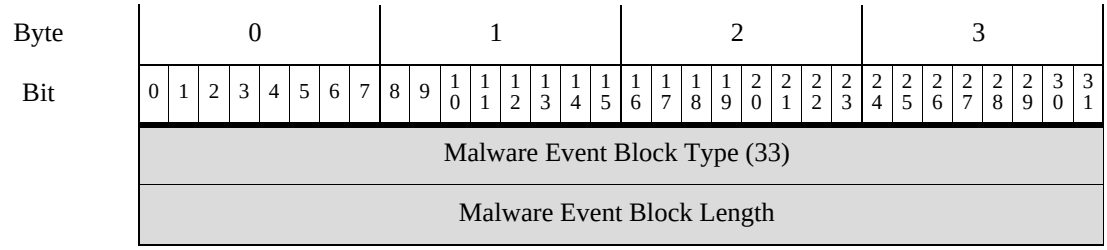
*Table B-14 Malware Event Data Block for 5.1.1.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Access Control Policy UUID | uint8[16] | Identification number that acts as a unique identifier for the access control policy that triggered the event. |
| Disposition | uint8 | The malware status of the file. Possible values include:<br>• 1 — CLEAN — The file is clean and does not contain malware.<br>• 2 — UNKNOWN — It is unknown whether the file contains malware.<br>• 3 — MALWARE — The file contains malware.<br>• 4 — CACHE_MISS — The software was unable to send a request to the Cisco cloud for a disposition.<br>• 5 — NO_CLOUD_RESP — The Cisco cloud services did not respond to the request. |
| Retrospective Disposition | uint8 | Disposition of the file if the disposition is updated. If the disposition is not updated, this field contains the same value as the Disposition field. The possible values are the same as the Disposition field. |
| String Block Type | uint32 | Initiates a String data block containing the URI. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the URI data block, including eight bytes for the block type and header fields plus the number of bytes in the URI field. |
| URI | string | URI of the connection. |
| Source Port | uint16 | Port number for the source of the connection. |
| Destination Port | uint16 | Port number for the destination of the connection. |

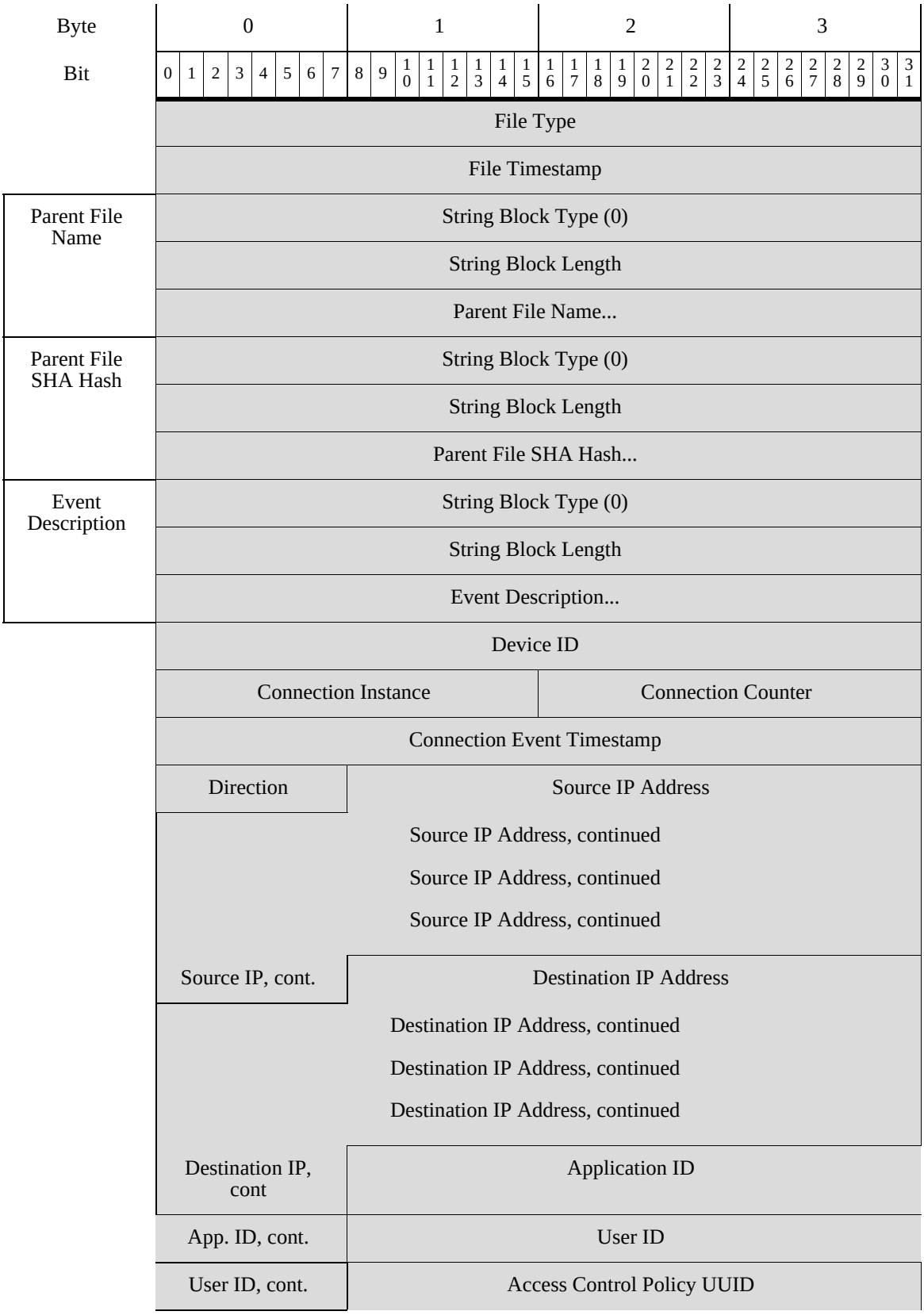
# Malware Event Data Block 5.2.x

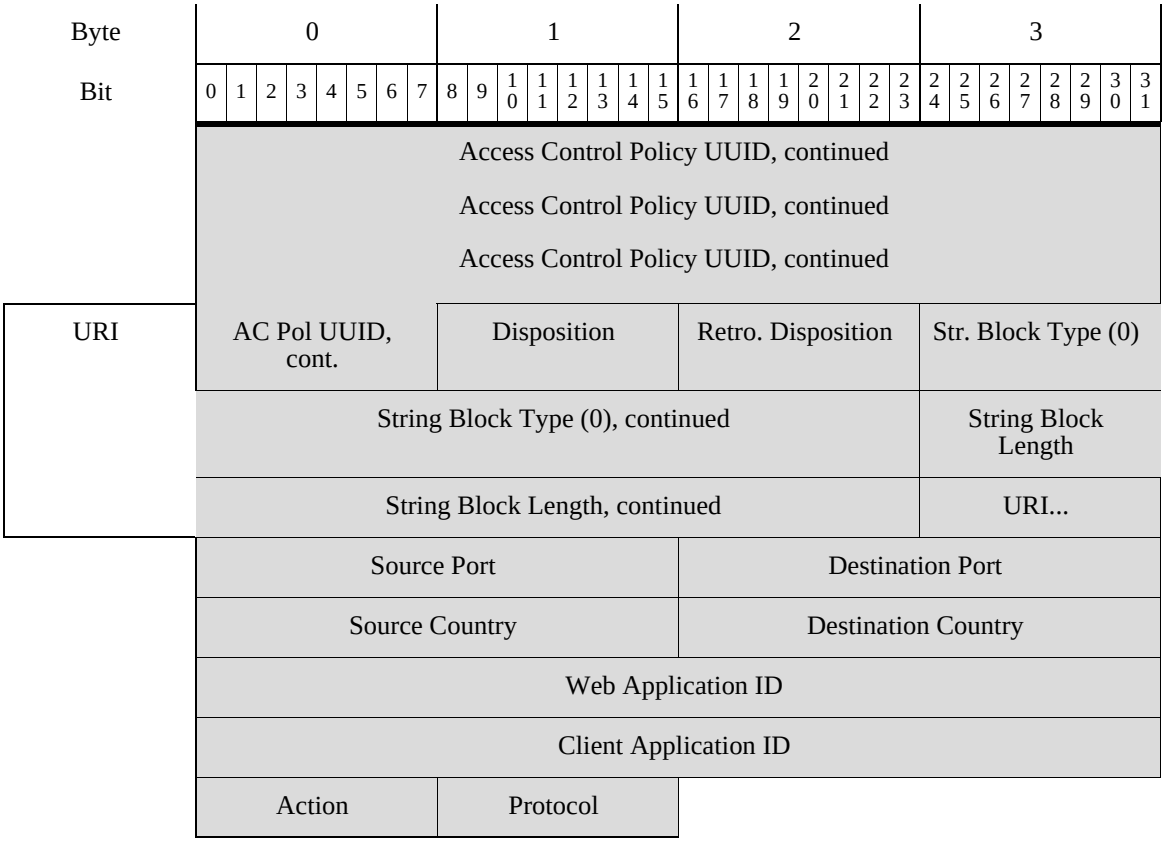
The eStreamer service uses the malware event data block to store information on malware events. These events contain information on malware detected or quarantined within a cloud, the detection method, and hosts and users affected by the malware. The malware event data block has a block type of 33 in the series 2 group of blocks. You request the event as part of the malware event record by setting the malware event flag—bit 30 in the request flags field—in the request message with an event version of 3 and an event code of 101.

The following graphic shows the structure of the malware event data block:



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Agent UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Agent UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Agent UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Agent UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Malware Event Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Type ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Detection Name | Event Subtype ID | | | | | | | | Detector ID | | | | | | | | String Block Type (0) | | | | | | | | | | | | | | | |
| | String Block Type (0), cont. | | | | | | | | | | | | | | | | String Block Length | | | | | | | | | | | | | | | |
| | String Block Length, cont. | | | | | | | | | | | | | | | | Detection Name... | | | | | | | | | | | | | | | |
| User | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | User... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Path | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Path... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File SHA Hash | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File SHA Hash... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Size | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
File Type
File Timestamp
Parent File Name
String Block Type (0)
String Block Length
Parent File Name...
Parent File SHA Hash
String Block Type (0)
String Block Length
Parent File SHA Hash...
Event Description
String Block Type (0)
String Block Length
Event Description...
Device ID
Connection Instance
Connection Counter
Connection Event Timestamp
Direction
Source IP Address
Source IP Address, continued
Source IP Address, continued
Source IP Address, continued
Source IP, cont.
Destination IP Address
Destination IP Address, continued
Destination IP Address, continued
Destination IP Address, continued
Destination IP, cont
Application ID
App. ID, cont.
User ID
User ID, cont.
Access Control Policy UUID




The following table describes the fields in the malware event data block.

***Table B-15 Malware Event Data Block for 5.2.x Fields***

| Field | Data Type | Description |
|---|---|---|
| Malware Event Block Type | uint32 | Initiates a malware event data block. This value is always 33. |
| Malware Event Block Length | uint32 | Total number of bytes in the malware event data block, including eight bytes for the malware event block type and length fields, plus the number of bytes of data that follows. |
| Agent UUID | uint8[16] | The internal unique ID of the AMP for Endpoints agent reporting the malware event. |
| Cloud UUID | uint8[16] | The internal unique ID of the malware awareness network from which the malware event originated. |
| Malware Event Timestamp | uint32 | The malware event generation timestamp. |
| Event Type ID | uint32 | The internal ID of the malware event type. |
| Event Subtype ID | uint8 | The internal ID of the action that led to malware detection. |
| Detector ID | uint8 | The internal ID of the detection technology that detected the malware. |
| String Block Type | uint32 | Initiates a String data block containing the detection name. This value is always 0. |

*Table B-15 Malware Event Data Block for 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the Detection Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Detection Name field. |
| Detection Name | string | The name of the detected or quarantined malware. |
| String Block Type | uint32 | Initiates a String data block containing the username. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the User String data block, including eight bytes for the block type and header fields plus the number of bytes in the User field. |
| User | string | The user of the computer where the Cisco Agent is installed and where the malware event occurred. Note that these users are not tied to user discovery. |
| String Block Type | uint32 | Initiates a String data block containing the file name. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Name field. |
| File Name | string | The name of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file path. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Path String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Path field. |
| File Path | string | The file path, not including the file name, of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file SHA hash. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the File SHA Hash field. |
| File SHA Hash | string | The rendered string of the SHA-256 hash value of the detected or quarantined file. |
| File Size | uint32 | The size in bytes of the detected or quarantined file. |
| File Type | uint8 | The file type of the detected or quarantined file. |
| File Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the creation of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the parent file name. This value is always 0. |

*Table B-15 Malware Event Data Block for 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the Parent File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File Name field. |
| Parent File Name | string | The name of the file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the parent file SHA hash. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Parent File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File SHA Hash field. |
| Parent File SHA Hash | string | The SHA-256 hash value of the parent file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the event description. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Event Description String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Description field. |
| Event Description | string | The additional event information associated with the event type. |
| Device ID | uint32 | ID for the device that generated the event. |
| Connection Instance | uint16 | Snort instance on the device that generated the event. Used to link the event with a connection or IDS event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Connection Event Timestamp | uint32 | Timestamp of the connection event. |
| Direction | uint8 | Indicates whether the file was uploaded or downloaded. Can have the following values:<br>• `1` — Download<br>• `2` — Upload<br>Currently the value depends on the protocol (for example, if the connection is HTTP it is a download). |
| Source IP Address | uint8[16] | IPv4 or IPv6 address for the source of the connection. |
| Destination IP Address | uint8[16] | IPv4 or IPv6 address for the destination of the connection. |
| Application ID | uint32 | ID number that maps to the application using the file transfer. |
| User ID | uint32 | Identification number for the user logged into the destination host, as identified by the system. |

*Table B-15 Malware Event Data Block for 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Access Control Policy UUID | uint8[16] | Identification number that acts as a unique identifier for the access control policy that triggered the event. |
| Disposition | uint8 | The malware status of the file. Possible values include:<br>• 1 — CLEAN — The file is clean and does not contain malware.<br>• 2 — NEUTRAL — It is unknown whether the file contains malware.<br>• 3 — MALWARE — The file contains malware.<br>• 4 — CACHE_MISS — The software was unable to send a request to the Cisco cloud for a disposition, or the Cisco cloud services did not respond to the request. |
| Retrospective Disposition | uint8 | Disposition of the file if the disposition is updated. If the disposition is not updated, this field contains the same value as the Disposition field. The possible values are the same as the Disposition field. |
| String Block Type | uint32 | Initiates a String data block containing the URI. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the URI data block, including eight bytes for the block type and header fields plus the number of bytes in the URI field. |
| URI | string | URI of the connection. |
| Source Port | uint16 | Port number for the source of the connection. |
| Destination Port | uint16 | Port number for the destination of the connection. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint 16 | Code for the country of the destination host. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |

*Table B-15 Malware Event Data Block for 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Action | uint8 | The action taken on the file based on the file type. Can have the following values:<br>• 1 — Detect<br>• 2 — Block<br>• 3 — Malware Cloud Lookup<br>• 4 — Malware Block<br>• 5 — Malware Allow List |
| Protocol | uint8 | IANA protocol number specified by the user. For example:<br>• 1 — ICMP<br>• 4 — IP<br>• 6 — TCP<br>• 17 — UDP<br>This is currently only TCP. |

# Malware Event Data Block 5.3

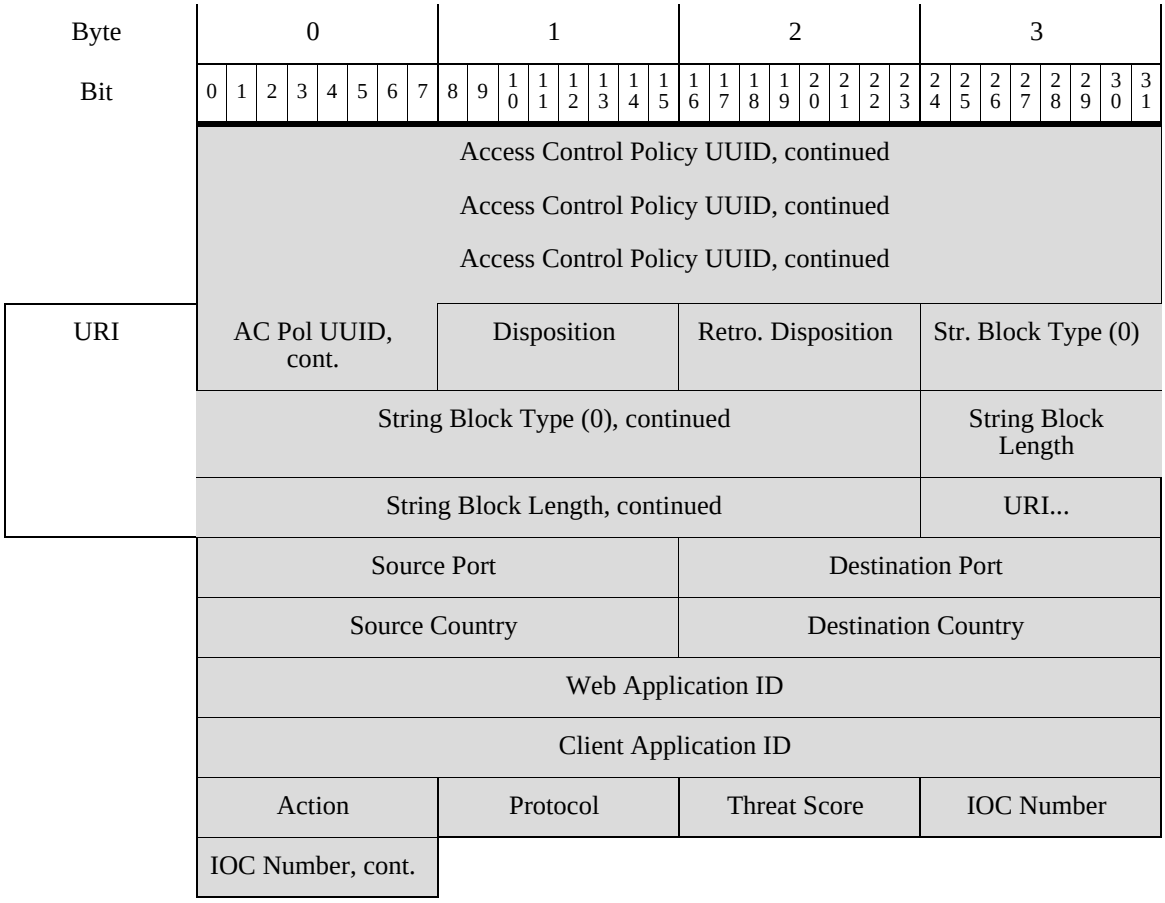
The eStreamer service uses the malware event data block to store information on malware events. These events contain information on malware detected or quarantined within a cloud, the detection method, and hosts and users affected by the malware. The malware event data block has a block type of 35 in the series 2 group of blocks. You request the event as part of the malware event record by setting the malware event flag—bit 30 in the request flags field—in the request message with an event version of 4 and an event code of 101.

The following graphic shows the structure of the malware event data block:

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Malware Event Block Type (35) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Malware Event Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Agent UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Agent UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Agent UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Agent UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Cloud UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Malware Event Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Type ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Subtype ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Detection Name | Detector ID | | | | | | | | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Type (0), cont. | | | | | | | | String Block Length | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length, cont. | | | | | | | | Detection Name... | | | | | | | | | | | | | | | | | | | | | | | |
| User | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | User... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Path | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Path... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File SHA Hash | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File SHA Hash... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Size | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Type | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Parent File Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Parent File Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Parent File SHA Hash | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Parent File SHA Hash... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event Description | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Description... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Connection Instance | | | | | | | | | | | | | | | | Connection Counter | | | | | | | | | | | | | | | |
| | Connection Event Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Direction | | | | | | | | Source IP Address | | | | | | | | | | | | | | | | | | | | | | | |
| | Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Source IP, cont. | | | | | | | | Destination IP Address | | | | | | | | | | | | | | | | | | | | | | | |
| | Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Destination IP, cont | | | | | | | | Application ID | | | | | | | | | | | | | | | | | | | | | | | |
| | App. ID, cont. | | | | | | | | User ID | | | | | | | | | | | | | | | | | | | | | | | |
| | User ID, cont. | | | | | | | | Access Control Policy UUID | | | | | | | | | | | | | | | | | | | | | | | |




The following table describes the fields in the malware event data block.

***Table B-16 Malware Event Data Block for 5.3 Fields***

| Field | Data Type | Description |
|---|---|---|
| Malware Event Block Type | uint32 | Initiates a malware event data block. This value is always `35`. |
| Malware Event Block Length | uint32 | Total number of bytes in the malware event data block, including eight bytes for the malware event block type and length fields, plus the number of bytes of data that follows. |
| Agent UUID | uint8[16] | The internal unique ID of the AMP for Endpoints agent reporting the malware event. |
| Cloud UUID | uint8[16] | The internal unique ID of the malware awareness network from which the malware event originated. |
| Malware Event Timestamp | uint32 | The malware event generation timestamp. |
| Event Type ID | uint32 | The internal ID of the malware event type. |
| Event Subtype ID | uint32 | The internal ID of the action that led to malware detection. |
| Detector ID | uint8 | The internal ID of the detection technology that detected the malware. |

*Table B-16 Malware Event Data Block for 5.3 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Type | uint32 | Initiates a String data block containing the detection name. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the Detection Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Detection Name field. |
| Detection Name | string | The name of the detected or quarantined malware. |
| String Block Type | uint32 | Initiates a String data block containing the username. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the User String data block, including eight bytes for the block type and header fields plus the number of bytes in the User field. |
| User | string | The user of the computer where the Cisco Agent is installed and where the malware event occurred. Note that these users are not tied to user discovery. |
| String Block Type | uint32 | Initiates a String data block containing the file name. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Name field. |
| File Name | string | The name of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file path. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Path String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Path field. |
| File Path | string | The file path, not including the file name, of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file SHA hash. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the File SHA Hash field. |
| File SHA Hash | string | The rendered string of the SHA-256 hash value of the detected or quarantined file. |
| File Size | uint32 | The size in bytes of the detected or quarantined file. |
| File Type | uint8 | The file type of the detected or quarantined file. The meaning of this field is transmitted in the metadata with this event. See AMP for Endpoints File Type Metadata, page 3-38 for more information. |
| File Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the creation of the detected or quarantined file. |

*Table B-16 Malware Event Data Block for 5.3 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Type | uint32 | Initiates a String data block containing the parent file name. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Parent File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File Name field. |
| Parent File Name | string | The name of the file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the parent file SHA hash. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Parent File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File SHA Hash field. |
| Parent File SHA Hash | string | The SHA-256 hash value of the parent file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the event description. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Event Description String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Description field. |
| Event Description | string | The additional event information associated with the event type. |
| Device ID | uint32 | ID for the device that generated the event. |
| Connection Instance | uint16 | Snort instance on the device that generated the event. Used to link the event with a connection or IDS event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Connection Event Timestamp | uint32 | Timestamp of the connection event. |
| Direction | uint8 | Indicates whether the file was uploaded or downloaded. Can have the following values:<br>• `1` — Download<br>• `2` — Upload<br>Currently the value depends on the protocol (for example, if the connection is HTTP it is a download). |
| Source IP Address | uint8[16] | IPv4 or IPv6 address for the source of the connection. |
| Destination IP Address | uint8[16] | IPv4 or IPv6 address for the destination of the connection. |
| Application ID | uint32 | ID number that maps to the application using the file transfer. |

*Table B-16 Malware Event Data Block for 5.3 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| User ID | uint32 | Identification number for the user logged into the destination host, as identified by the system. |
| Access Control Policy UUID | uint8[16] | Identification number that acts as a unique identifier for the access control policy that triggered the event. |
| Disposition | uint8 | The malware status of the file. Possible values include:<br>• 1 — CLEAN The file is clean and does not contain malware.<br>• 2 — UNKNOWN It is unknown whether the file contains malware.<br>• 3 — MALWARE The file contains malware.<br>• 4 — UNAVAILABLE The software was unable to send a request to the Cisco cloud for a disposition, or the Cisco cloud services did not respond to the request.<br>• 5 — CUSTOM SIGNATURE The file matches a user-defined hash, and is treated in a fashion designated by the user. |
| Retrospective Disposition | uint8 | Disposition of the file if the disposition is updated. If the disposition is not updated, this field contains the same value as the Disposition field. The possible values are the same as the Disposition field. |
| String Block Type | uint32 | Initiates a String data block containing the URI. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the URI data block, including eight bytes for the block type and header fields plus the number of bytes in the URI field. |
| URI | string | URI of the connection. |
| Source Port | uint16 | Port number for the source of the connection. |
| Destination Port | uint16 | Port number for the destination of the connection. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint 16 | Code for the country of the destination host. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |

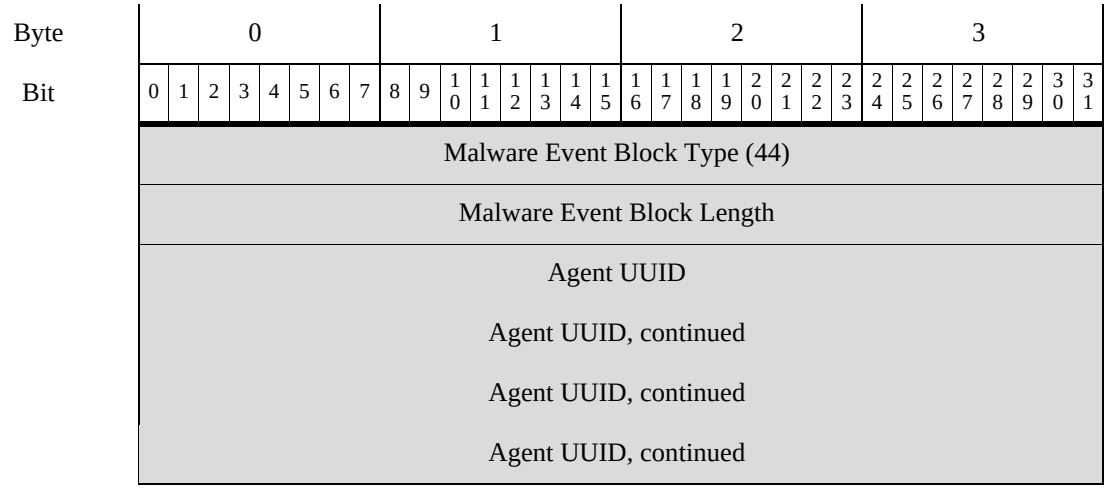
*Table B-16 Malware Event Data Block for 5.3 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Action | uint8 | The action taken on the file based on the file type. Can have the following values:<br>• 1 — Detect<br>• 2 — Block<br>• 3 — Malware Cloud Lookup<br>• 4 — Malware Block<br>• 5 — Malware Allow List |
| Protocol | uint8 | IANA protocol number specified by the user. For example:<br>• 1 — ICMP<br>• 4 — IP<br>• 6 — TCP<br>• 17 — UDP<br>This is currently only TCP. |
| Threat Score | uint8 | A numeric value from 0 to 100 based on the potentially malicious behaviors observed during dynamic analysis. |
| IOC Number | uint16 | ID Number of the compromise associated with this event. |

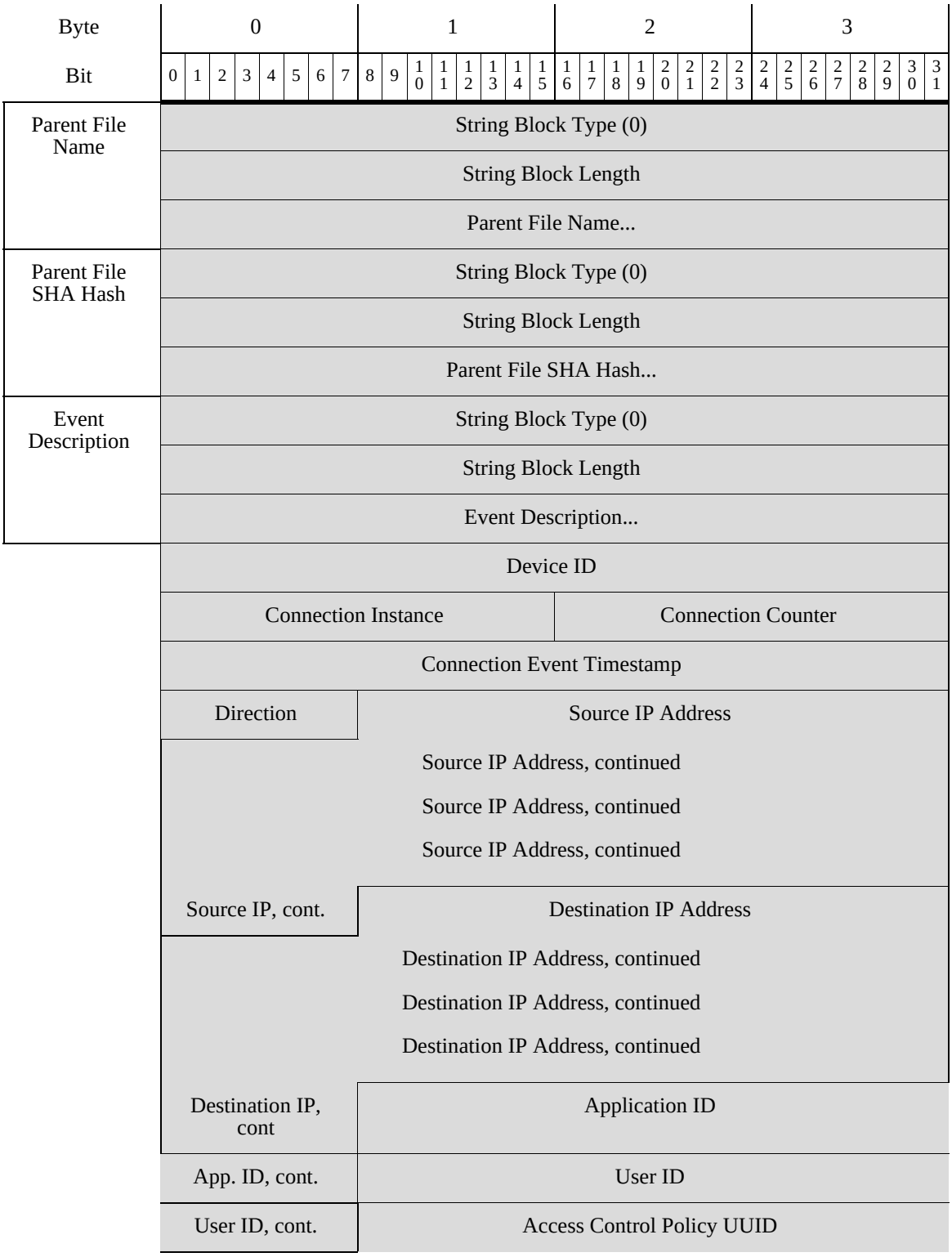
# Malware Event Data Block 5.3.1

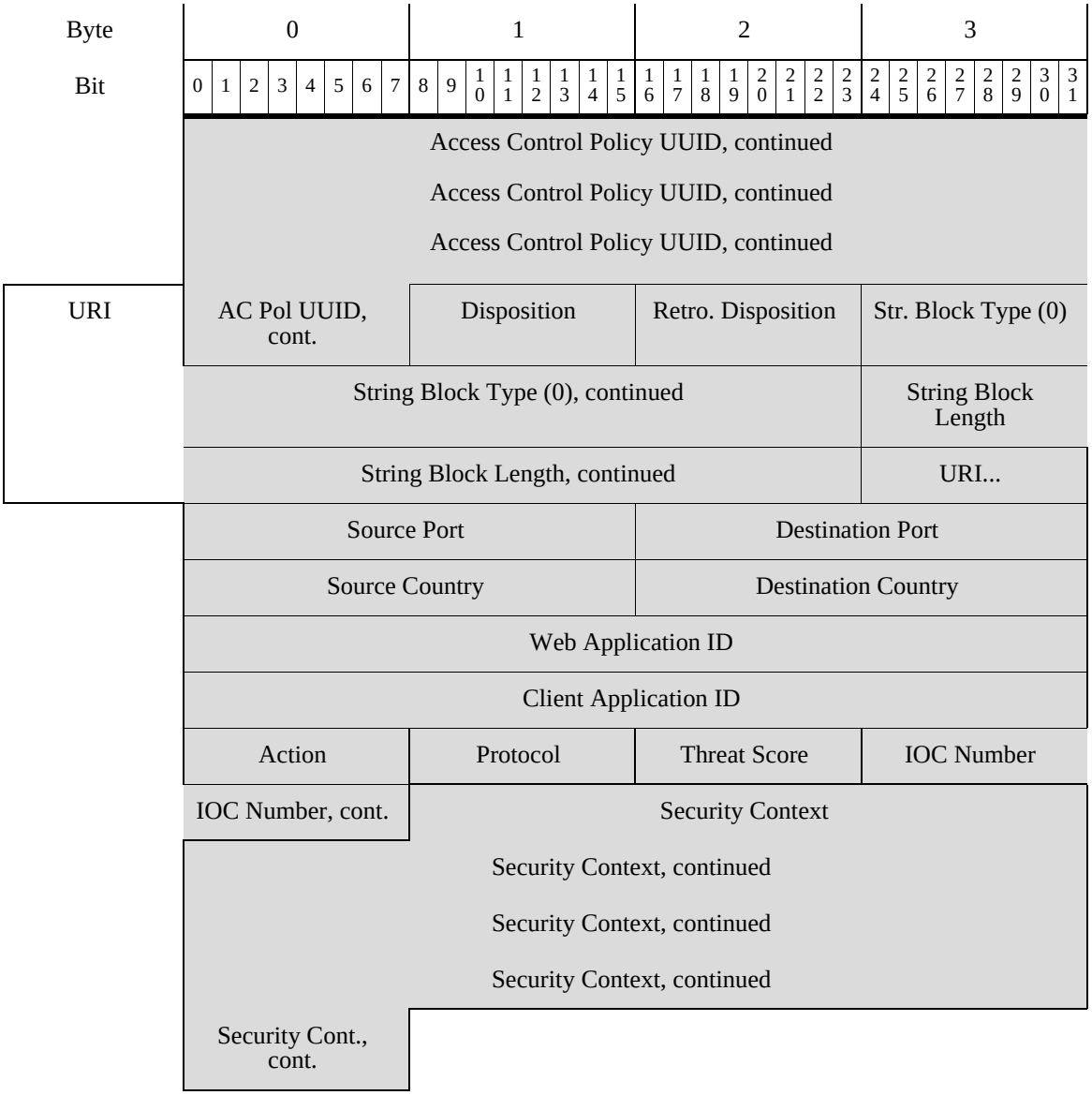
The eStreamer service uses the malware event data block to store information on malware events. These events contain information on malware detected or quarantined within a cloud, the detection method, and hosts and users affected by the malware. The malware event data block has a block type of 44 in the series 2 group of blocks. It supersedes block 35. You request the event as part of the malware event record by setting the malware event flag—bit 30 in the request flags field—in the request message with an event version of 5 and an event code of 101.

The following graphic shows the structure of the malware event data block:



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Cloud UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Malware Event Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Type ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Subtype ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Detection Name | Detector ID | | | | | | | | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Type (0), cont. | | | | | | | | String Block Length | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length, cont. | | | | | | | | Detection Name... | | | | | | | | | | | | | | | | | | | | | | | |
| User | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | User... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Path | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Path... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File SHA Hash | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File SHA Hash... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Size | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Type | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Parent File Name
String Block Type (0)
String Block Length
Parent File Name...
Parent File SHA Hash
String Block Type (0)
String Block Length
Parent File SHA Hash...
Event Description
String Block Type (0)
String Block Length
Event Description...
Device ID
Connection Instance
Connection Counter
Connection Event Timestamp
Direction
Source IP Address
Source IP Address, continued
Source IP Address, continued
Source IP Address, continued
Source IP, cont.
Destination IP Address
Destination IP Address, continued
Destination IP Address, continued
Destination IP Address, continued
Destination IP, cont
Application ID
App. ID, cont.
User ID
User ID, cont.
Access Control Policy UUID




The following table describes the fields in the malware event data block.

***Table B-17 Malware Event Data Block for 5.3.1 Fields***

| Field | Data Type | Description |
|---|---|---|
| Malware Event Block Type | uint32 | Initiates a malware event data block. This value is always `44`. |
| Malware Event Block Length | uint32 | Total number of bytes in the malware event data block, including eight bytes for the malware event block type and length fields, plus the number of bytes of data that follows. |
| Agent UUID | uint8[16] | The internal unique ID of the AMP for Endpoints agent reporting the malware event. |
| Cloud UUID | uint8[16] | The internal unique ID of the Cisco Advanced Malware Protection cloud from which the malware event originated. |

*Table B-17 Malware Event Data Block for 5.3.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Malware Event Timestamp | uint32 | The malware event generation timestamp. |
| Event Type ID | uint32 | The internal ID of the malware event type. |
| Event Subtype ID | uint32 | The internal ID of the action that led to malware detection. |
| Detector ID | uint8 | The internal ID of the detection technology that detected the malware. |
| String Block Type | uint32 | Initiates a String data block containing the detection name. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the Detection Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Detection Name field. |
| Detection Name | string | The name of the detected or quarantined malware. |
| String Block Type | uint32 | Initiates a String data block containing the username. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the User String data block, including eight bytes for the block type and header fields plus the number of bytes in the User field. |
| User | string | The user of the computer where the Cisco Agent is installed and where the malware event occurred. Note that these users are not tied to user discovery. |
| String Block Type | uint32 | Initiates a String data block containing the file name. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Name field. |
| File Name | string | The name of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file path. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Path String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Path field. |
| File Path | string | The file path, not including the file name, of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file SHA hash. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the File SHA Hash field. |
| File SHA Hash | string | The rendered string of the SHA-256 hash value of the detected or quarantined file. |

*Table B-17 Malware Event Data Block for 5.3.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| File Size | uint32 | The size in bytes of the detected or quarantined file. |
| File Type | uint8 | The file type of the detected or quarantined file. The meaning of this field is transmitted in the metadata with this event. See AMP for Endpoints File Type Metadata, page 3-38 for more information. |
| File Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the creation of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the parent file name. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the Parent File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File Name field. |
| Parent File Name | string | The name of the file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the parent file SHA hash. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the Parent File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File SHA Hash field. |
| Parent File SHA Hash | string | The SHA-256 hash value of the parent file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the event description. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the Event Description String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Description field. |
| Event Description | string | The additional event information associated with the event type. |
| Device ID | uint32 | ID for the device that generated the event. |
| Connection Instance | uint16 | Snort instance on the device that generated the event. Used to link the event with a connection or IDS event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Connection Event Timestamp | uint32 | Timestamp of the connection event. |

*Table B-17 Malware Event Data Block for 5.3.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Direction | uint8 | Indicates whether the file was uploaded or downloaded. Can have the following values:<br>• 1 — Download<br>• 2 — Upload<br>Currently the value depends on the protocol (for example, if the connection is HTTP it is a download). |
| Source IP Address | uint8[16] | IPv4 or IPv6 address for the source of the connection. |
| Destination IP Address | uint8[16] | IPv4 or IPv6 address for the destination of the connection. |
| Application ID | uint32 | ID number that maps to the application using the file transfer. |
| User ID | uint32 | Identification number for the user logged into the destination host, as identified by the system. |
| Access Control Policy UUID | uint8[16] | Identification number that acts as a unique identifier for the access control policy that triggered the event. |
| Disposition | uint8 | The malware status of the file. Possible values include:<br>• 1 — CLEAN The file is clean and does not contain malware.<br>• 2 — UNKNOWN It is unknown whether the file contains malware.<br>• 3 — MALWARE The file contains malware.<br>• 4 — UNAVAILABLE The software was unable to send a request to the Cisco cloud for a disposition, or the Cisco cloud services did not respond to the request.<br>• 5 — CUSTOM SIGNATURE The file matches a user-defined hash, and is treated in a fashion designated by the user. |
| Retrospective Disposition | uint8 | Disposition of the file if the disposition is updated. If the disposition is not updated, this field contains the same value as the Disposition field. The possible values are the same as the Disposition field. |
| String Block Type | uint32 | Initiates a String data block containing the URI. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the URI data block, including eight bytes for the block type and header fields plus the number of bytes in the URI field. |
| URI | string | URI of the connection. |
| Source Port | uint16 | Port number for the source of the connection. |
| Destination Port | uint16 | Port number for the destination of the connection. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint 16 | Code for the country of the destination host. |

*Table B-17 Malware Event Data Block for 5.3.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| Action | uint8 | The action taken on the file based on the file type. Can have the following values:<br>• 1 — Detect<br>• 2 — Block<br>• 3 — Malware Cloud Lookup<br>• 4 — Malware Block<br>• 5 — Malware Allow List |
| Protocol | uint8 | IANA protocol number specified by the user. For example:<br>• 1 — ICMP<br>• 4 — IP<br>• 6 — TCP<br>• 17 — UDP<br>This is currently only TCP. |
| Threat Score | uint8 | A numeric value from 0 to 100 based on the potentially malicious behaviors observed during dynamic analysis. |
| IOC Number | uint16 | ID number of the compromise associated with this event. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |

# Malware Event Data Block 5.4.x

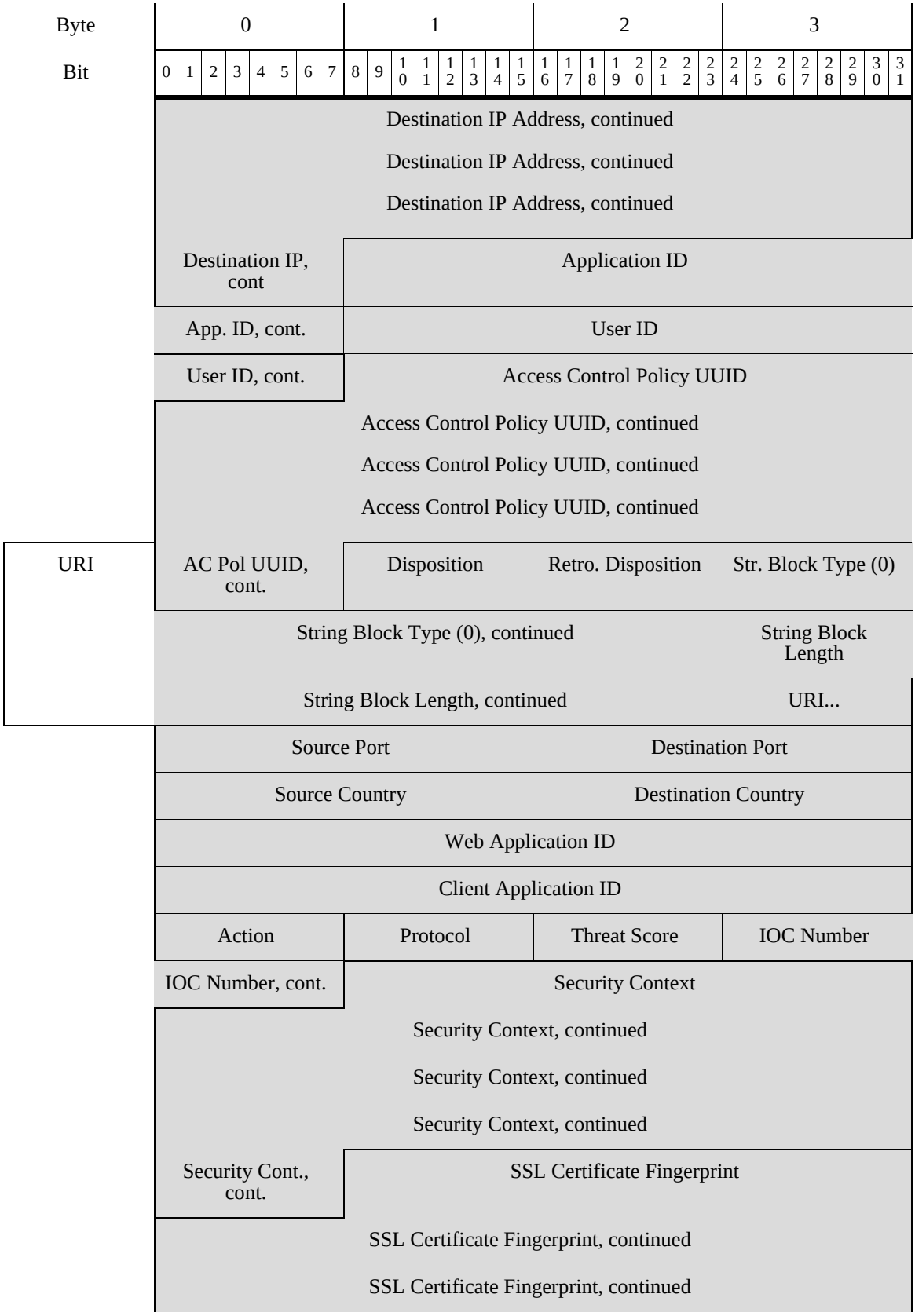
The eStreamer service uses the malware event data block to store information on malware events. These events contain information on malware detected or quarantined within a cloud, the detection method, and hosts and users affected by the malware. The malware event data block has a block type of 47 in the series 2 group of blocks. It supersedes block 44 and is superseded by block . Fields for SSL and file archive support have been added.

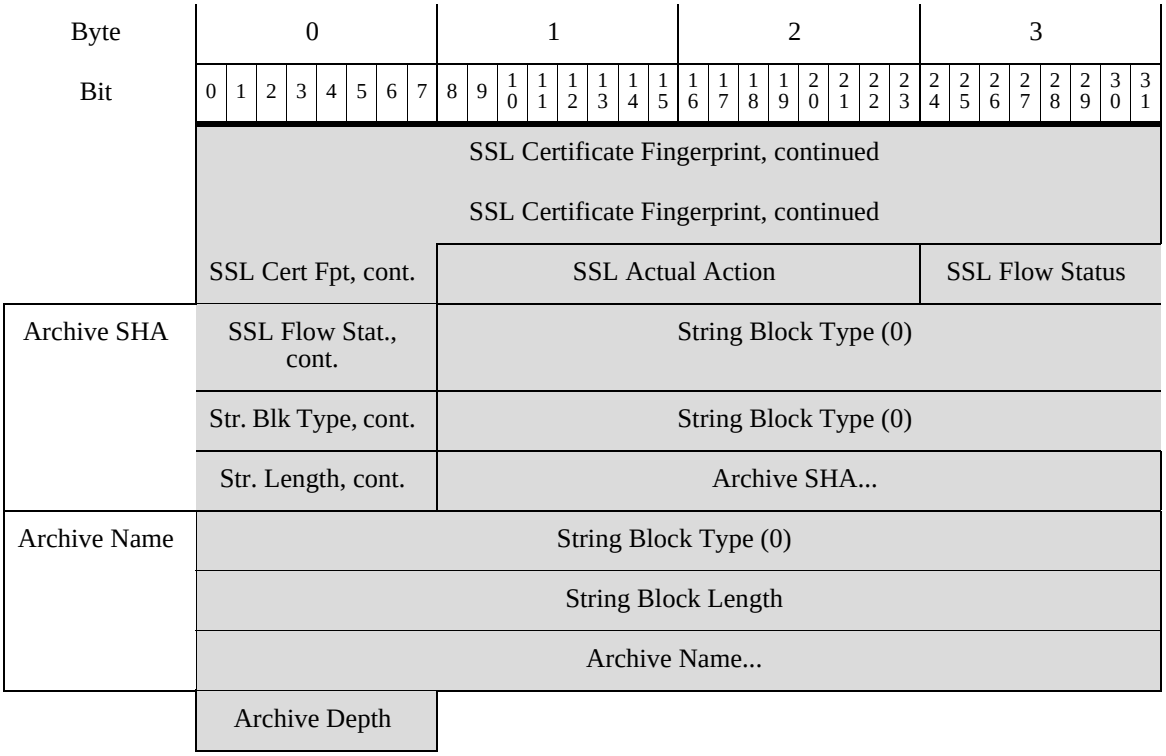
You request the event as part of the malware event record by setting the malware event flag—bit 30 in the request flags field—in the request message with an event version of 6 and an event code of 101.

The following graphic shows the structure of the malware event data block:

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Malware Event Block Type (47) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Malware Event Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Agent UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Agent UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Agent UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Agent UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Cloud UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Malware Event Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Type ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Subtype ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Detection Name | Detector ID | | | | | | | | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Type (0), cont. | | | | | | | | String Block Length | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length, cont. | | | | | | | | Detection Name... | | | | | | | | | | | | | | | | | | | | | | | |
| User | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | User... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
File Path
String Block Type (0)
String Block Length
File Path...
File SHA Hash
String Block Type (0)
String Block Length
File SHA Hash...
File Size
File Type
File Timestamp
Parent File Name
String Block Type (0)
String Block Length
Parent File Name...
Parent File SHA Hash
String Block Type (0)
String Block Length
Parent File SHA Hash...
Event Description
String Block Type (0)
String Block Length
Event Description...
Device ID
Connection Instance
Connection Counter
Connection Event Timestamp
Direction
Source IP Address
Source IP Address, continued
Source IP Address, continued
Source IP Address, continued
Source IP, cont.
Destination IP Address


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Destination IP Address, continued
Destination IP Address, continued
Destination IP Address, continued
Destination IP, cont
Application ID
App. ID, cont.
User ID
User ID, cont.
Access Control Policy UUID
Access Control Policy UUID, continued
Access Control Policy UUID, continued
Access Control Policy UUID, continued
URI
AC Pol UUID, cont.
Disposition
Retro. Disposition
Str. Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
URI...
Source Port
Destination Port
Source Country
Destination Country
Web Application ID
Client Application ID
Action
Protocol
Threat Score
IOC Number
IOC Number, cont.
Security Context
Security Context, continued
Security Context, continued
Security Context, continued
Security Cont., cont.
SSL Certificate Fingerprint
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued




The following table describes the fields in the malware event data block.

***Table B-18 Malware Event Data Block for 5.4.x Fields***

| Field | Data Type | Description |
|---|---|---|
| Malware Event Block Type | uint32 | Initiates a malware event data block. This value is always 47. |
| Malware Event Block Length | uint32 | Total number of bytes in the malware event data block, including eight bytes for the malware event block type and length fields, plus the number of bytes of data that follows. |
| Agent UUID | uint8[16] | The internal unique ID of the AMP for Endpoints agent reporting the malware event. |
| Cloud UUID | uint8[16] | The internal unique ID of the Cisco Advanced Malware Protection cloud from which the malware event originated. |
| Malware Event Timestamp | uint32 | The malware event generation timestamp. |
| Event Type ID | uint32 | The internal ID of the malware event type. |
| Event Subtype ID | uint32 | The internal ID of the action that led to malware detection. |
| Detector ID | uint8 | The internal ID of the detection technology that detected the malware. |
| String Block Type | uint32 | Initiates a String data block containing the detection name. This value is always 0. |

*Table B-18 Malware Event Data Block for 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the Detection Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Detection Name field. |
| Detection Name | string | The name of the detected or quarantined malware. |
| String Block Type | uint32 | Initiates a String data block containing the username. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the User String data block, including eight bytes for the block type and header fields plus the number of bytes in the User field. |
| User | string | The user of the computer where the Cisco Agent is installed and where the malware event occurred. Note that these users are not tied to user discovery. |
| String Block Type | uint32 | Initiates a String data block containing the file name. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Name field. |
| File Name | string | The name of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file path. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Path String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Path field. |
| File Path | string | The file path, not including the file name, of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file SHA hash. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the File SHA Hash field. |
| File SHA Hash | string | The rendered string of the SHA-256 hash value of the detected or quarantined file. |
| File Size | uint32 | The size in bytes of the detected or quarantined file. |
| File Type | uint8 | The file type of the detected or quarantined file. The meaning of this field is transmitted in the metadata with this event. See AMP for Endpoints File Type Metadata, page 3-38 for more information. |
| File Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the creation of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the parent file name. This value is always 0. |

*Table B-18 Malware Event Data Block for 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the Parent File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File Name field. |
| Parent File Name | string | The name of the file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the parent file SHA hash. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Parent File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File SHA Hash field. |
| Parent File SHA Hash | string | The SHA-256 hash value of the parent file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the event description. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Event Description String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Description field. |
| Event Description | string | The additional event information associated with the event type. |
| Device ID | uint32 | ID for the device that generated the event. |
| Connection Instance | uint16 | Snort instance on the device that generated the event. Used to link the event with a connection or IDS event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Connection Event Timestamp | uint32 | Timestamp of the connection event. |
| Direction | uint8 | Indicates whether the file was uploaded or downloaded. Can have the following values:<br>• `1` — Download<br>• `2` — Upload<br>Currently the value depends on the protocol (for example, if the connection is HTTP it is a download). |
| Source IP Address | uint8[16] | IPv4 or IPv6 address for the source of the connection. |
| Destination IP Address | uint8[16] | IPv4 or IPv6 address for the destination of the connection. |
| Application ID | uint32 | ID number that maps to the application using the file transfer. |
| User ID | uint32 | Identification number for the user logged into the destination host, as identified by the system. |

*Table B-18 Malware Event Data Block for 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Access Control Policy UUID | uint8[16] | Identification number that acts as a unique identifier for the access control policy that triggered the event. |
| Disposition | uint8 | The malware status of the file. Possible values include:<br>• 1 — CLEAN The file is clean and does not contain malware.<br>• 2 — UNKNOWN It is unknown whether the file contains malware.<br>• 3 — MALWARE The file contains malware.<br>• 4 — UNAVAILABLE The software was unable to send a request to the Cisco cloud for a disposition, or the Cisco cloud services did not respond to the request.<br>• 5 — CUSTOM SIGNATURE The file matches a user-defined hash, and is treated in a fashion designated by the user. |
| Retrospective Disposition | uint8 | Disposition of the file if the disposition is updated. If the disposition is not updated, this field contains the same value as the Disposition field. The possible values are the same as the Disposition field. |
| String Block Type | uint32 | Initiates a String data block containing the URI. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the URI data block, including eight bytes for the block type and header fields plus the number of bytes in the URI field. |
| URI | string | URI of the connection. |
| Source Port | uint16 | Port number for the source of the connection. |
| Destination Port | uint16 | Port number for the destination of the connection. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint 16 | Code for the country of the destination host. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |

*Table B-18 Malware Event Data Block for 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Action | uint8 | The action taken on the file based on the file type. Can have the following values:<br>• 1 — Detect<br>• 2 — Block<br>• 3 — Malware Cloud Lookup<br>• 4 — Malware Block<br>• 5 — Malware Allow List<br>• 6 — Cloud Lookup Timeout<br>• 7 — Custom Detection<br>• 8 — Custom Detection Block<br>• 9 — Archive Block (Depth Exceeded)<br>• 10 — Archive Block (Encrypted)<br>• 11 — Archive Block (Failed to Inspect) |
| Protocol | uint8 | IANA protocol number specified by the user. For example:<br>• 1 — ICMP<br>• 4 — IP<br>• 6 — TCP<br>• 17 — UDP<br>This is currently only TCP. |
| Threat Score | uint8 | A numeric value from 0 to 100 based on the potentially malicious behaviors observed during dynamic analysis. |
| IOC Number | uint16 | ID number of the compromise associated with this event. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |
| SSL Certificate Fingerprint | uint8[20] | SHA1 hash of the SSL Server certificate. |

*Table B-18 Malware Event Data Block for 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Actual Action | uint16 | The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |

*Table B-18 Malware Event Data Block for 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Status | uint16 | Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'No Match'<br>• 2 — 'Success'<br>• 3 — 'Uncached Session'<br>• 4 — 'Unknown Cipher Suite'<br>• 5 — 'Unsupported Cipher Suite'<br>• 6 — 'Unsupported SSL Version'<br>• 7 — 'SSL Compression Used'<br>• 8 — 'Session Undecryptable in Passive Mode'<br>• 9 — 'Handshake Error'<br>• 10 — 'Decryption Error'<br>• 11 — 'Pending Server Name Category Lookup'<br>• 12 — 'Pending Common Name Category Lookup'<br>• 13 — 'Internal Error'<br>• 14 — 'Network Parameters Unavailable'<br>• 15 — 'Invalid Server Certificate Handle'<br>• 16 — 'Server Certificate Fingerprint Unavailable'<br>• 17 — 'Cannot Cache Subject DN'<br>• 18 — 'Cannot Cache Issuer DN'<br>• 19 — 'Unknown SSL Version'<br>• 20 — 'External Certificate List Unavailable'<br>• 21 — 'External Certificate Fingerprint Unavailable'<br>• 22 — 'Internal Certificate List Invalid'<br>• 23 — 'Internal Certificate List Unavailable'<br>• 24 — 'Internal Certificate Unavailable'<br>• 25 — 'Internal Certificate Fingerprint Unavailable'<br>• 26 — 'Server Certificate Validation Unavailable'<br>• 27 — 'Server Certificate Validation Failure'<br>• 28 — 'Invalid Action' |
| String Block Type | uint32 | Initiates a String data block containing the Archive SHA. This value is always 0. |

*Table B-18 Malware Event Data Block for 5.4.x Fields (continued)*

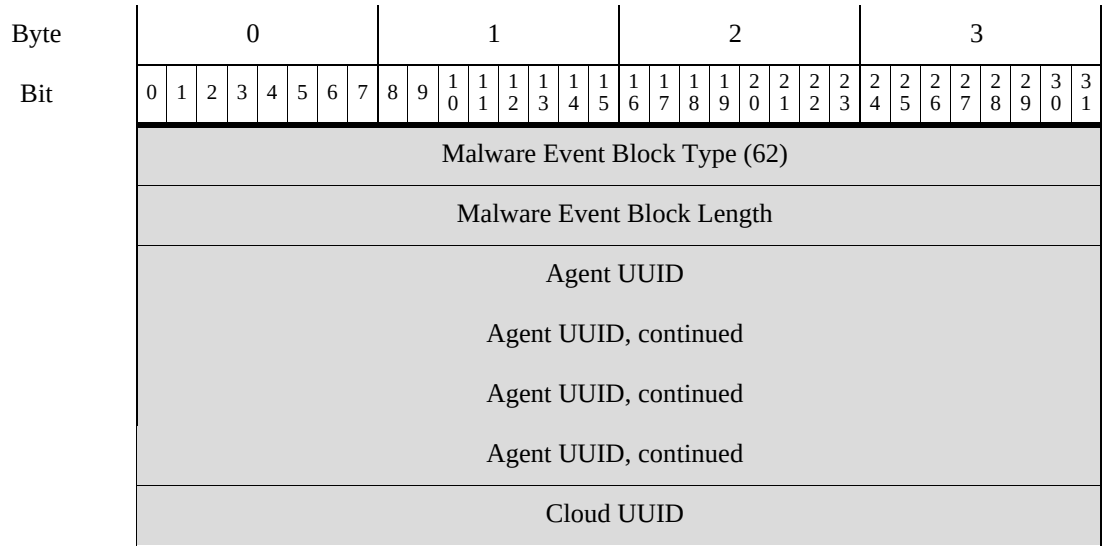
| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the Archive SHA String data block, including eight bytes for the block type and header fields plus the number of bytes in the intrusion policy name. |
| Archive SHA | string | SHA1 hash of the parent archive in which the file is contained. |
| String Block Type | uint32 | Initiates a String data block containing the Archive Name. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Archive Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the intrusion policy name. |
| Archive Name | string | Name of the parent archive. |
| Archive Depth | uint8 | Number of layers in which the file is nested. For example, if a text file is in a zip archive, this has a value of `1`. |

# Malware Event Data Block 6.x

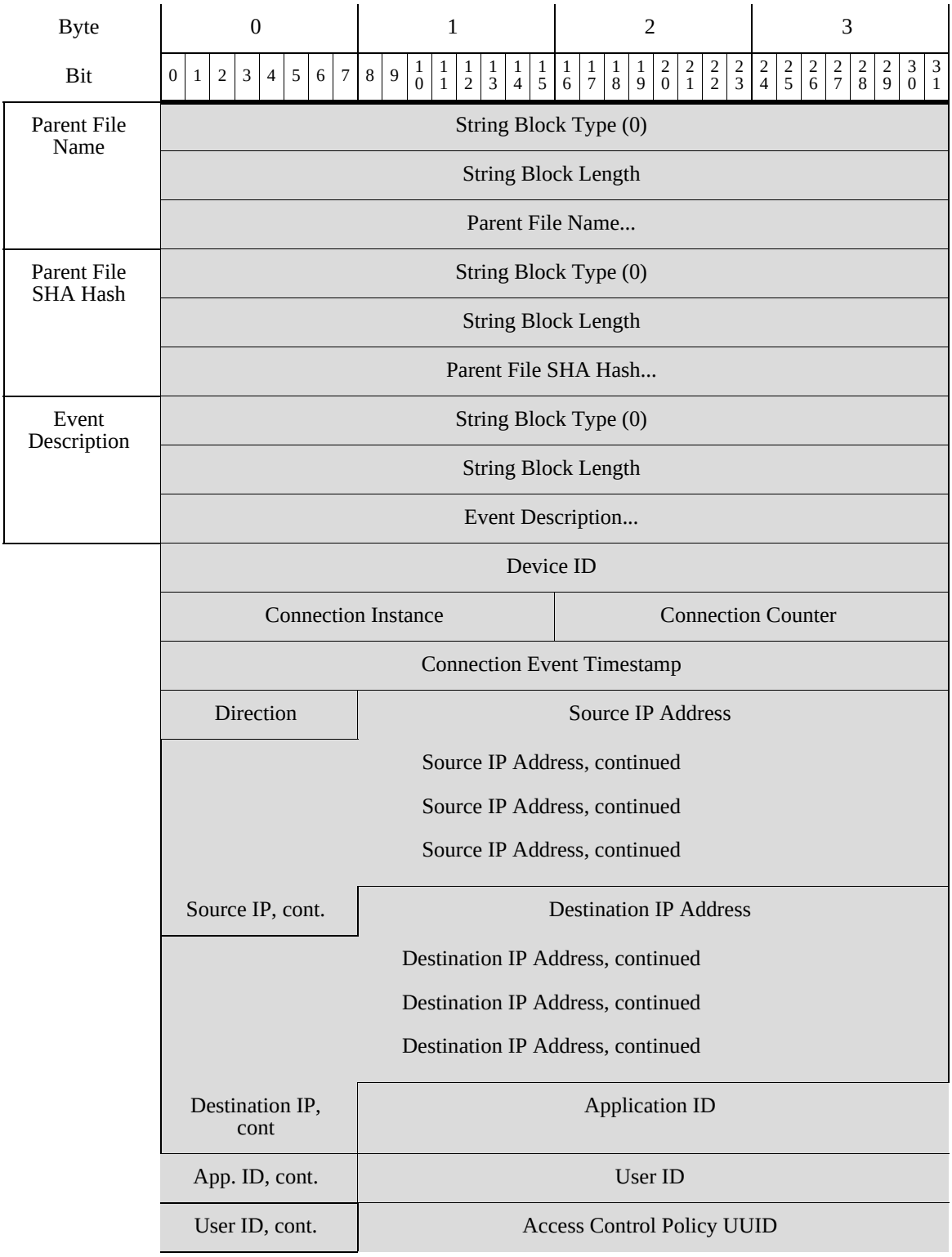
The eStreamer service uses the malware event data block to store information on malware events. These events contain information on malware detected or quarantined within a cloud, the detection method, and hosts and users affected by the malware. The malware event data block has a block type of 62 in the series 2 group of blocks. It supersedes block 47. A field for HTTP response has been added. It is superseded by block 80.

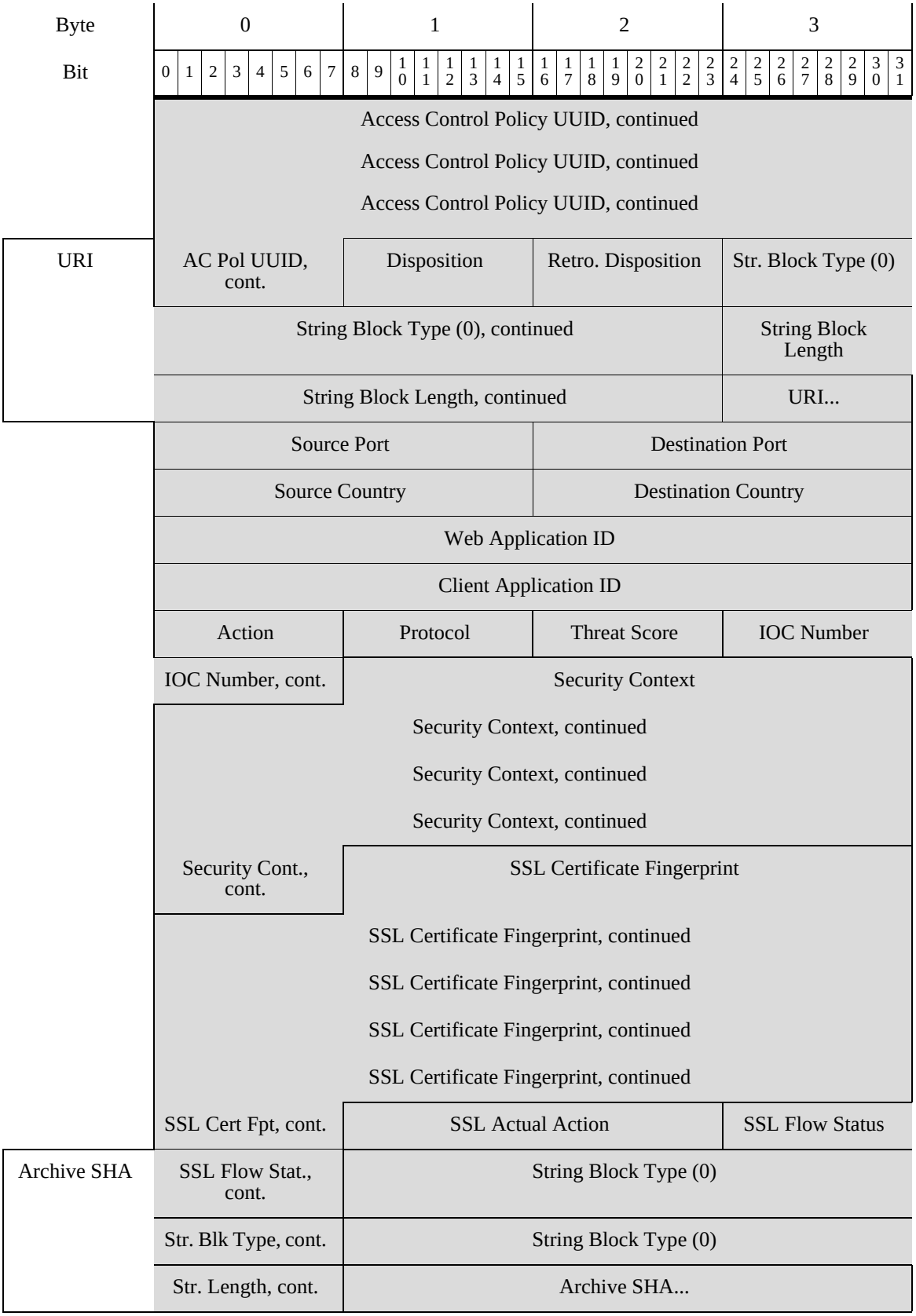
You request the event as part of the malware event record by setting the malware event flag—bit 30 in the request flags field—in the request message with an event version of 7 and an event code of 101.

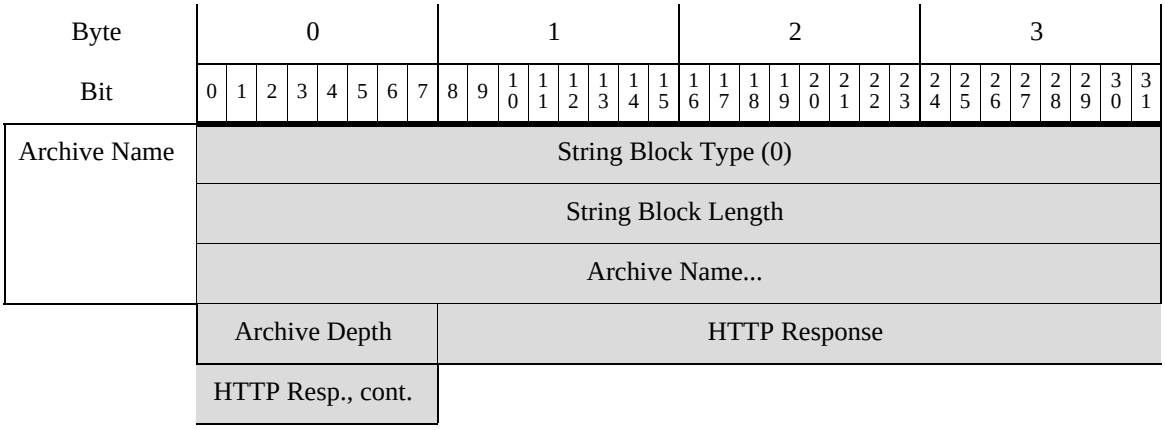
The following graphic shows the structure of the malware event data block.




Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Cloud UUID, continued
Cloud UUID, continued
Cloud UUID, continued
Malware Event Timestamp
Event Type ID
Event Subtype ID
Detection Name
Detector ID
String Block Type (0)
String Block Type (0), cont.
String Block Length
String Block Length, cont.
Detection Name...
User
String Block Type (0)
String Block Length
User...
File Name
String Block Type (0)
String Block Length
File Name...
File Path
String Block Type (0)
String Block Length
File Path...
File SHA Hash
String Block Type (0)
String Block Length
File SHA Hash...
File Size
File Type
File Timestamp


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Parent File Name
String Block Type (0)
String Block Length
Parent File Name...
Parent File SHA Hash
String Block Type (0)
String Block Length
Parent File SHA Hash...
Event Description
String Block Type (0)
String Block Length
Event Description...
Device ID
Connection Instance
Connection Counter
Connection Event Timestamp
Direction
Source IP Address
Source IP Address, continued
Source IP Address, continued
Source IP Address, continued
Source IP, cont.
Destination IP Address
Destination IP Address, continued
Destination IP Address, continued
Destination IP Address, continued
Destination IP, cont
Application ID
App. ID, cont.
User ID
User ID, cont.
Access Control Policy UUID


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Access Control Policy UUID, continued
Access Control Policy UUID, continued
Access Control Policy UUID, continued
URI
AC Pol UUID, cont.
Disposition
Retro. Disposition
Str. Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
URI...
Source Port
Destination Port
Source Country
Destination Country
Web Application ID
Client Application ID
Action
Protocol
Threat Score
IOC Number
IOC Number, cont.
Security Context
Security Context, continued
Security Context, continued
Security Context, continued
Security Cont., cont.
SSL Certificate Fingerprint
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Cert Fpt, cont.
SSL Actual Action
SSL Flow Status
Archive SHA
SSL Flow Stat., cont.
String Block Type (0)
Str. Blk Type, cont.
String Block Type (0)
Str. Length, cont.
Archive SHA...




The following table describes the fields in the malware event data block.

***Table B-19 Malware Event Data Block for 6.x Fields***

| Field | Data Type | Description |
|---|---|---|
| Malware Event Block Type | uint32 | Initiates a malware event data block. This value is always 62. |
| Malware Event Block Length | uint32 | Total number of bytes in the malware event data block, including eight bytes for the malware event block type and length fields, plus the number of bytes of data that follows. |
| Agent UUID | uint8[16] | The internal unique ID of the AMP for Endpoints agent reporting the malware event. |
| Cloud UUID | uint8[16] | The internal unique ID of the AMP cloud from which the malware event originated. |
| Malware Event Timestamp | uint32 | The malware event generation timestamp. |
| Event Type ID | uint32 | The internal ID of the malware event type. |
| Event Subtype ID | uint32 | The internal ID of the action that led to malware detection. |
| Detector ID | uint8 | The internal ID of the detection technology that detected the malware. |
| String Block Type | uint32 | Initiates a String data block containing the detection name. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Detection Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Detection Name field. |
| Detection Name | string | The name of the detected or quarantined malware. |
| String Block Type | uint32 | Initiates a String data block containing the username. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the User String data block, including eight bytes for the block type and header fields plus the number of bytes in the User field. |

*Table B-19 Malware Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| User | string | The user of the computer where the Cisco Agent is installed and where the malware event occurred. Note that these users are not tied to user discovery. |
| String Block Type | uint32 | Initiates a String data block containing the file name. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Name field. |
| File Name | string | The name of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file path. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File Path String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Path field. |
| File Path | string | The file path, not including the file name, of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the file SHA hash. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the File SHA Hash field. |
| File SHA Hash | string | The rendered string of the SHA-256 hash value of the detected or quarantined file. |
| File Size | uint32 | The size in bytes of the detected or quarantined file. |
| File Type | uint32 | The file type of the detected or quarantined file. The meaning of this field is transmitted in the metadata with this event. See AMP for Endpoints File Type Metadata, page 3-38 for more information. |
| File Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the creation of the detected or quarantined file. |
| String Block Type | uint32 | Initiates a String data block containing the parent file name. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the Parent File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File Name field. |
| Parent File Name | string | The name of the file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the parent file SHA hash. This value is always 0. |

*Table B-19 Malware Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the Parent File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File SHA Hash field. |
| Parent File SHA Hash | string | The SHA-256 hash value of the parent file accessing the detected or quarantined file when detection occurred. |
| String Block Type | uint32 | Initiates a String data block containing the event description. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Event Description String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Description field. |
| Event Description | string | The additional event information associated with the event type. |
| Device ID | uint32 | ID for the device that generated the event. |
| Connection Instance | uint16 | Snort instance on the device that generated the event. Used to link the event with a connection or IDS event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Connection Event Timestamp | uint32 | Timestamp of the connection event. |
| Direction | uint8 | Indicates whether the file was uploaded or downloaded. Can have the following values:<br>• `1` — Download<br>• `2` — Upload<br>Currently the value depends on the protocol (for example, if the connection is HTTP it is a download). |
| Source IP Address | uint8[16] | IPv4 or IPv6 address for the source of the connection. |
| Destination IP Address | uint8[16] | IPv4 or IPv6 address for the destination of the connection. |
| Application ID | uint32 | ID number that maps to the application using the file transfer. |
| User ID | uint32 | Identification number for the user logged into the destination host, as identified by the system. |
| Access Control Policy UUID | uint8[16] | Identification number that acts as a unique identifier for the access control policy that triggered the event. |

*Table B-19 Malware Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Disposition | uint8 | The malware status of the file. Possible values include:<br>• 1 — CLEAN The file is clean and does not contain malware.<br>• 2 — UNKNOWN It is unknown whether the file contains malware.<br>• 3 — MALWARE The file contains malware.<br>• 4 — UNAVAILABLE The software was unable to send a request to the AMP cloud for a disposition, or the AMP cloud services did not respond to the request.<br>• 5 — CUSTOM SIGNATURE The file matches a user-defined hash, and is treated in a fashion designated by the user. |
| Retrospective Disposition | uint8 | Disposition of the file if the disposition is updated. If the disposition is not updated, this field contains the same value as the Disposition field. The possible values are the same as the Disposition field. |
| String Block Type | uint32 | Initiates a String data block containing the URI. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the URI data block, including eight bytes for the block type and header fields plus the number of bytes in the URI field. |
| URI | string | URI of the connection. |
| Source Port | uint16 | Port number for the source of the connection. |
| Destination Port | uint16 | Port number for the destination of the connection. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint 16 | Code for the country of the destination host. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |

*Table B-19 Malware Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Action | uint8 | The action taken on the file based on the file type. Can have the following values:<br>• 1 — Detect<br>• 2 — Block<br>• 3 — Malware Cloud Lookup<br>• 4 — Malware Block<br>• 5 — Malware Allow List<br>• 6 — Cloud Lookup Timeout<br>• 7 — Custom Detection<br>• 8 — Custom Detection Block<br>• 9 — Archive Block (Depth Exceeded)<br>• 10 — Archive Block (Encrypted)<br>• 11 — Archive Block (Failed to Inspect) |
| Protocol | uint8 | IANA protocol number specified by the user. For example:<br>• 1 — ICMP<br>• 4 — IP<br>• 6 — TCP<br>• 17 — UDP<br>This is currently only TCP. |
| Threat Score | uint8 | A numeric value from 0 to 100 based on the potentially malicious behaviors observed during dynamic analysis. |
| IOC Number | uint16 | ID number of the compromise associated with this event. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |
| SSL Certificate Fingerprint | uint8[20] | SHA1 hash of the SSL Server certificate. |

*Table B-19 Malware Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Actual Action | uint16 | The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |

*Table B-19 Malware Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Status | uint16 | Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'No Match'<br>• 2 — 'Success'<br>• 3 — 'Uncached Session'<br>• 4 — 'Unknown Cipher Suite'<br>• 5 — 'Unsupported Cipher Suite'<br>• 6 — 'Unsupported SSL Version'<br>• 7 — 'SSL Compression Used'<br>• 8 — 'Session Undecryptable in Passive Mode'<br>• 9 — 'Handshake Error'<br>• 10 — 'Decryption Error'<br>• 11 — 'Pending Server Name Category Lookup'<br>• 12 — 'Pending Common Name Category Lookup'<br>• 13 — 'Internal Error'<br>• 14 — 'Network Parameters Unavailable'<br>• 15 — 'Invalid Server Certificate Handle'<br>• 16 — 'Server Certificate Fingerprint Unavailable'<br>• 17 — 'Cannot Cache Subject DN'<br>• 18 — 'Cannot Cache Issuer DN'<br>• 19 — 'Unknown SSL Version'<br>• 20 — 'External Certificate List Unavailable'<br>• 21 — 'External Certificate Fingerprint Unavailable'<br>• 22 — 'Internal Certificate List Invalid'<br>• 23 — 'Internal Certificate List Unavailable'<br>• 24 — 'Internal Certificate Unavailable'<br>• 25 — 'Internal Certificate Fingerprint Unavailable'<br>• 26 — 'Server Certificate Validation Unavailable'<br>• 27 — 'Server Certificate Validation Failure'<br>• 28 — 'Invalid Action' |
| String Block Type | uint32 | Initiates a String data block containing the Archive SHA. This value is always 0. |

*Table B-19 Malware Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the Archive SHA String data block, including eight bytes for the block type and header fields plus the number of bytes in the intrusion policy name. |
| Archive SHA | string | SHA1 hash of the parent archive in which the file is contained. |
| String Block Type | uint32 | Initiates a String data block containing the Archive Name. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Archive Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the intrusion policy name. |
| Archive Name | string | Name of the parent archive. |
| Archive Depth | uint8 | Number of layers in which the file is nested. For example, if a text file is in a zip archive, this has a value of `1`. |
| HTTP Response | uint32 | Response code of the HTTP Request. |

# Legacy Discovery Data Structures


- Legacy Discovery Event Header, page B-121
- Legacy Server Data Blocks, page B-123
- Legacy Client Application Data Blocks, page B-124
- Legacy Scan Result Data Blocks, page B-125
- Legacy Host Profile Data Blocks, page B-150
- Legacy OS Fingerprint Data Blocks, page B-157


# Legacy Discovery Event Header

## Discovery Event Header 5.0 - 5.1.1.x

Discovery and connection event messages contain a discovery event header. It conveys the type and subtype of the event, the time the event occurred, the device on which the event occurred, and the structure of the event data in the message. This header is followed by the actual host discovery, user, or connection event data. The structures associated with the different event type/subtype values are described in Host Discovery Structures by Event Type, page 4-44.

The event type and event subtype fields of the discovery event header identify the structure of the transmitted event message. Once the structure of the event data block is determined, your program can parse the message appropriately.

The shaded rows in the following diagram illustrate the format of the discovery event header.

| | Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Header Version (1) | | | | | | | | | | | | | | | | | Message Type (4) | | | | | | | | | | | | | | | |
| | Message Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Netmap ID | | | | | | | | | | | | | | | | | Record Type | | | | | | | | | | | | | | | |
| | Record Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | eStreamer Server Timestamp (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Reserved for Future Use (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Discovery Event Header | Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | MAC Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | MAC Address, continued | | | | | | | | | | | | | | | | | Reserved for future use | | | | | | | | | | | | | | | |
| | Event Second | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Microsecond | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Reserved (Internal) | | | | | | | | | Event Type | | | | | | | | | | | | | | | | | | | | | | | |
| | Event Subtype | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Number (Internal Use Only) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Position (Internal Use Only) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

The following table describes the discovery event header.

***Table B-20 Discovery Event Header Fields***

| Field | Data Types | Description |
|---|---|---|
| Device ID | uint32 | ID number of the device that generated the discovery event. You can obtain the metadata for the device by requesting Version 3 and 4 metadata. See Managed Device Record Metadata, page 3-33 for more information. |
| IP Address | uint32 | IP address of the host involved in the event. |
| MAC Address | uint8[6] | MAC address of the host involved in the event. |
| Reserved for future use | byte[2] | Two bytes of padding with values set to `0`. |
| Event Second | uint32 | UNIX timestamp (seconds since 01/01/1970) that the system generated the event. |

*Table B-20 Discovery Event Header Fields (continued)*

| Field | Data Types | Description |
|---|---|---|
| Event Microsecond | uint32 | Microsecond (one millionth of a second) increment that the system generated the event. |
| Reserved (Internal) | byte | Internal data from Cisco and can be disregarded. |
| Event Type | uint32 | Event type (`1000` for new events, `1001` for change events, `1002` for user input events, 1050 for full host profile). See Host Discovery Structures by Event Type, page 4-44 for a list of available event types. |
| Event Subtype | uint32 | Event subtype. See Host Discovery Structures by Event Type, page 4-44 for a list of available event subtypes. |
| File Number | byte[4] | Serial file number. This field is for Cisco internal use and can be disregarded. |
| File Position | byte[4] | Event's position in the serial file. This field is for Cisco internal use and can be disregarded. |

# Legacy Server Data Blocks

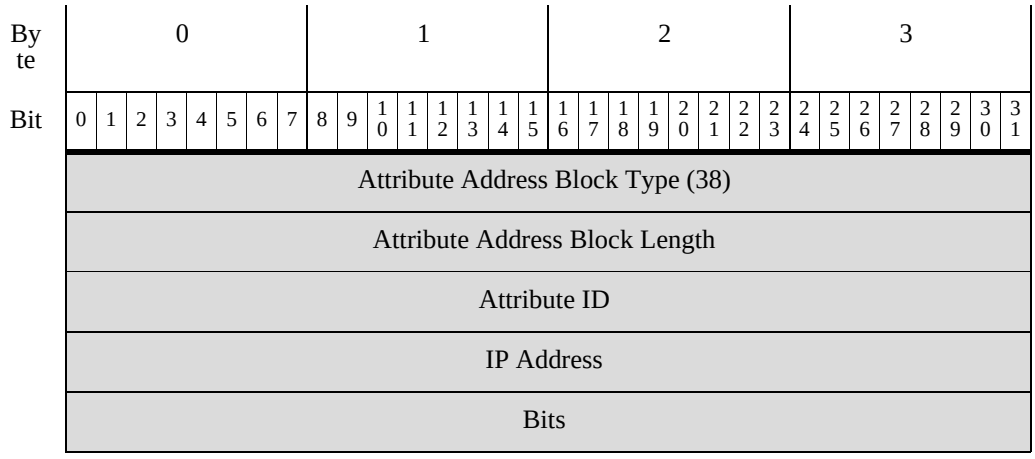
For more information, see the following sections:

- Attribute Address Data Block for 5.0 - 5.1.1.x, page B-123

# Attribute Address Data Block for 5.0 - 5.1.1.x

The Attribute Address data block contains an attribute list item and is used within an Attribute Definition data block. It has a block type of 38.

The following diagram shows the basic structure of an Attribute Address data block:




The following table describes the fields of the Attribute Address data block.

*Table B-21 Attribute Address Data Block Fields*

| Field | Data Type | Description |
|---|---|---|
| Attribute Address Block Type | uint32 | Initiates an Attribute Address data block. This value is always 38. |
| Attribute Address Block Length | uint32 | Number of bytes in the Attribute Address data block, including eight bytes for the attribute address block type and length, plus the number of bytes in the attribute address data that follows. |
| Attribute ID | uint32 | Identification number of the affected attribute, if applicable. |
| IP Address | uint8[4] | IP address of the host, if the address was automatically assigned, in IP address octets. |
| Bits | uint32 | Contains the significant bits used to calculate the netmask if an IP address was automatically assigned. |

# Legacy Client Application Data Blocks

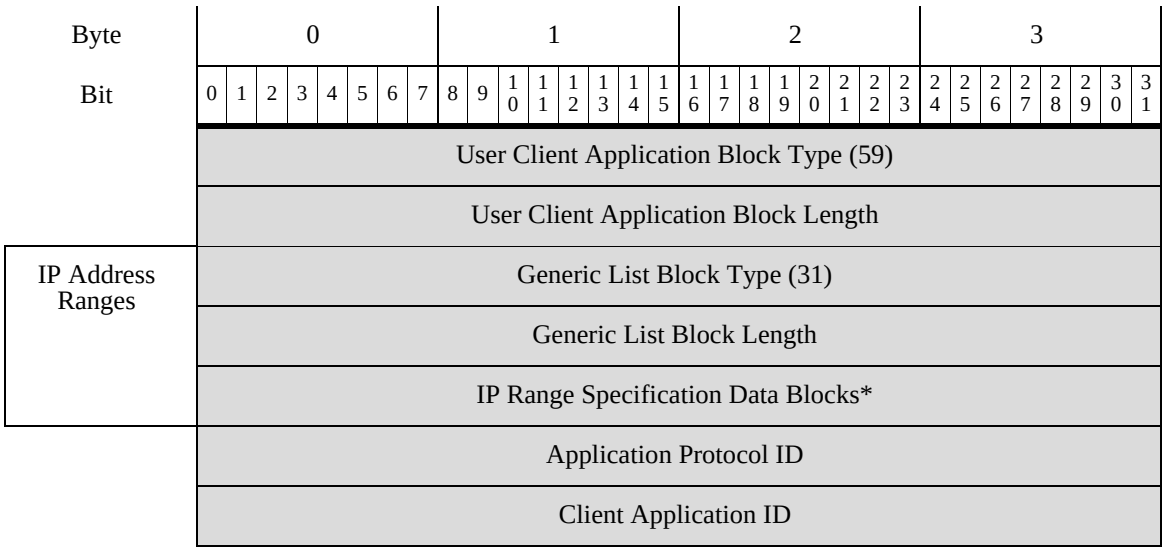
For more information, see the following sections:

- User Client Application Data Block for 5.0 - 5.1, page B-124

## User Client Application Data Block for 5.0 - 5.1

The User Client Application data block contains information about the source of the client application data, the identification number for the user who added the data, and the lists of IP address range data blocks. The User Client Application data block has a block type of 59.

The following diagram shows the basic structure of a User Client Application data block:



| | |
|---|---|
| Version | String Block Type (0) |
| | String Block Length |
| | Version... |

The following table describes the fields of the User Client Application data block.

***Table B-22 User Client Application Data Block Fields***

| Field | Number of Bytes | Description |
|---|---|---|
| User Client Application Block Type | uint32 | Initiates a User Client Application data block. This value is always . |
| User Client Application Block Length | uint32 | Total number of bytes in the User Client Application data block, including eight bytes for the user client application block type and length fields, plus the number of bytes of user client application data that follows. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising IP Range Specification data blocks conveying IP address range data. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated IP Range Specification data blocks. |
| IP Range Specification Data Blocks * | variable | IP Range Specification data blocks containing information about the IP address ranges for the user input. See Table 4-59User Server Data Block Fields, page 4-103 for a description of this data block. |
| Application Protocol ID | uint32 | The internal identification number for the application protocol, if applicable. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| String Block Type | uint32 | Initiates a String data block that contains the client application version. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the client application version String data block, including the string block type and length fields, plus the number of bytes in the version. |
| Version | string | Client application version. |

# Legacy Scan Result Data Blocks

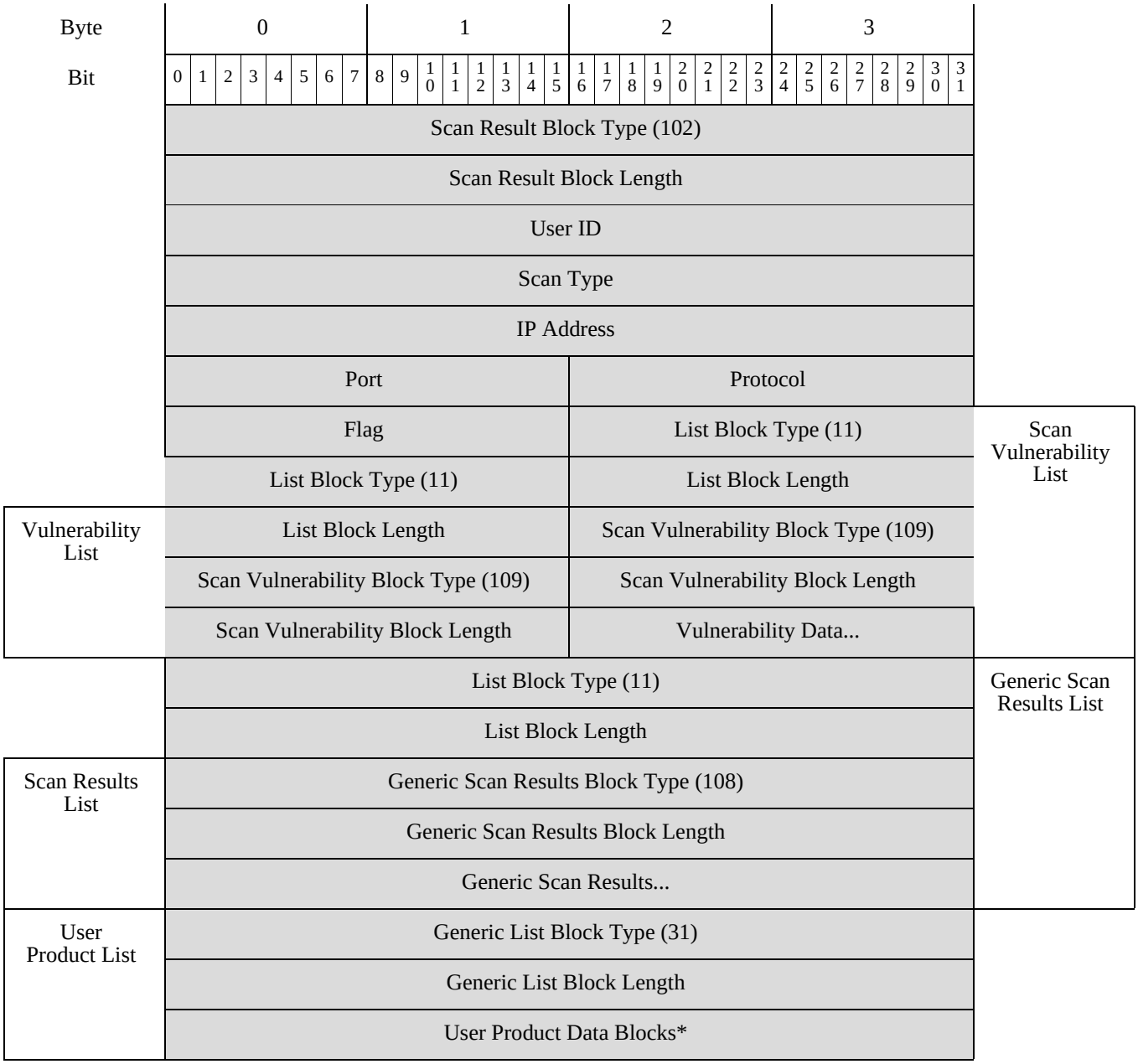
For more information, see the following sections:

- Scan Result Data Block 5.0 - 5.1.1.x, page B-126
- User Product Data Block for 5.0.x, page B-128
- User Information Data Block for 5.x, page B-148

## Scan Result Data Block 5.0 - 5.1.1.x

The Scan Result data block describes a vulnerability and is used within Add Scan Result events (event type 1002, subtype 11). The Scan Result data block has a block type of 102.

The following diagram shows the format of a Scan Result data block:




The following table describes the fields of the Scan Result data block.

*Table B-23 Scan Result Data Block Fields*

| Field | Data Type | Description |
|---|---|---|
| Scan Result Block Type | uint32 | Initiates a Scan Result data block. This value is always 102. |
| Scan Result Block Length | uint32 | Number of bytes in the Scan Vulnerability data block, including eight bytes for the scan vulnerability block type and length fields, plus the number of bytes of scan vulnerability data that follows. |
| User ID | uint32 | Contains the user identification number for the user who imported the scan result or ran the scan that produced the scan result. |
| Scan Type | uint32 | Indicates how the results were added to the system. |
| IP Address | uint32 | IP address of the host affected by the vulnerabilities in the result, in IP address octets. |
| Port | uint16 | Port used by the sub-server affected by the vulnerabilities in the results. |
| Protocol | uint16 | IANA protocol number. For example:<br>• 1 — ICMP<br>• 4 — IP<br>• 6 — TCP<br>• 17 — UDP |
| Flag | uint16 | Reserved |
| List Block Type | uint32 | Initiates a List data block comprising Scan Vulnerability data blocks conveying transport Scan Vulnerability data. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus all encapsulated Scan Vulnerability data blocks.<br>This field is followed by zero or more Scan Vulnerability data blocks. |
| Scan Vulnerability Block Type | uint32 | Initiates a Scan Vulnerability data block describing a vulnerability detected during a scan. This value is always 109. |
| Scan Vulnerability Block Length | uint32 | Number of bytes in the Scan Vulnerability data block, including eight bytes for the scan vulnerability block type and length fields, plus the number of bytes in the scan vulnerability data that follows. |
| Vulnerability Data | string | Information relating to each vulnerability. |
| List Block Type | uint32 | Initiates a List data block comprising Scan Vulnerability data blocks conveying transport Scan Vulnerability data. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus all encapsulated Scan Vulnerability data blocks.<br>This field is followed by zero or more Scan Vulnerability data blocks. |
| Generic Scan Results Block Type | uint32 | Initiates a Generic Scan Results data block describing server and operating system data detected during a scan. This value is always 108. |

*Table B-23 Scan Result Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Generic Scan Results Block Length | uint32 | Number of bytes in the Generic Scan Results data block, including eight bytes for the generic scan results block type and length fields, plus the number of bytes in the scan result data that follows. |
| Generic Scan Results Data | string | Information relating to each scan result. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising User Product data blocks conveying host input data from a third party application. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated User Product data blocks. |
| User Product Data Blocks * | variable | User Product data blocks containing host input data. See User Product Data Block 5.1+, page 4-171 for a description of this data block. |

## User Product Data Block for 5.0.x

The User Product data block conveys host input data imported from a third party application, including third party application string mappings. This data block is used in Connection Statistics Data Block 6.0.x, page B-224 and User Server and Operating System Messages, page 4-57. The User Product data block has a block type of 65 for 4.10.x, and a block type of 118 for 5.0 - 5.0.x. The block types have the same structure.

**Note** An asterisk(*) next to a data block name in the following diagram indicates that multiple instances of the data block may occur.

The following diagram shows the format of the User Product data block:

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | User Product Data Block Type (65 \| 118) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | User Product Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Source ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Source Type | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| IP Address Ranges | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | IP Range Specification Data Blocks* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Port | | | | | | | | | | | | | | | | Protocol | | | | | | | | | | | | | | | |
| | Drop User Product | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Custom Vendor String | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Custom Vendor String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Custom Product String | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Custom Product String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Custom Version String | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Custom Version String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Software ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Server ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Vendor ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Product ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Major Version String | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Major Version String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Minor Version String | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Minor Version String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Revision String | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Revision String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| To Major String | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | To Major Version String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| To Minor String | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | To Minor Version String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| To Revision String | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | To Revision String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Build String | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Build String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Patch String | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Patch String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Extension String | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Extension String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| OS UUID | Operating System UUID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System UUID cont. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System UUID cont. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System UUID cont. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| List of Fixes | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Fix List Data Blocks* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

The following table describes the components of the User Product data block.

*Table B-24 User Product Data Block Fields for 4.10.x, 5.0-5.0.x*

| Field | Data Type | Description |
|---|---|---|
| User Product Data Block Type | uint32 | Initiates a User Product data block. This value is `65` for version 4.10.x and `118` for version 5.0 - 5.0.x. |
| User Product Block Length | uint32 | Total number of bytes in the User Product data block, including eight bytes for the user product block type and length fields, plus the number of bytes in the user product data that follows. |
| Source ID | uint32 | Identification number of the source that imported the data. |
| Source Type | uint32 | The source type of the source that supplied the data. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising IP Range Specification data blocks conveying IP address range data. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated IP Range Specification data blocks. |
| IP Range Specification Data Blocks * | variable | IP Range Specification data blocks containing information about the IP address ranges for the user input. See IP Address Range Data Block for 5.2+, page 4-95 for a description of this data block. |
| Port | uint16 | Port specified by the user. |
| Protocol | uint16 | IANA protocol number specified by the user. For example:<br>• `1` — ICMP<br>• `4` — IP<br>• `6` — TCP<br>• `17` — UDP |
| Drop User Product | uint32 | Indicates whether the user OS definition was deleted from the host:<br>• `0` — No<br>• `1` — Yes |
| String Block Type | uint32 | Initiates a String data block containing the custom vendor name specified in the user input. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the custom vendor String data block, including eight bytes for the block type and length fields, plus the number of bytes in the vendor name. |
| Custom Vendor Name | string | The custom vendor name specified in the user input. |
| String Block Type | uint32 | Initiates a String data block containing the custom product name specified in the user input. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the custom product String data block, including eight bytes for the block type and length fields, plus the number of bytes in the product name. |
| Custom Product Name | string | The custom product name specified in the user input. |
| String Block Type | uint32 | Initiates a String data block containing the custom version specified in the user input. This value is always `0`. |

*Table B-24 User Product Data Block Fields for 4.10.x, 5.0-5.0.x (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | Number of bytes in the custom version String data block, including eight bytes for the block type and length fields, plus the number of bytes in the version. |
| Custom Version | string | The custom version specified in the user input. |
| Software ID | uint32 | The identifier for a specific revision of a server or operating system in the Cisco database. |
| Server ID | uint32 | The Cisco application identifier for the application protocol on the host server specified in user input. |
| Vendor ID | uint32 | The identifier for the vendor of a third party operating system specified when the third party operating system is mapped to a Cisco 3D operating system definition. |
| Product ID | uint32 | The product identification string of a third party operating system string specified when the third party operating system string is mapped to a Cisco 3D operating system definition. |
| String Block Type | uint32 | Initiates a String data block containing the major version number of the Cisco 3D operating system definition that a third party operating system string in the user input is mapped to. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the major String data block, including eight bytes for the block type and length fields, plus the number of bytes in the version. |
| Major Version | string | Major version of the Cisco 3D operating system definition that a third party operating system string is mapped to. |
| String Block Type | uint32 | Initiates a String data block containing the minor version number of the Cisco 3D operating system definition that a third party operating system string is mapped to. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the minor String data block, including eight bytes for the block type and length fields, plus the number of bytes in the version. |
| Minor Version | string | Minor version number of the Cisco 3D operating system definition that a third party operating system string in the user input is mapped to. |
| String Block Type | uint32 | Initiates a String data block containing the revision number of the Cisco operating system definition that a third party operating system string in the user input is mapped to. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the revision String data block, including eight bytes for the block type and length fields, plus the number of bytes in the revision number. |
| Revision | string | Revision number of the Cisco 3D operating system definition that a third party operating system string in the user input is mapped to. |
| String Block Type | uint32 | Initiates a String data block containing the last major version of the Cisco 3D operating system definition that a third party operating system string is mapped to. This value is always `0`. |

*Table B-24 User Product Data Block Fields for 4.10.x, 5.0-5.0.x (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | Number of bytes in the To Major String data block, including eight bytes for the block type and length fields, plus the number of bytes in the version. |
| To Major | string | Last version number in a range of major version numbers of the Cisco 3D operating system definition that a third party operating system string in the user input is mapped to. |
| String Block Type | uint32 | Initiates a String data block containing the last minor version of the Cisco 3D operating system definition that a third party operating system string is mapped to. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the To Minor String data block, including eight bytes for the block type and length fields, plus the number of bytes in the version. |
| To Minor | string | Last version number in a range of minor version numbers of the Cisco 3D operating system definition that a third party operating system string in the user input is mapped to. |
| String Block Type | uint32 | Initiates a String data block containing the Last revision number of the Cisco 3D operating system definition that a third party operating system string is mapped to. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the To Revision String data block, including eight bytes for the block type and length fields, plus the number of bytes in the revision number. |
| To Revision | string | Last revision number in a range of revision numbers of the Cisco 3D operating system definitions that a third party operating system string in the user input is mapped to. |
| String Block Type | uint32 | Initiates a String data block containing the build number of the Cisco 3D operating system that the third party operating system string is mapped. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the build String data block, including eight bytes for the block type and length fields, plus the number of bytes in the build number. |
| Build | string | Build number of the Cisco 3D operating system that the third party operating system string in the user input is mapped to. |
| String Block Type | uint32 | Initiates a String data block containing the patch number of the Cisco 3D operating system that the third party operating system string is mapped to. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the patch String data block, including eight bytes for the block type and length fields, plus the number of bytes in the patch number. |
| Patch | string | Patch number of the Cisco 3D operating system that the third party operating system string in the user input is mapped to. |
| String Block Type | uint32 | Initiates a String data block containing the extension number of the Cisco 3D operating system that the third party operating system string is mapped. This value is always 0. |

*Table B-24 User Product Data Block Fields for 4.10.x, 5.0-5.0.x (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | Number of bytes in the extension String data block, including eight bytes for the block type and length fields, plus the number of bytes in the extension number. |
| Extension | string | Extension number of the Cisco 3D operating system that the third party operating system string in the user input is mapped to. |
| UUID | uint8 [x16] | Contains the unique identification number for the operating system. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Fix List data blocks conveying user input data regarding what fixes have been applied to hosts in the specified IP address ranges. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Fix List data blocks. |
| Fix List Data Blocks * | variable | Fix List data blocks containing information about fixes applied to the hosts. See Fix List Data Block, page 4-102 for a description of this data block. |

# Legacy User Login Data Blocks

See the following sections for more information:

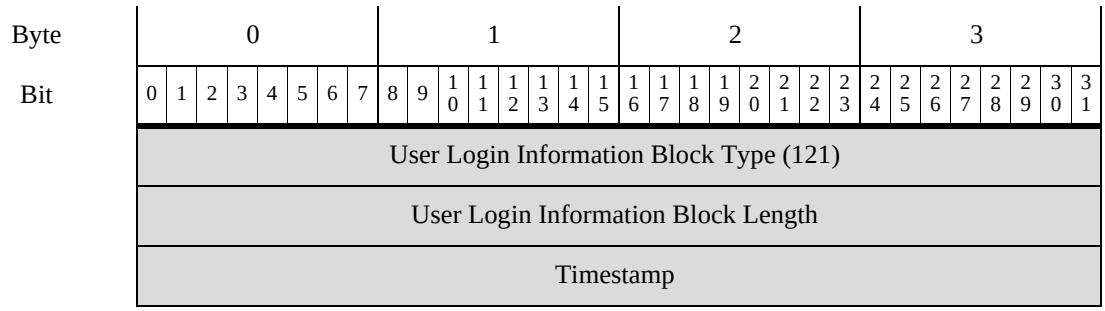
- User Login Information Data Block for 5.0 - 5.0.2, page B-134
- User Login Information Data Block 5.1-5.4.x, page B-136
- User Login Information Data Block 6.0.x, page B-138
- User Login Information Data Block 6.1.x, page B-141
- User Information Data Block for 5.x, page B-148

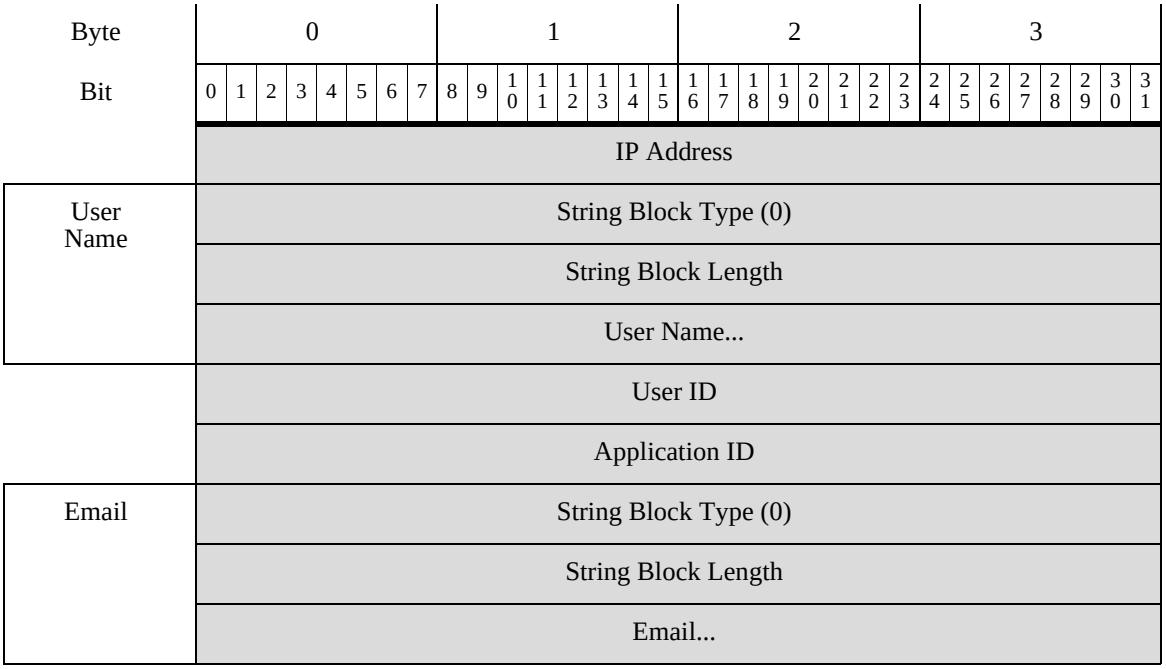
## User Login Information Data Block for 5.0 - 5.0.2

The User Login Information data block is used in User Information Update messages and conveys changes in login information for a detected user. For more information, see User Information Update Message Block, page 4-62.

The User Login Information data block has a block type of 121 for version 5.0 - 5.0.2.

The graphic below shows the format of the User Login Information data block:






The following table describes the components of the User Login Information data block.

***Table B-25 User Login Information Data Block Fields 5.0 - 5.0.2***

| Field | Data Type | Description |
|---|---|---|
| User Login Information Block Type | uint32 | Initiates a User Login Information data block. This value is `121` for version 5.0 - 5.0.2. |
| User Login Information Block Length | uint32 | Total number of bytes in the User Login Information data block, including eight bytes for the user login information block type and length fields, plus the number of bytes in the user login information data that follows. |
| Timestamp | uint32 | Timestamp of the event. |
| IP Address | uint8[4] | IP address from the host where the user was detected logging in, in IP address octets. |
| String Block Type | uint32 | Initiates a String data block containing the username for the user. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the username String data block, including eight bytes for the block type and length fields, plus the number of bytes in the username. |
| Username | string | The user name for the user. |
| User ID | uint32 | Identification number of the user. |
| Application ID | uint32 | The application ID for the application protocol used in the connection that the login information was derived from. |
| String Block Type | uint32 | Initiates a String data block containing the email address for the user. This value is always `0`. |

*Table B-25 User Login Information Data Block Fields 5.0 - 5.0.2 (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | Number of bytes in the email address String data block, including eight bytes for the block type and length fields, plus the number of bytes in the email address. |
| Email | string | The email address for the user. |

## User Login Information Data Block 5.1-5.4.x

The User Login Information data block is used in User Information Update messages and conveys changes in login information for a detected user. For more information, see User Account Update Message Data Block, page 4-179.

The User Login Information data block has a block type of 73 for version 4.7 - 4.10.x, a block type of 121 in the series 1 group of blocks for version 5.0 - 5.0.2, and a block type of 127 in the series 1 group of blocks for version 5.1-5.4.x.

The graphic below shows the format of the User Login Information data block:

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | User Login Information Block Type (127) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | User Login Information Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | IPv4 Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | User Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | User ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Application ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Email | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Email... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | IPv6 Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reported By | Login Type | | | | | | | | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Type (0), cont. | | | | | | | | String Block Length | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | Reported By... | | | | | | | | | | | | | | | | | | | | | | | |

The following table describes the components of the User Login Information data block.

***Table B-26 User Login Information Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| User Login Information Block Type | uint32 | Initiates a User Login Information data block. This value is `127` for version 5.1+. |
| User Login Information Block Length | uint32 | Total number of bytes in the User Login Information data block, including eight bytes for the user login information block type and length fields, plus the number of bytes in the user login information data that follows. |
| Timestamp | uint32 | Timestamp of the event. |
| IPv4 Address | uint32 | This field is reserved but no longer populated. The IPv4 address is stored in the IPv6 Address field. See IP Addresses, page 1-4 for more information. |
| String Block Type | uint32 | Initiates a String data block containing the username for the user. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the username String data block, including eight bytes for the block type and length fields, plus the number of bytes in the username. |
| Username | string | The user name for the user. |
| User ID | uint32 | Identification number of the user. |
| Application ID | uint32 | The application ID for the application protocol used in the connection that the login information was derived from. |
| String Block Type | uint32 | Initiates a String data block containing the email address for the user. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the email address String data block, including eight bytes for the block type and length fields, plus the number of bytes in the email address. |
| Email | string | The email address for the user. |
| IPv6 Address | uint8[16] | IPv6 address from the host where the user was detected logging in, in IP address octets. |

*Table B-26 User Login Information Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Login Type | uint8 | The type of user login detected. |
| String Block Type | uint32 | Initiates a String data block containing the Reported By value. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the Reported By String data block, including eight bytes for the block type and length fields, plus the number of bytes in the Reported By field. |
| Reported By | string | The name of the Active Directory server reporting a login. |

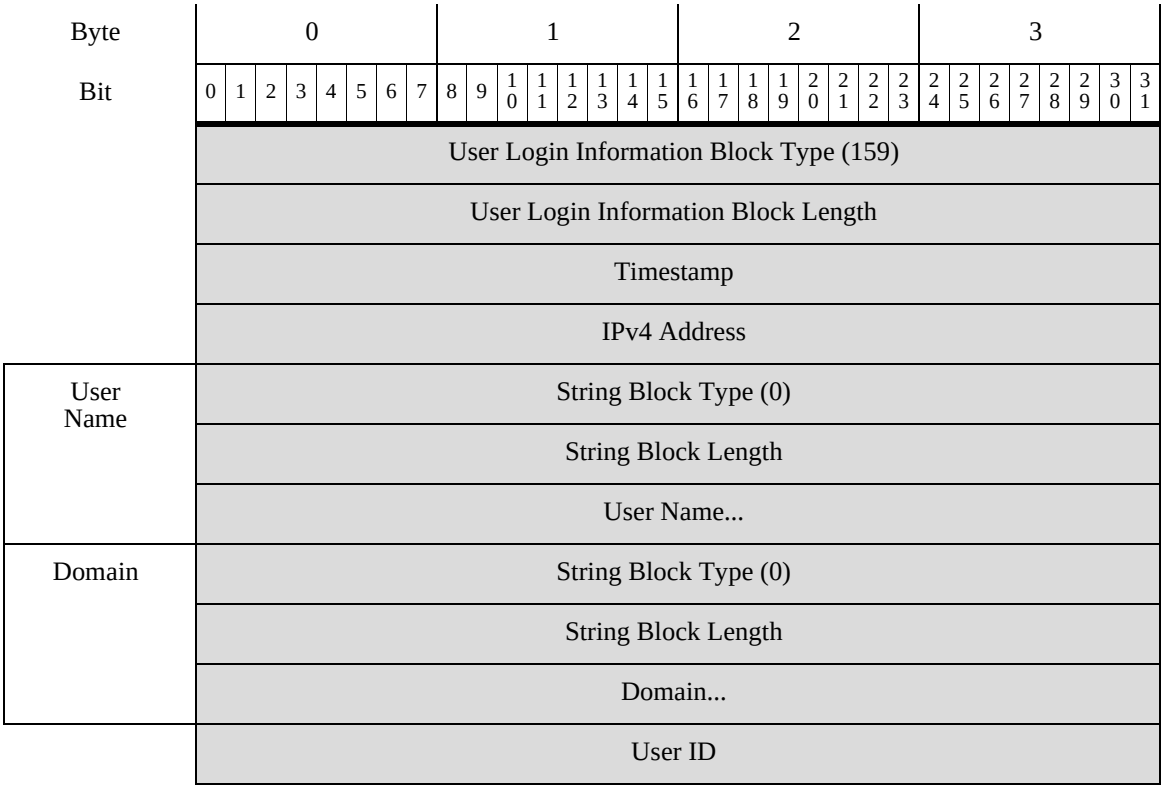
## User Login Information Data Block 6.0.x

The User Login Information data block is used in User Information Update messages and conveys changes in login information for a detected user. For more information, see User Account Update Message Data Block, page 4-179.

he User Login Information data block has a block type of 159 for version 6.0.x. It has new ISE integration endpoint profile, Security Intelligence fields.

The User Login Information data block has a block type of 73 for version 4.7 - 4.10.x, a block type of 121 in the series 1 group of blocks for version 5.0 - 5.0.2, and a block type of 127 in the series 1 group of blocks for version 5.1+. See User Login Information Data Block 5.1-5.4.x, page B-136 for more information.

The graphic below shows the format of the User Login Information data block:



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Realm ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Endpoint Profile ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Security Group ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Protocol | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Email | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Email... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | IPv6 Address<br>IPv6 Address, continued<br>IPv6 Address, continued<br>IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Location IPv6 Address<br>Location IPv6 Address, continued<br>Location IPv6 Address, continued<br>Location IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reported By | Login Type | | | | | | | | Auth. Type | | | | | | | | String Block Type (0) | | | | | | | | | | | | | | | |
| | String Block Type (0), cont. | | | | | | | | | | | | | | | | String Block Length | | | | | | | | | | | | | | | |
| | String Block Length, cont. | | | | | | | | | | | | | | | | Reported By... | | | | | | | | | | | | | | | |

The following table describes the components of the User Login Information data block.

***Table B-27 User Login Information Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| User Login Information Block Type | uint32 | Initiates a User Login Information data block. This value is `159` for version 6.0.x. |
| User Login Information Block Length | uint32 | Total number of bytes in the User Login Information data block, including eight bytes for the user login information block type and length fields, plus the number of bytes in the user login information data that follows. |
| Timestamp | uint32 | Timestamp of the event. |

*Table B-27 User Login Information Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| IPv4 Address | uint32 | This field is reserved but no longer populated. The IPv4 address is stored in the IPv6 Address field. See IP Addresses, page 1-4 for more information. |
| String Block Type | uint32 | Initiates a String data block containing the username for the user. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the username String data block, including eight bytes for the block type and length fields, plus the number of bytes in the username. |
| Username | string | The user name for the user. |
| String Block Type | uint32 | Initiates a String data block containing the domain. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the username String data block, including eight bytes for the block type and length fields, plus the number of bytes in the domain. |
| Domain | string | Domain in which the user logged in. |
| User ID | uint32 | Identification number of the user. |
| Realm ID | uint32 | Integer ID which corresponds to an identity realm. |
| Endpoint Profile ID | uint32 | ID number of the type of device used by the connection endpoint. This is unique for each DC and resolved in metadata. |
| Security Group ID | uint32 | ID number of the network traffic group. |
| Protocol | uint32 | Protocol used to detect or report the user. Possible values are:<br>• `165` - FTP<br>• `426` - SIP<br>• `547` - AOL Instant Messenger<br>• `683` - IMAP<br>• `710` - LDAP<br>• `767` - NTP<br>• `773` - Oracle Database<br>• `788` - POP3<br>• `1755` - MDNS |
| String Block Type | uint32 | Initiates a String data block containing the email address for the user. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the email address String data block, including eight bytes for the block type and length fields, plus the number of bytes in the email address. |
| Email | string | The email address for the user. |
| IPv6 Address | uint8[16] | IPv6 address from the host where the user was detected logging in, in IP address octets. |
| Location IPv6 Address | uint8[16] | Most recent IP address on which the user logged in. Can be either an IPv4 or IPv6 address. |

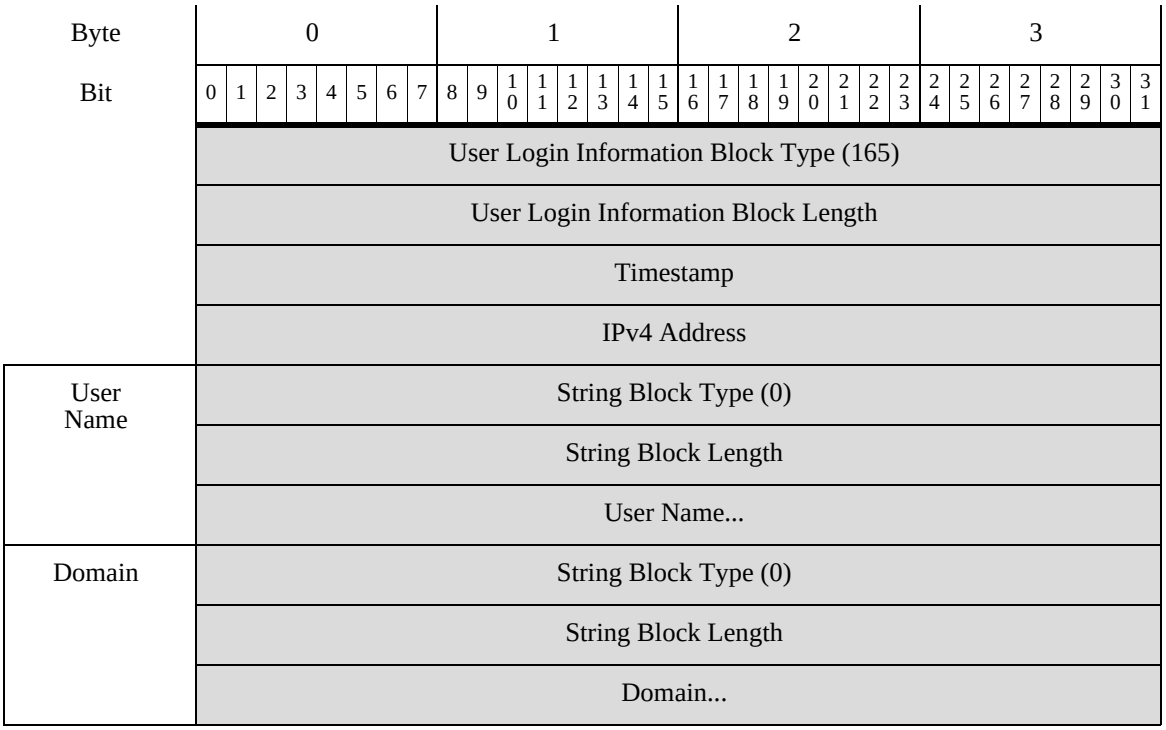
*Table B-27 User Login Information Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Login Type | uint8 | The type of user login detected. |
| Authentication Type | uint8 | Type of authentication used by the user. Values may be:<br>• 0 - no authorization required<br>• 1 - passive authentication, AD agent, or ISE session<br>• 2 - captive portal successful authentication<br>• 3 - captive portal guest authentication<br>• 4 - captive portal failed authentication |
| String Block Type | uint32 | Initiates a String data block containing the Reported By value. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the Reported By String data block, including eight bytes for the block type and length fields, plus the number of bytes in the Reported By field. |
| Reported By | string | The name of the Active Directory server reporting a login. |

## User Login Information Data Block 6.1.x

The User Login Information data block has a block type of 165 in the series 1 group of blocks for version 6.1+. It has new port and tunneling fields. It supersedes block type 159. See User Login Information Data Block 6.0.x, page B-138 for more information. It is superseded by block type 167.

The graphic below shows the format of the User Login Information data block:



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | User ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Realm ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Endpoint Profile ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Security Group ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Protocol | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Port | | | | | | | | | | | | | | | | Range Start | | | | | | | | | | | | | | | |
| | Start Port | | | | | | | | | | | | | | | | End Port | | | | | | | | | | | | | | | |
| Email | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Email... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | IPv6 Address<br>IPv6 Address, continued<br>IPv6 Address, continued<br>IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Location IPv6 Address<br>Location IPv6 Address, continued<br>Location IPv6 Address, continued<br>Location IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reported By | Login Type | | | | | | | | Auth. Type | | | | | | | | String Block Type (0) | | | | | | | | | | | | | | | |
| | String Block Type (0), cont. | | | | | | | | | | | | | | | | String Block Length | | | | | | | | | | | | | | | |
| | String Block Length, cont. | | | | | | | | | | | | | | | | Reported By... | | | | | | | | | | | | | | | |

The following table describes the components of the User Login Information data block.

*Table B-28 User Login Information Data Block Fields*

| Field | Data Type | Description |
|---|---|---|
| User Login Information Block Type | uint32 | Initiates a User Login Information data block. This value is `165` for version 6.1+. |
| User Login Information Block Length | uint32 | Total number of bytes in the User Login Information data block, including eight bytes for the user login information block type and length fields, plus the number of bytes in the user login information data that follows. |
| Timestamp | uint32 | Timestamp of the event. |
| IPv4 Address | uint32 | This field is reserved but no longer populated. The IPv4 address is stored in the IPv6 Address field. See IP Addresses, page 1-4 for more information. |
| String Block Type | uint32 | Initiates a String data block containing the username for the user. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the username String data block, including eight bytes for the block type and length fields, plus the number of bytes in the username. |
| Username | string | The user name for the user. |
| String Block Type | uint32 | Initiates a String data block containing the domain. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the username String data block, including eight bytes for the block type and length fields, plus the number of bytes in the domain. |
| Domain | string | Domain in which the user logged in. |
| User ID | uint32 | Identification number of the user. |
| Realm ID | uint32 | Integer ID which corresponds to an identity realm. |
| Endpoint Profile ID | uint32 | ID number of the type of device used by the connection endpoint. This is unique for each DC and resolved in metadata. |
| Security Group ID | uint32 | ID number of the network traffic group. |
| Protocol | uint32 | Protocol used to detect or report the user. Possible values are:<br>• `165` - FTP<br>• `426` - SIP<br>• `547` - AOL Instant Messenger<br>• `683` - IMAP<br>• `710` - LDAP<br>• `767` - NTP<br>• `773` - Oracle Database<br>• `788` - POP3<br>• `1755` - MDNS |
| Port | uint16 | The port number on which the user was detected. |

*Table B-28 User Login Information Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Range Start | uint16 | The start port in the port range used by the TS Agent. |
| Start Port | uint16 | The start port in the range the TS Agent assigned to the individual user. |
| End Port | uint16 | The end port in the range the TS Agent assigned to the individual user. |
| String Block Type | uint32 | Initiates a String data block containing the email address for the user. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the email address String data block, including eight bytes for the block type and length fields, plus the number of bytes in the email address. |
| Email | string | The email address for the user. |
| IPv6 Address | uint8[16] | IPv6 address from the host where the user was detected logging in, in IP address octets. |
| Location IPv6 Address | uint8[16] | Most recent IP address on which the user logged in. Can be either an IPv4 or IPv6 address. |
| Login Type | uint8 | The type of user login detected. |
| Authentication Type | uint8 | Type of authentication used by the user. Values may be:<br>• `0` - no authorization required<br>• `1` - passive authentication, AD agent, or ISE session<br>• `2` - captive portal successful authentication<br>• `3` - captive portal guest authentication<br>• `4` - captive portal failed authentication |
| String Block Type | uint32 | Initiates a String data block containing the Reported By value. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the Reported By String data block, including eight bytes for the block type and length fields, plus the number of bytes in the Reported By field. |
| Reported By | string | The name of the Active Directory server reporting a login. |

# User Login Information Data Block 6.1.x

The User Login Information data block is used in User Information Update messages and conveys changes in login information for a detected user. For more information, see User Information Update Message Block, page 4-62.

The User Login Information data block has a block type of 165 in the series 1 group of blocks for version 6.1x. It has new port and tunneling fields. It supersedes block type 159. It is superseded by block type 167. See User Login Information Data Block 6.0.x, page B-138 for more information.

The graphic below shows the format of the User Login Information data block:

| | Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | User Login Information Block Type (165) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | User Login Information Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | IPv4 Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | User Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Domain | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Domain... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | User ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Realm ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Endpoint Profile ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Security Group ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Protocol | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Port | | | | | | | | | | | | | | | | | Range Start | | | | | | | | | | | | | | | |
| | Start Port | | | | | | | | | | | | | | | | | End Port | | | | | | | | | | | | | | | |
| Email | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Email... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | IPv6 Address<br>IPv6 Address, continued<br>IPv6 Address, continued<br>IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Location IPv6 Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Location IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Location IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Location IPv6 Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reported By | Login Type | | | | | | | | Auth. Type | | | | | | | | String Block Type (0) | | | | | | | | | | | | | | | |
| | String Block Type (0), cont. | | | | | | | | | | | | | | | | String Block Length | | | | | | | | | | | | | | | |
| | String Block Length, cont. | | | | | | | | | | | | | | | | Reported By... | | | | | | | | | | | | | | | |
| Domain | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Description... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

The following table describes the components of the User Login Information data block.

***Table B-29 User Login Information Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| User Login Information Block Type | uint32 | Initiates a User Login Information data block. This value is `165` for version 6.2+. |
| User Login Information Block Length | uint32 | Total number of bytes in the User Login Information data block, including eight bytes for the user login information block type and length fields, plus the number of bytes in the user login information data that follows. |
| Timestamp | uint32 | Timestamp of the event. |
| IPv4 Address | uint32 | This field is reserved but no longer populated. The IPv4 address is stored in the IPv6 Address field. See IP Addresses, page 1-4 for more information. |
| String Block Type | uint32 | Initiates a String data block containing the username for the user. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the username String data block, including eight bytes for the block type and length fields, plus the number of bytes in the username. |
| Username | string | The user name for the user. |
| String Block Type | uint32 | Initiates a String data block containing the domain. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the username String data block, including eight bytes for the block type and length fields, plus the number of bytes in the domain. |

*Table B-29 User Login Information Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Domain | string | Domain in which the user logged in. |
| User ID | uint32 | Identification number of the user. |
| Realm ID | uint32 | Integer ID which corresponds to an identity realm. |
| Endpoint Profile ID | uint32 | ID number of the type of device used by the connection endpoint. This is unique for each DC and resolved in metadata. |
| Security Group ID | uint32 | ID number of the network traffic group. |
| Protocol | uint32 | Protocol used to detect or report the user. Possible values are:<br>• 165 - FTP<br>• 426 - SIP<br>• 547 - AOL Instant Messenger<br>• 683 - IMAP<br>• 710 - LDAP<br>• 767 - NTP<br>• 773 - Oracle Database<br>• 788 - POP3<br>• 1755 - MDNS |
| Port | uint16 | The port number on which the user was detected. |
| Range Start | uint16 | The start port in the port range used by the TS Agent. |
| Start Port | uint16 | The start port in the range the TS Agent assigned to the individual user. |
| End Port | uint16 | The end port in the range the TS Agent assigned to the individual user. |
| String Block Type | uint32 | Initiates a String data block containing the email address for the user. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the email address String data block, including eight bytes for the block type and length fields, plus the number of bytes in the email address. |
| Email | string | The email address for the user. |
| IPv6 Address | uint8[16] | IPv6 address from the host where the user was detected logging in, in IP address octets. |
| Location IPv6 Address | uint8[16] | Most recent IP address on which the user logged in. Can be either an IPv4 or IPv6 address. |
| Login Type | uint8 | The type of user login detected. |
| Authentication Type | uint8 | Type of authentication used by the user. Values may be:<br>• 0 - no authorization required<br>• 1 - passive authentication, AD agent, or ISE session<br>• 2 - captive portal successful authentication<br>• 3 - captive portal guest authentication<br>• 4 - captive portal failed authentication |

*Table B-29 User Login Information Data Block Fields (continued)*

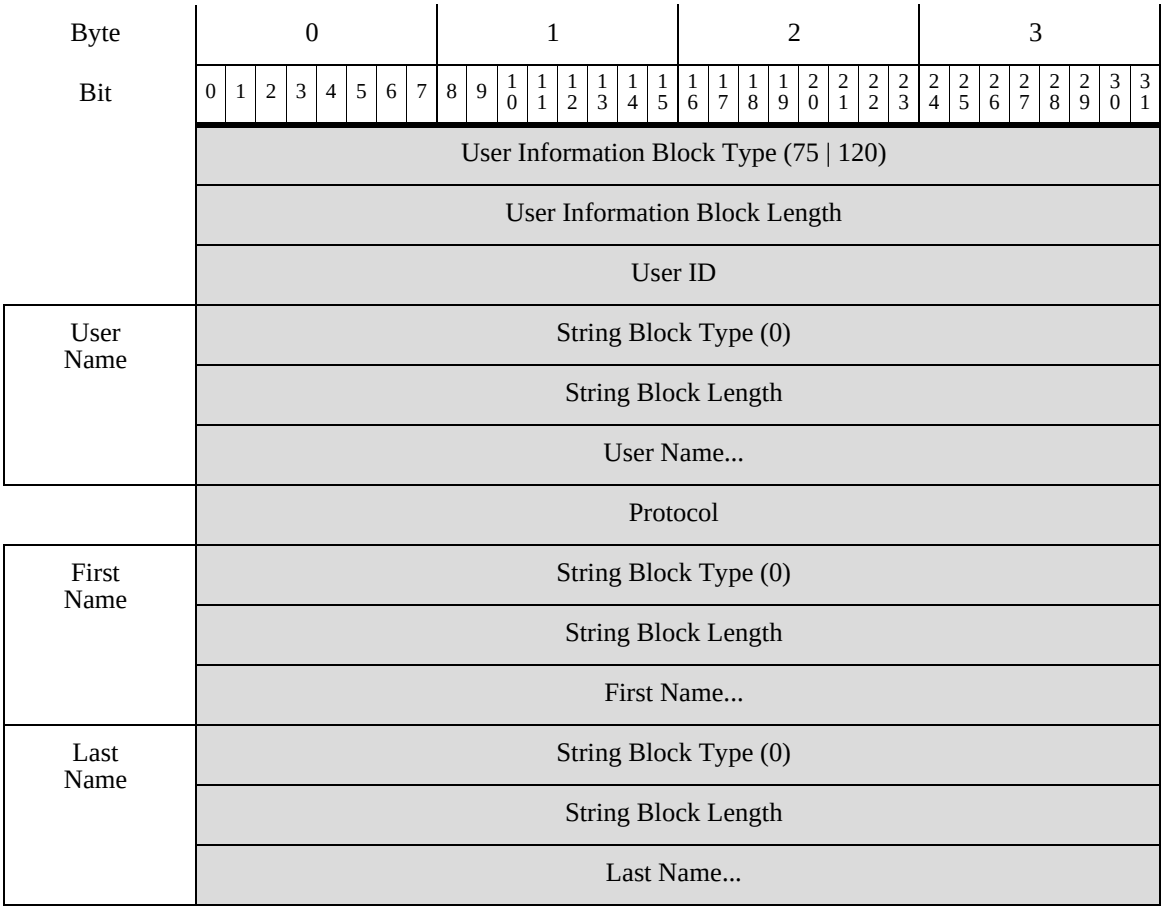
| Field | Data Type | Description |
|---|---|---|
| String Block Type | uint32 | Initiates a String data block containing the Reported By value. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the Reported By String data block, including eight bytes for the block type and length fields, plus the number of bytes in the Reported By field. |
| Reported By | string | The name of the Active Directory server reporting a login. |

## User Information Data Block for 5.x

The User Information data block is used in User Modification messages and conveys information for a user detected, removed, or dropped. For more information, see User Modification Messages, page 4-61

The User Information data block has a block type of 75 in the series 1 group of blocks for version 4.7 - 4.10.x and a block type of 120 in the series 1 group of blocks for 5.x. The structures are the same for block types 75 and 120.

The following diagram shows the format of the User Information data block:



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Email | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Email... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Department | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Department... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Phone | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Phone... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

The following table describes the components of the User Information data block.

***Table B-30 User Information Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| User Information Block Type | uint32 | Initiates a User Information data block. This value is `75` for version 4.7 - 4.10.x and a value of `120` for 5.0+. |
| User Information Block Length | uint32 | Total number of bytes in the User Information data block, including eight bytes for the user information block type and length fields plus the number of bytes in the user information data that follows. |
| User ID | uint32 | Identification number of the user. |
| String Block Type | uint32 | Initiates a String data block containing the username for the user. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the username String data block, including eight bytes for the block type and length fields plus the number of bytes in the username. |
| Username | string | The username for the user. |
| Protocol | uint32 | The protocol for the packet containing the user information. |
| String Block Type | uint32 | Initiates a String data block containing the first name of the user. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the first name String data block, including eight bytes for the block type and length fields plus the number of bytes in the first name. |
| First Name | string | The first name for the user. |
| String Block Type | uint32 | Initiates a String data block containing the last name for the user. This value is always `0`. |

*Table B-30 User Information Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | Number of bytes in the user last name String data block, including eight bytes for the block type and length fields, plus the number of bytes in the last name. |
| Last Name | string | The last name for the user. |
| String Block Type | uint32 | Initiates a String data block containing the email address for the user. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the email address String data block, including eight bytes for the block type and length fields, plus the number of bytes in the email address. |
| Email | string | The email address for the user. |
| String Block Type | uint32 | Initiates a String data block containing the department for the user. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the department String data block, including eight bytes for the block type and length fields, plus the number of bytes in the department. |
| Department | string | The department for the user. |
| String Block Type | uint32 | Initiates a String data block containing the phone number for the user. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the phone number String data block, including eight bytes for the block type and length fields, plus the number of bytes in the phone number. |
| Phone | string | The phone number for the user. |

# Legacy Host Profile Data Blocks

See the following sections for more information:

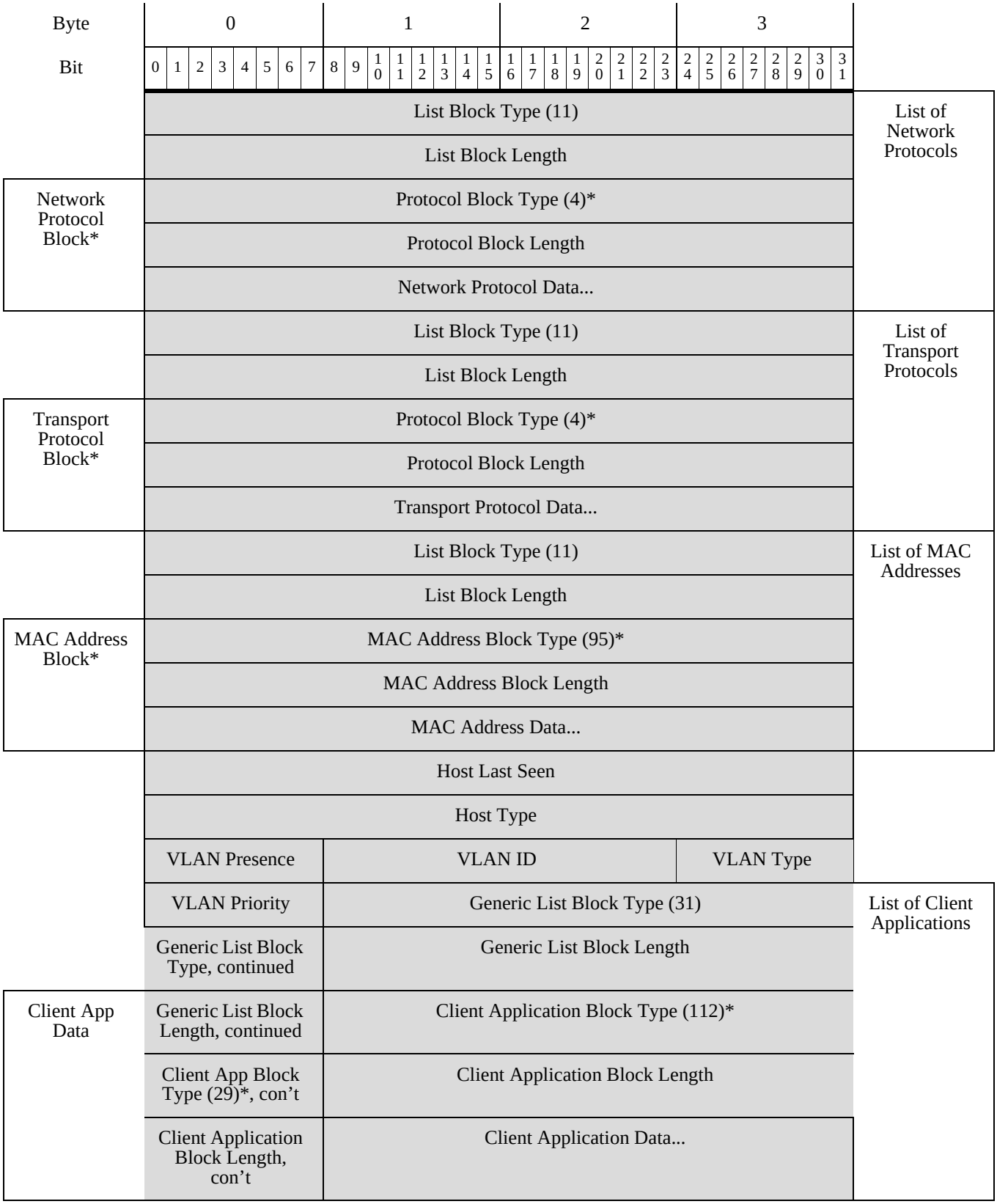
- Host Profile Data Block for 5.0 - 5.0.2, page B-150

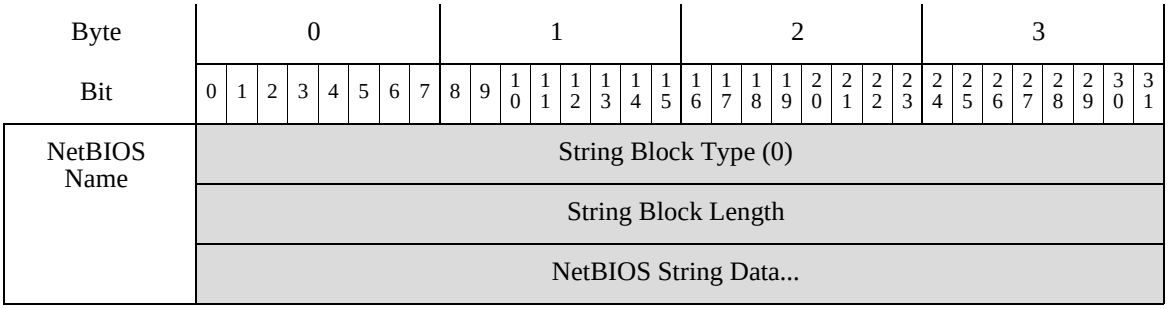
## Host Profile Data Block for 5.0 - 5.0.2

The following diagram shows the format of a Host Profile data block in versions 5.0 to 5.0.2. The Host Profile data block also does not include a host criticality value, but does include a VLAN presence indicator. In addition, a Host Profile data block can convey a NetBIOS name for the host. This Host Profile data block has a block type of 91.

**Note** An asterisk(*) next to a block type field in the following diagram indicates the message may contain zero or more instances of the series 1 data block.

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Host Profile Block Type (91) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Host Profile Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Server Fingerprints | Hops | | | | | | | | Primary/Secondary | | | | | | | | Generic List Block Type (31) | | | | | | | | | | | | | | | |
| | Generic List Block Type, continued | | | | | | | | | | | | | | | | Generic List Block Length | | | | | | | | | | | | | | | |
| | Generic List Block Length, continued | | | | | | | | | | | | | | | | Server Fingerprint Data Blocks* | | | | | | | | | | | | | | | |
| Client Fingerprints | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Client Fingerprint Data Blocks* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| SMB Fingerprints | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SMB Fingerprint Data Blocks* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DHCP Fingerprints | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | DHCP Fingerprint Data Blocks* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | List of TCP Servers |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| TCP Server Block* | Server Block Type (36) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Server Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | TCP Server Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | List of UDP Servers |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| UDP Server Block* | Server Block Type (36)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Server Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | UDP Server Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
List Block Type (11)
List Block Length
Protocol Block Type (4)*
Protocol Block Length
Network Protocol Data...
List Block Type (11)
List Block Length
Protocol Block Type (4)*
Protocol Block Length
Transport Protocol Data...
List Block Type (11)
List Block Length
MAC Address Block Type (95)*
MAC Address Block Length
MAC Address Data...
Host Last Seen
Host Type
VLAN Presence
VLAN ID
VLAN Type
VLAN Priority
Generic List Block Type (31)
Generic List Block Type, continued
Generic List Block Length
Generic List Block Length, continued
Client Application Block Type (112)*
Client App Block Type (29)*, con't
Client Application Block Length
Client Application Block Length, con't
Client Application Data...
List of Network Protocols
Network Protocol Block*
List of Transport Protocols
Transport Protocol Block*
List of MAC Addresses
MAC Address Block*
List of Client Applications
Client App Data




The following table describes the fields of the host profile data block returned by version 4.9 to version 5.0.2.

***Table B-31 Host Profile Data Block for 5.0 - 5.0.2 Fields***

| Field | Data Type | Description |
|---|---|---|
| Host Profile Block Type | uint32 | Initiates the Host Profile data block for 4.9 to 5.0.2. This data block has a block type of `91`. |
| Host Profile Block Length | uint32 | Number of bytes in the Host Profile data block, including eight bytes for the host profile block type and length fields, plus the number of bytes included in the host profile data that follows. |
| IP Address | uint8[4] | IP address of the host described in the profile, in IP address octets. |
| Hops | uint8 | Number of hops from the host to the device. |
| Primary/ Secondary | uint8 | Indicates whether the host is in the primary or secondary network of the device that detected it:<br>• `0` — Host is in the primary network.<br>• `1` — Host is in the secondary network. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a server fingerprint. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Server Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a server fingerprint. See Operating System Fingerprint Data Block for 5.0 - 5.0.2, page B-157 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a client fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |

*Table B-31 Host Profile Data Block for 5.0 - 5.0.2 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Operating System Fingerprint (Client Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a client fingerprint. See Operating System Fingerprint Data Block for 5.0 - 5.0.2, page B-157 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using an SMB fingerprint. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (SMB Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using an SMB fingerprint. See Operating System Fingerprint Data Block for 5.0 - 5.0.2, page B-157 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a DHCP fingerprint. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (DHCP Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a DHCP fingerprint. See Operating System Fingerprint Data Block for 5.0 - 5.0.2, page B-157 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Server data blocks conveying TCP server data. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus all encapsulated Server data blocks.<br>This field is followed by zero or more Server data blocks. |
| Server Block Type | uint32 | Initiates a Server data block. This value is always `89`. |
| Server Block Length | uint32 | Number of bytes in the Server data block, including eight bytes for the server block type and length fields, plus the number of bytes of TCP server data that follows. |
| TCP Server Data | variable | Data fields describing a TCP server (as documented for earlier versions of the product). |
| List Block Type | uint32 | Initiates a List data block comprising Server data blocks conveying UDP server data. This value is always `11`. |

*Table B-31 Host Profile Data Block for 5.0 - 5.0.2 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus all encapsulated Server data blocks.<br>This field is followed by zero or more Server data blocks. |
| Server Block Type | uint32 | Initiates a Server data block describing a UDP server. This value is always `89`. |
| Server Block Length | uint32 | Number of bytes in the Server data block, including eight bytes for the server block type and length fields, plus the number of bytes of UDP server data that follows. |
| UDP Server Data | variable | Data fields describing a UDP server (as documented for earlier versions of the product). |
| List Block Type | uint32 | Initiates a List data block comprising Protocol data blocks conveying network protocol data. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus all encapsulated Protocol data blocks.<br>This field is followed by zero or more Protocol data blocks. |
| Protocol Block Type | uint32 | Initiates a Protocol data block describing a network protocol. This value is always `4`. |
| Protocol Block Length | uint32 | Number of bytes in the Protocol data block, including eight bytes for the protocol block type and length fields, plus the number of bytes in the protocol data that follows. |
| Network Protocol Data | uint16 | Data field containing a network protocol number, as documented in Protocol Data Block, page 4-75. |
| List Block Type | uint32 | Initiates a List data block comprising Protocol data blocks conveying transport protocol data. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus all encapsulated Protocol data blocks.<br>This field is followed by zero or more transport protocol data blocks. |
| Protocol Block Type | uint32 | Initiates a Protocol data block describing a transport protocol. This value is always `4`. |
| Protocol Block Length | uint32 | Number of bytes in the protocol data block, including eight bytes for the protocol block type and length, plus the number of bytes in the protocol data that follows. |
| Transport Protocol Data | variable | Data field containing a transport protocol number, as documented in Protocol Data Block, page 4-75. |
| List Block Type | uint32 | Initiates a List data block comprising MAC Address data blocks. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the list, including the list header and all encapsulated MAC Address data blocks. |

*Table B-31 Host Profile Data Block for 5.0 - 5.0.2 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Host MAC Address Block Type | uint32 | Initiates a Host MAC Address data block. This value is always 95. |
| Host MAC Address Block Length | uint32 | Number of bytes in the Host MAC Address data block, including eight bytes for the Host MAC address block type and length fields, plus the number of bytes in the Host MAC address data that follows. |
| Host MAC Address Data | variable | Host MAC address data fields described in Host MAC Address 4.9+, page 4-115. |
| Host Last Seen | uint32 | UNIX timestamp that represents the last time the system detected host activity. |
| Host Type | uint32 | Indicates the host type. The following values may appear:<br>• 0 — Host<br>• 1 — Router<br>• 2 — Bridge<br>• 3 — NAT device<br>• 4 — LB (load balancer) |
| VLAN Presence | uint8 | Indicates whether a VLAN is present:<br>• 0 — Yes<br>• 1 — No |
| VLAN ID | uint16 | VLAN identification number that indicates which VLAN the host is a member of. |
| VLAN Type | uint8 | Type of packet encapsulated in the VLAN tag. |
| VLAN Priority | uint8 | Priority value included in the VLAN tag. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Client Application data blocks conveying client application data. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated client application data blocks. |
| Client Application Block Type | uint32 | Initiates a client application block. This value is always 5. |
| Client Application Block Length | uint32 | Number of bytes in the client application block, including eight bytes for the client application block type and length fields, plus the number of bytes in the client application data that follows. |
| Client Application Data | variable | Client application data fields describing a client application, as documented in Host Client Application Data Block for 5.0+, page 4-156. |
| String Block Type | uint32 | Initiates a string data block for the NetBIOS name. This value is set to 0 to indicate string data. |

*Table B-31 Host Profile Data Block for 5.0 - 5.0.2 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | Indicates the number of bytes in the NetBIOS name data block, including eight bytes for the string block type and length, plus the number of bytes in the NetBIOS name. |
| NetBIOS String Data | Variable | Contains the NetBIOS name of the host described in the host profile. |

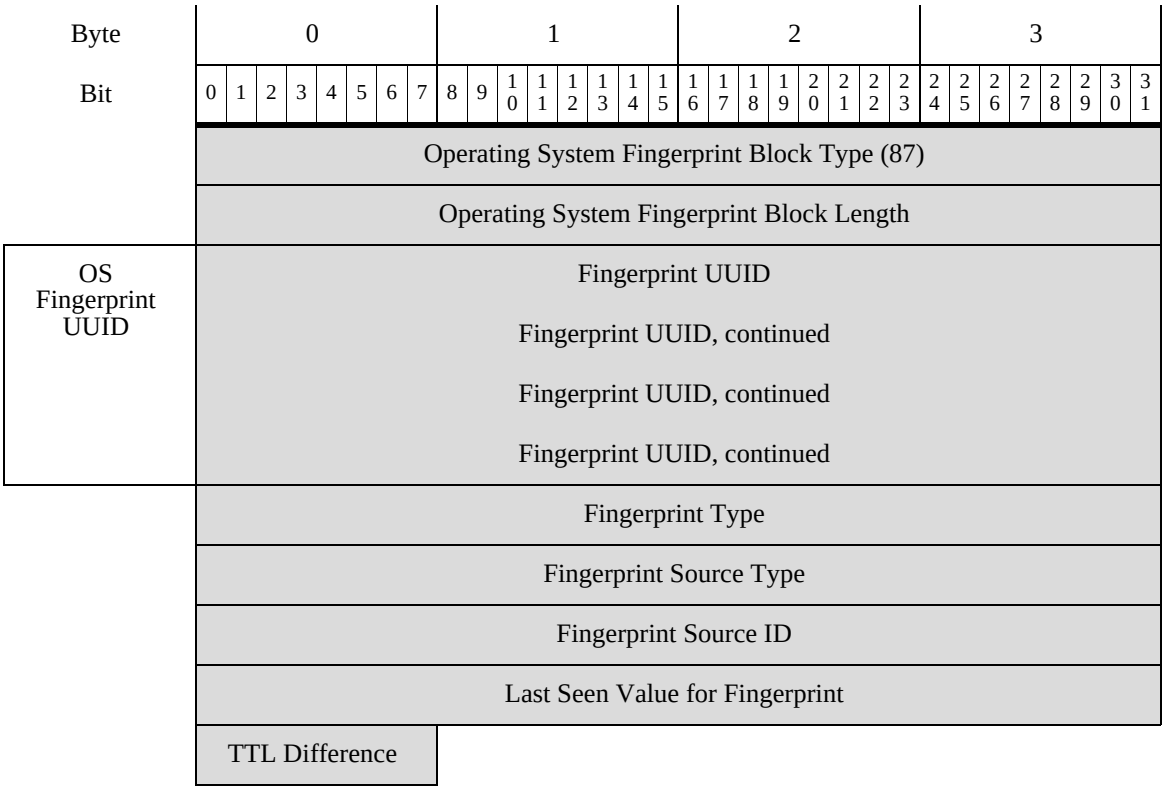
# Legacy OS Fingerprint Data Blocks

See the following sections for more information:

- Operating System Fingerprint Data Block for 5.0 - 5.0.2, page B-157

## Operating System Fingerprint Data Block for 5.0 - 5.0.2

The Operating System Fingerprint data block has a block type of 87. The block includes a fingerprint Universally Unique Identifier (UUID), as well as the fingerprint type, the fingerprint source type, and the fingerprint source ID. The following diagram shows the format of an Operating System Fingerprint data block for version 5.0 to version 5.0.2.




The following table describes the fields of the operating system fingerprint data block.

*Table B-32 Operating System Fingerprint Data Block Fields*

| Field | Data Type | Description |
|---|---|---|
| Operating System Fingerprint Data Block Type | uint32 | Initiates the operating system data block. This value is always `87`. |
| Operating System Data Block Length | uint32 | Number of bytes in the Operating System Fingerprint data block. This value should always be `41`: eight bytes for the data block type and length fields, sixteen bytes for the fingerprint UUID value, four bytes for the fingerprint type, four bytes for the fingerprint source type, four bytes for the fingerprint source ID, four bytes for the last seen value, and one byte for the TTL difference. |
| Fingerprint UUID | uint8[16] | Fingerprint identification number, in octets, that acts as a unique identifier for the operating system. The fingerprint UUID maps to the operating system name, vendor, and version in the vulnerability database (VDB). |
| Fingerprint Type | uint32 | Indicates the type of fingerprint. |
| Fingerprint Source Type | uint32 | Indicates the type (i.e., user or scanner) of the source that supplied the operating system fingerprint. |
| Fingerprint Source ID | uint32 | Indicates the ID of the source that supplied the operating system fingerprint. |
| Last Seen | uint32 | Indicates when the fingerprint was last seen in traffic. |
| TTL Difference | uint8 | Indicates the difference between the TTL value in the fingerprint and the TTL value seen in the packet used to fingerprint the host. |

# Legacy Connection Data Structures

For more information, see the following sections:


- Connection Statistics Data Block 5.0 - 5.0.2, page B-159
- Connection Statistics Data Block 5.1, page B-163
- Connection Statistics Data Block 5.2.x, page B-169
- Connection Chunk Data Block for 5.0 - 5.1, page B-175
- Connection Chunk Data Block for 5.1.1-6.0.x, page B-176
- Connection Statistics Data Block 5.1.1.x, page B-178
- Connection Statistics Data Block 5.3, page B-184
- Connection Statistics Data Block 5.3.1, page B-191
- Connection Statistics Data Block 5.4, page B-198
- Connection Statistics Data Block 5.4.1, page B-211
- Connection Statistics Data Block 6.0.x, page B-224
- Connection Statistics Data Block 6.1.x, page B-239
- Connection Statistics Data Block 6.2-6.7.x, page B-256

- Connection Statistics Data Block 7.0, page B-272

# Connection Statistics Data Block 5.0 - 5.0.2

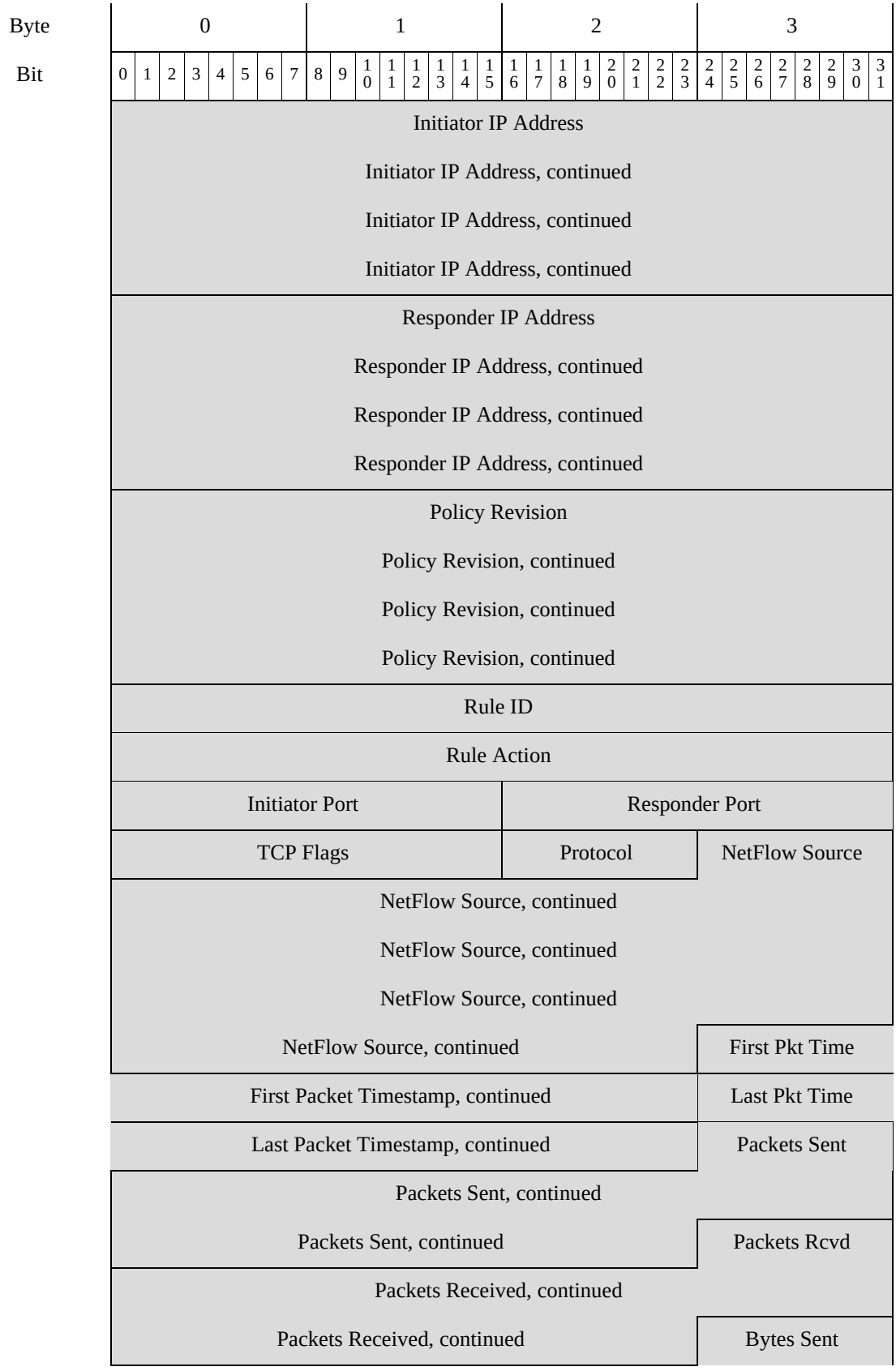
The Connection Statistics data block is used in Connection Data messages. The Connection Statistics data block for version 5.0 - 5.0.2 has a block type of 115.

For more information on the Connection Statistics Data message, see Connection Statistics Data Message, page 4-53.

The following diagram shows the format of a Connection Statistics data block for 5.0 - 5.0.2:

::

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Connection Data Block Type (115) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Connection Data Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Initiator IP Address
Initiator IP Address, continued
Initiator IP Address, continued
Initiator IP Address, continued
Responder IP Address
Responder IP Address, continued
Responder IP Address, continued
Responder IP Address, continued
Policy Revision
Policy Revision, continued
Policy Revision, continued
Policy Revision, continued
Rule ID
Rule Action
Initiator Port
Responder Port
TCP Flags
Protocol
NetFlow Source
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
First Pkt Time
First Packet Timestamp, continued
Last Pkt Time
Last Packet Timestamp, continued
Packets Sent
Packets Sent, continued
Packets Sent, continued
Packets Rcvd
Packets Received, continued
Packets Received, continued
Bytes Sent

| | Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | | Bytes Sent, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | Packets Received, continued | | | | | | | | | | | | | | | | | | | | | | | | Bytes Rcvd | | | | | | | |
| | | Bytes Received, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | Bytes Received, continued | | | | | | | | | | | | | | | | | | | | | | | | User ID | | | | | | | |
| | | User ID, continued | | | | | | | | | | | | | | | | | | | | | | | | Application Protocol ID | | | | | | | |
| | | Application Protocol ID, continued | | | | | | | | | | | | | | | | | | | | | | | | URL Category | | | | | | | |
| | | URL Category, continued | | | | | | | | | | | | | | | | | | | | | | | | URL Reputation | | | | | | | |
| | | URL Reputation, continued | | | | | | | | | | | | | | | | | | | | | | | | Client App ID | | | | | | | |
| | | Client Application ID, continued | | | | | | | | | | | | | | | | | | | | | | | | Web App ID | | | | | | | |
| | | Web Application ID, continued | | | | | | | | | | | | | | | | | | | | | | | | String Block Type (0) | | | | | | | |
| | Client App URL | String Block Type, continued | | | | | | | | | | | | | | | | | | | | | | | | String Block Length | | | | | | | |
| | | String Block Length, continued | | | | | | | | | | | | | | | | | | | | | | | | Client Application URL... | | | | | | | |
| | NetBIOS Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | NetBIOS Name.... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Client App Version | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | Client Application Version... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

The following table describes the fields of the Connection Statistics data block for 5.0 - 5.0.2.

*Table B-33 Connection Statistics Data Block 5.0 - 5.0.2 Fields*

| Field | Data Type | Description |
|---|---|---|
| Connection Statistics Data Block Type | uint32 | Initiates a Connection Statistics data block for 5.0 to 5.0.2. The value is always `115`. |
| Connection Statistics Data Block Length | uint32 | Number of bytes in the Connection Statistics data block, including eight bytes for the connection statistics block type and length fields, plus the number of bytes in the connection data that follows. |
| Device ID | uint32 | The device that detected the connection event. |
| Ingress Zone | uint8[16] | Ingress security zone in the event that triggered the policy violation. |
| Egress Zone | uint8[16] | Egress security zone in the event that triggered the policy violation. |
| Ingress Interface | uint8[16] | Interface for the inbound traffic. |
| Egress Interface | uint8[16] | Interface for the outbound traffic. |
| Initiator IP Address | uint8[16] | IP address of the host that initiated the session described in the connection event, in IP address octets. |
| Responder IP Address | uint8[16] | IP address of the host that responded to the initiating host, in IP address octets. |
| Policy Revision | uint8[16] | Revision number of the rule associated with the triggered correlation event, if applicable. |
| Rule ID | uint32 | Internal identifier for the rule that triggered the event, if applicable. |
| Rule Action | uint32 | The action selected in the user interface for that rule (allow, block, and so forth). |
| Initiator Port | uint16 | Port used by the initiating host. |
| Responder Port | uint16 | Port used by the responding host. |
| TCP Flags | uint16 | Indicates any TCP flags for the connection event. |
| Protocol | uint8 | The IANA-specified protocol number. |
| NetFlow Source | uint8[16] | IP address of the NetFlow-enabled device that exported the data for the connection |
| First Packet Timestamp | uint32 | UNIX timestamp of the date and time the first packet was exchanged in the session. |
| Last Packet Timestamp | uint32 | UNIX timestamp of the date and time the last packet was exchanged in the session. |
| Packets Sent | uint64 | Number of packets transmitted by the initiating host. |
| Packets Received | uint64 | Number of packets transmitted by the responding host. |
| Bytes Sent | uint64 | Number of bytes transmitted by the initiating host. |
| Bytes Received | uint64 | Number of bytes transmitted by the responding host. |
| User ID | uint32 | Internal identification number for the user who last logged into the host that generated the traffic. |
| Application Protocol ID | uint32 | Application ID of the application protocol. |

*Table B-33 Connection Statistics Data Block 5.0 - 5.0.2 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| URL Category | uint32 | The internal identification number of the URL category. |
| URL Reputation | uint32 | The internal identification number for the URL reputation. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| String Block Type | uint32 | Initiates a String data block for the client application URL. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the client application URL String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the client application URL string. |
| Client Application URL | string | URL the client application accessed, if applicable (`/files/index.html`, for example). |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for the client application version. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block for the client application version, including eight bytes for the string block type and length, plus the number of bytes in the version. |
| Client Application Version | string | Client application version. |

# Connection Statistics Data Block 5.1

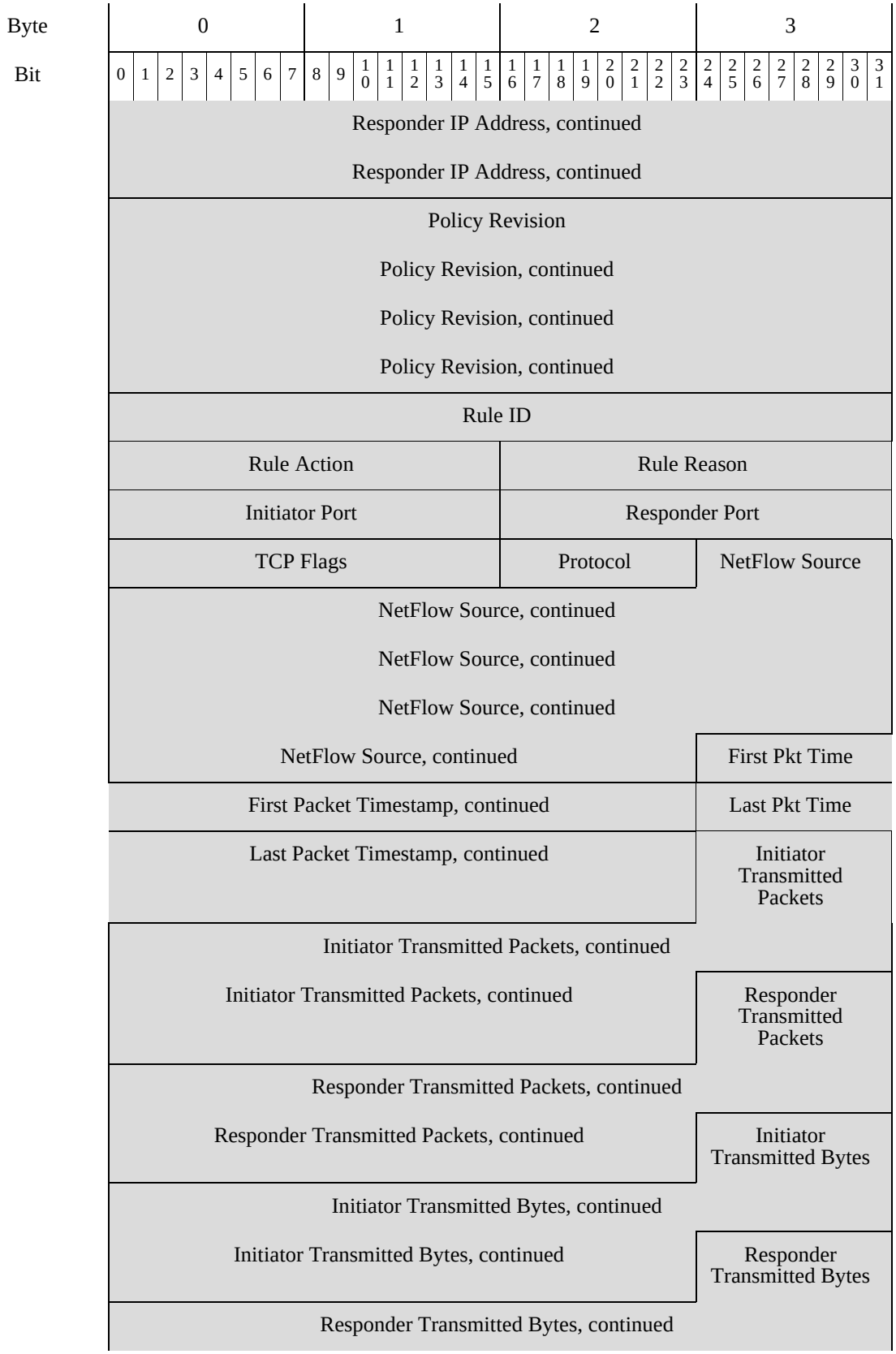
The Connection Statistics data block is used in Connection Data messages. Changes to the Connection data block between 5.0.2 and 5.1 include the addition of new fields with configuration parameters introduced in 5.1 (rule action reason, monitor rules, Security Intelligence source/destination, Security Intelligence layer). The Connection Statistics data block for version 5.1 has a block type of 126.

For more information on the Connection Statistics Data message, see Connection Statistics Data Message, page 4-53.

The following diagram shows the format of a Connection Statistics data block for 5.1:

::

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Connection Data Block Type (126) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Connection Data Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Responder IP Address, continued
Responder IP Address, continued
Policy Revision
Policy Revision, continued
Policy Revision, continued
Policy Revision, continued
Rule ID
Rule Action
Rule Reason
Initiator Port
Responder Port
TCP Flags
Protocol
NetFlow Source
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
First Pkt Time
First Packet Timestamp, continued
Last Pkt Time
Last Packet Timestamp, continued
Initiator Transmitted Packets
Initiator Transmitted Packets, continued
Initiator Transmitted Packets, continued
Responder Transmitted Packets
Responder Transmitted Packets, continued
Responder Transmitted Packets, continued
Initiator Transmitted Bytes
Initiator Transmitted Bytes, continued
Initiator Transmitted Bytes, continued
Responder Transmitted Bytes
Responder Transmitted Bytes, continued

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Responder Transmitted Bytes, continued | | | | | | | | | | | | | | | | | | | | | | | | User ID | | | | | | | |
| | User ID, continued | | | | | | | | | | | | | | | | | | | | | | | | Application Protocol ID | | | | | | | |
| | Application Protocol ID, continued | | | | | | | | | | | | | | | | | | | | | | | | URL Category | | | | | | | |
| | URL Category, continued | | | | | | | | | | | | | | | | | | | | | | | | URL Reputation | | | | | | | |
| | URL Reputation, continued | | | | | | | | | | | | | | | | | | | | | | | | Client App ID | | | | | | | |
| | Client Application ID, continued | | | | | | | | | | | | | | | | | | | | | | | | Web App ID | | | | | | | |
| | Web Application ID, continued | | | | | | | | | | | | | | | | | | | | | | | | String Block Type (0) | | | | | | | |
| Client App URL | String Block Type, continued | | | | | | | | | | | | | | | | | | | | | | | | String Block Length | | | | | | | |
| | String Block Length, continued | | | | | | | | | | | | | | | | | | | | | | | | Client Application URL... | | | | | | | |
| NetBIOS Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | NetBIOS Name.... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client App Version | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Client Application Version... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Sec. Int. Src/Dst | | | | | | | | Sec. Int. Rep Layer | | | | | | | | | | | | | | | | | | | | | | | |

The following table describes the fields of the Connection Statistics data block for 5.1.

***Table B-34 Connection Statistics Data Block 5.1 Fields***

| Field | Data Type | Description |
|---|---|---|
| Connection Statistics Data Block Type | uint32 | Initiates a Connection Statistics data block for 5.1. The value is always `126`. |
| Connection Statistics Data Block Length | uint32 | Number of bytes in the Connection Statistics data block, including eight bytes for the connection statistics block type and length fields, plus the number of bytes in the connection data that follows. |
| Device ID | uint32 | The device that detected the connection event. |
| Ingress Zone | uint8[16] | Ingress security zone in the event that triggered the policy violation. |
| Egress Zone | uint8[16] | Egress security zone in the event that triggered the policy violation. |
| Ingress Interface | uint8[16] | Interface for the inbound traffic. |
| Egress Interface | uint8[16] | Interface for the outbound traffic. |
| Initiator IP Address | uint8[16] | IP address of the host that initiated the session described in the connection event, in IP address octets. |
| Responder IP Address | uint8[16] | IP address of the host that responded to the initiating host, in IP address octets. |
| Policy Revision | uint8[16] | Revision number of the rule associated with the triggered correlation event, if applicable. |
| Rule ID | uint32 | Internal identifier for the rule that triggered the event, if applicable. |
| Rule Action | uint16 | The action selected in the user interface for that rule (allow, block, and so forth). |
| Rule Reason | uint16 | The reason the rule triggered the event. |
| Initiator Port | uint16 | Port used by the initiating host. |
| Responder Port | uint16 | Port used by the responding host. |
| TCP Flags | uint16 | Indicates any TCP flags for the connection event. |
| Protocol | uint8 | The IANA-specified protocol number. |
| NetFlow Source | uint8[16] | IP address of the NetFlow-enabled device that exported the data for the connection. |
| First Packet Timestamp | uint32 | UNIX timestamp of the date and time the first packet was exchanged in the session. |
| Last Packet Timestamp | uint32 | UNIX timestamp of the date and time the last packet was exchanged in the session. |
| Initiator Transmitted Packets | uint64 | Number of packets transmitted by the initiating host. |
| Responder Transmitted Packets | uint64 | Number of packets transmitted by the responding host. |
| Initiator Transmitted Bytes | uint64 | Number of bytes transmitted by the initiating host. |

*Table B-34 Connection Statistics Data Block 5.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Responder Transmitted Bytes | uint64 | Number of bytes transmitted by the responding host. |
| User ID | uint32 | Internal identification number for the user who last logged into the host that generated the traffic. |
| Application Protocol ID | uint32 | Application ID of the application protocol. |
| URL Category | uint32 | The internal identification number of the URL category. |
| URL Reputation | uint32 | The internal identification number for the URL reputation. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| String Block Type | uint32 | Initiates a String data block for the client application URL. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the client application URL String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the client application URL string. |
| Client Application URL | string | URL the client application accessed, if applicable (`/files/index.html`, for example). |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for the client application version. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block for the client application version, including eight bytes for the string block type and length, plus the number of bytes in the version. |
| Client Application Version | string | Client application version. |
| Monitor Rule 1 | uint32 | The ID of the first monitor rule associated with the connection event. |
| Monitor Rule 2 | uint32 | The ID of the second monitor rule associated with the connection event. |
| Monitor Rule 3 | uint32 | The ID of the third monitor rule associated with the connection event. |
| Monitor Rule 4 | uint32 | The ID of the fourth monitor rule associated with the connection event. |

*Table B-34 Connection Statistics Data Block 5.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Monitor Rule 5 | uint32 | The ID of the fifth monitor rule associated with the connection event. |
| Monitor Rule 6 | uint32 | The ID of the sixth monitor rule associated with the connection event. |
| Monitor Rule 7 | uint32 | The ID of the seventh monitor rule associated with the connection event. |
| Monitor Rule 8 | uint32 | The ID of the eighth monitor rule associated with the connection event. |
| Security Intelligence Source/ Destination | uint8 | Whether the source or destination IP address matched the IP block list. |
| Security Intelligence Layer | uint8 | The IP layer that matched the IP block list. |

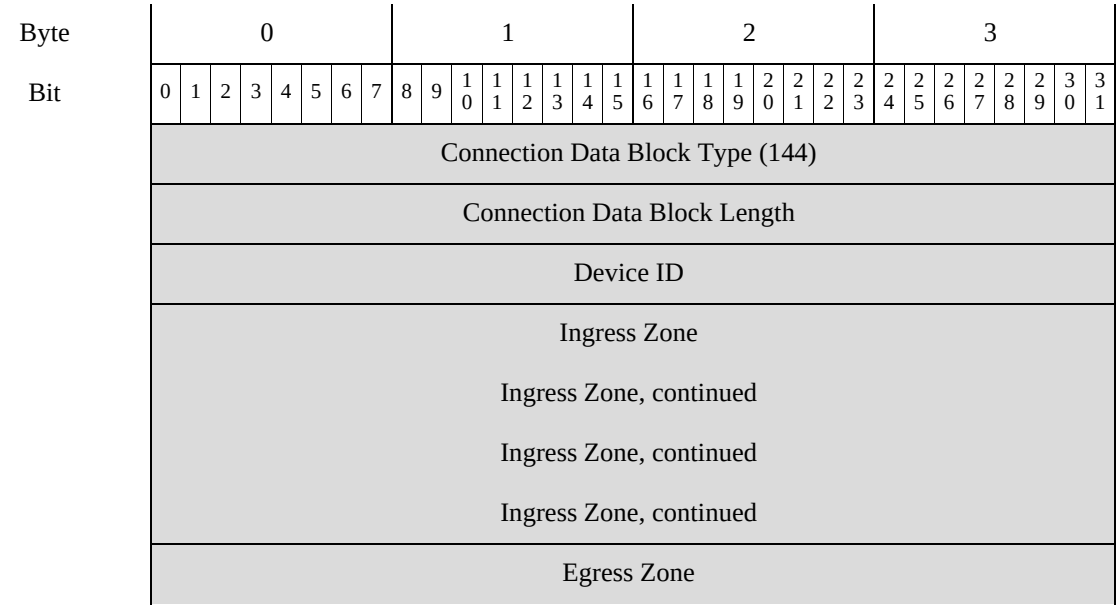
# Connection Statistics Data Block 5.2.x

The connection statistics data block is used in connection data messages. Changes to the connection data block between versions 5.1.1 and 5.2 include the addition of new fields to support geolocation. The connection statistics data block for version 5.2.x has a block type of 144 in the series 1 group of blocks. It deprecates block type 137, Connection Statistics Data Block 5.1.1.x, page B-178.

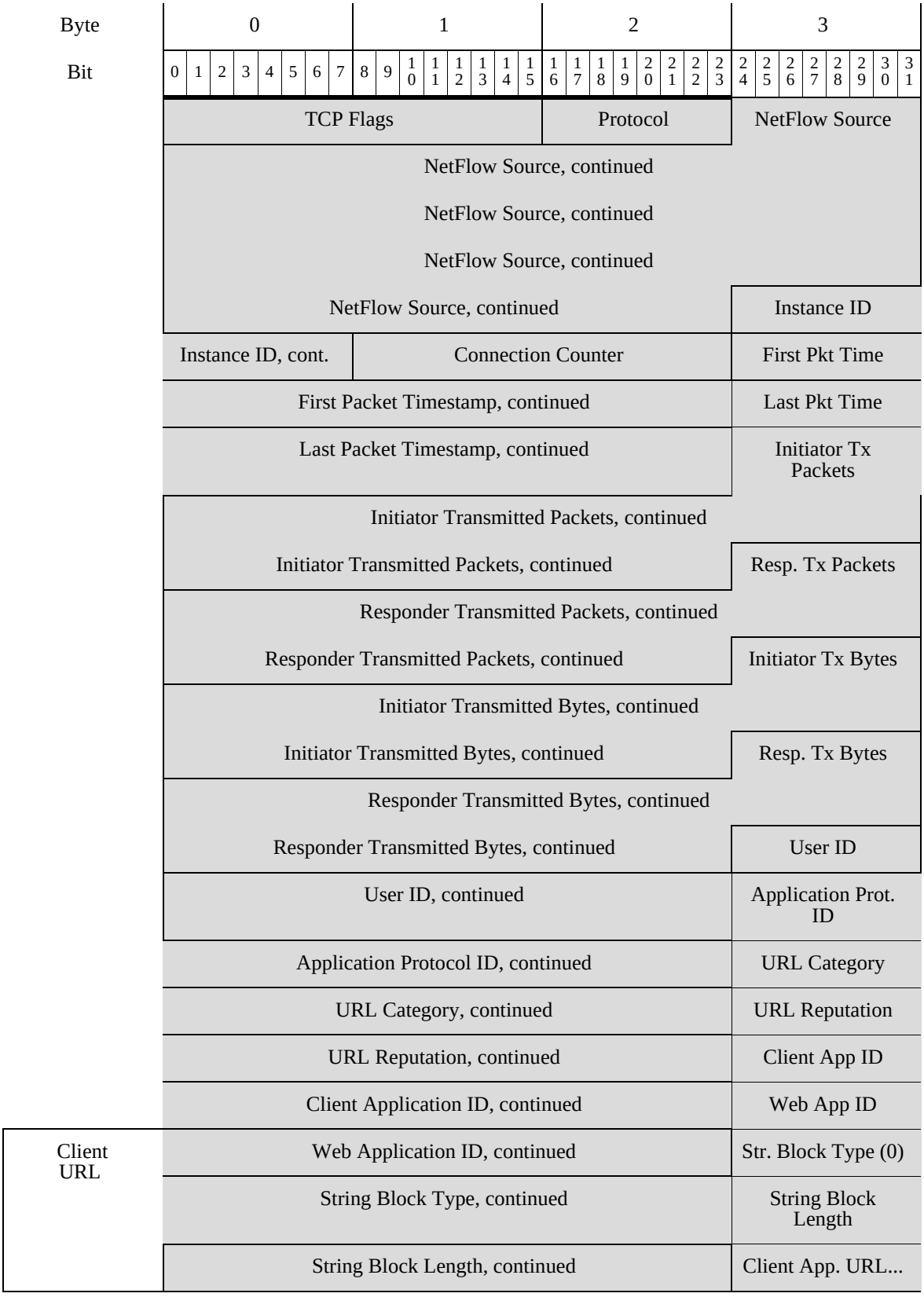
For more information on the Connection Statistics Data message, see Connection Statistics Data Message, page 4-53.

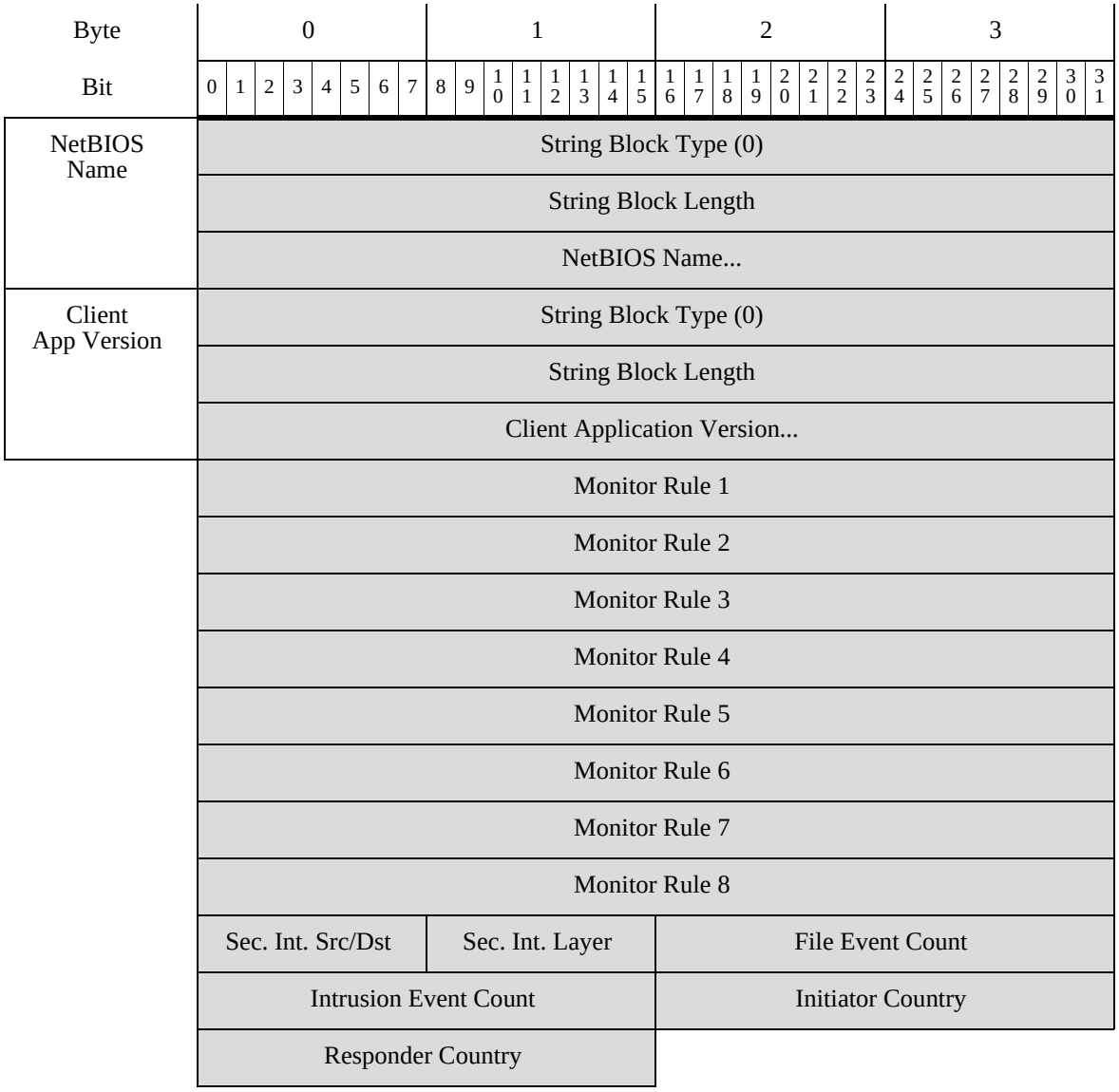
The following diagram shows the format of a Connection Statistics data block for 5.2.x:

::



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Action | | | | | | | | | | | | | | | | | Rule Reason | | | | | | | | | | | | | | | |
| Initiator Port | | | | | | | | | | | | | | | | | Responder Port | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
TCP Flags
Protocol
NetFlow Source
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
Instance ID
Instance ID, cont.
Connection Counter
First Pkt Time
First Packet Timestamp, continued
Last Pkt Time
Last Packet Timestamp, continued
Initiator Tx Packets
Initiator Transmitted Packets, continued
Initiator Transmitted Packets, continued
Resp. Tx Packets
Responder Transmitted Packets, continued
Responder Transmitted Packets, continued
Initiator Tx Bytes
Initiator Transmitted Bytes, continued
Initiator Transmitted Bytes, continued
Resp. Tx Bytes
Responder Transmitted Bytes, continued
Responder Transmitted Bytes, continued
User ID
User ID, continued
Application Prot. ID
Application Protocol ID, continued
URL Category
URL Category, continued
URL Reputation
URL Reputation, continued
Client App ID
Client Application ID, continued
Web App ID
Client URL
Web Application ID, continued
Str. Block Type (0)
String Block Type, continued
String Block Length
String Block Length, continued
Client App. URL...




The following table describes the fields of the Connection Statistics data block for 5.2.x:

***Table B-35 Connection Statistics Data Block 5.2.x Fields***

| Field | Data Type | Description |
|---|---|---|
| Connection Statistics Data Block Type | uint32 | Initiates a Connection Statistics data block for 5.2.x. The value is always `144`. |
| Connection Statistics Data Block Length | uint32 | Number of bytes in the Connection Statistics data block, including eight bytes for the connection statistics block type and length fields, plus the number of bytes in the connection data that follows. |
| Device ID | uint32 | The device that detected the connection event. |
| Ingress Zone | uint8[16] | Ingress security zone in the event that triggered the policy violation. |

*Table B-35 Connection Statistics Data Block 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Egress Zone | uint8[16] | Egress security zone in the event that triggered the policy violation. |
| Ingress Interface | uint8[16] | Interface for the inbound traffic. |
| Egress Interface | uint8[16] | Interface for the outbound traffic. |
| Initiator IP Address | uint8[16] | IP address of the host that initiated the session described in the connection event, in IP address octets. |
| Responder IP Address | uint8[16] | IP address of the host that responded to the initiating host, in IP address octets. |
| Policy Revision | uint8[16] | Revision number of the rule associated with the triggered correlation event, if applicable. |
| Rule ID | uint32 | Internal identifier for the rule that triggered the event, if applicable. |
| Rule Action | uint16 | The action selected in the user interface for that rule (allow, block, and so forth). |
| Rule Reason | uint16 | The reason the rule triggered the event. |
| Initiator Port | uint16 | Port used by the initiating host. |
| Responder Port | uint16 | Port used by the responding host. |
| TCP Flags | uint16 | Indicates any TCP flags for the connection event. |
| Protocol | uint8 | The IANA-specified protocol number. |
| NetFlow Source | uint8[16] | IP address of the NetFlow-enabled device that exported the data for the connection. |
| Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| First Packet Timestamp | uint32 | UNIX timestamp of the date and time the first packet was exchanged in the session. |
| Last Packet Timestamp | uint32 | UNIX timestamp of the date and time the last packet was exchanged in the session. |
| Initiator Transmitted Packets | uint64 | Number of packets transmitted by the initiating host. |
| Responder Transmitted Packets | uint64 | Number of packets transmitted by the responding host. |
| Initiator Transmitted Bytes | uint64 | Number of bytes transmitted by the initiating host. |
| Responder Transmitted Bytes | uint64 | Number of bytes transmitted by the responding host. |
| User ID | uint32 | Internal identification number for the user who last logged into the host that generated the traffic. |
| Application Protocol ID | uint32 | Application ID of the application protocol. |

*Table B-35 Connection Statistics Data Block 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| URL Category | uint32 | The internal identification number of the URL category. |
| URL Reputation | uint32 | The internal identification number for the URL reputation. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| String Block Type | uint32 | Initiates a String data block for the client application URL. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the client application URL String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the client application URL string. |
| Client Application URL | string | URL the client application accessed, if applicable (`/files/index.html`, for example). |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for the client application version. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block for the client application version, including eight bytes for the string block type and length, plus the number of bytes in the version. |
| Client Application Version | string | Client application version. |
| Monitor Rule 1 | uint32 | The ID of the first monitor rule associated with the connection event. |
| Monitor Rule 2 | uint32 | The ID of the second monitor rule associated with the connection event. |
| Monitor Rule 3 | uint32 | The ID of the third monitor rule associated with the connection event. |
| Monitor Rule 4 | uint32 | The ID of the fourth monitor rule associated with the connection event. |
| Monitor Rule 5 | uint32 | The ID of the fifth monitor rule associated with the connection event. |
| Monitor Rule 6 | uint32 | The ID of the sixth monitor rule associated with the connection event. |
| Monitor Rule 7 | uint32 | The ID of the seventh monitor rule associated with the connection event. |

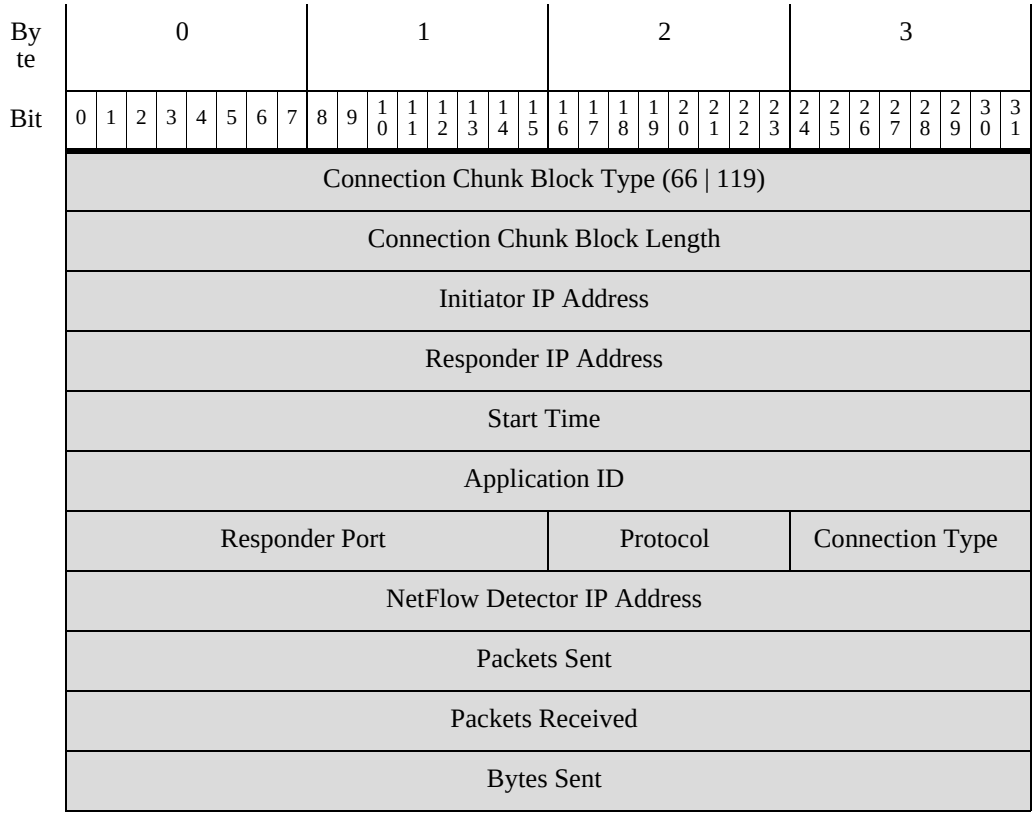
*Table B-35 Connection Statistics Data Block 5.2.x Fields (continued)*

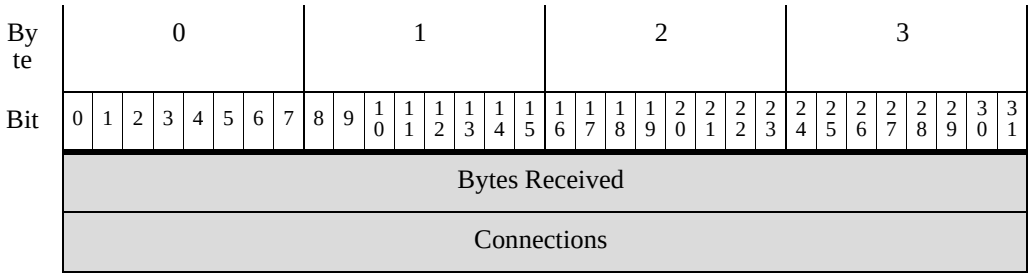
| Field | Data Type | Description |
|---|---|---|
| Monitor Rule 8 | uint32 | The ID of the eighth monitor rule associated with the connection event. |
| Security Intelligence Source/ Destination | uint8 | Whether the source or destination IP address matched the IP block list. |
| Security Intelligence Layer | uint8 | The IP layer that matched the IP block list. |
| File Event Count | uint16 | Value used to distinguish between file events that happen during the same second. |
| Intrusion Event Count | uint16 | Value used to distinguish between intrusion events that happen during the same second. |
| Initiator Country | uint16 | Code for the country of the initiating host. |
| Responder Country | uint16 | Code for the country of the responding host. |

# Connection Chunk Data Block for 5.0 - 5.1

The Connection Chunk data block conveys connection data detected by a NetFlow device. The Connection Chunk data block has a block type of 66 for pre-4.10.1 versions. For versions 5.0 - 5.1, it has a block type of 119.

The following diagram shows the format of the Connection Chunk data block:






The following table describes the components of the Connection Chunk data block:

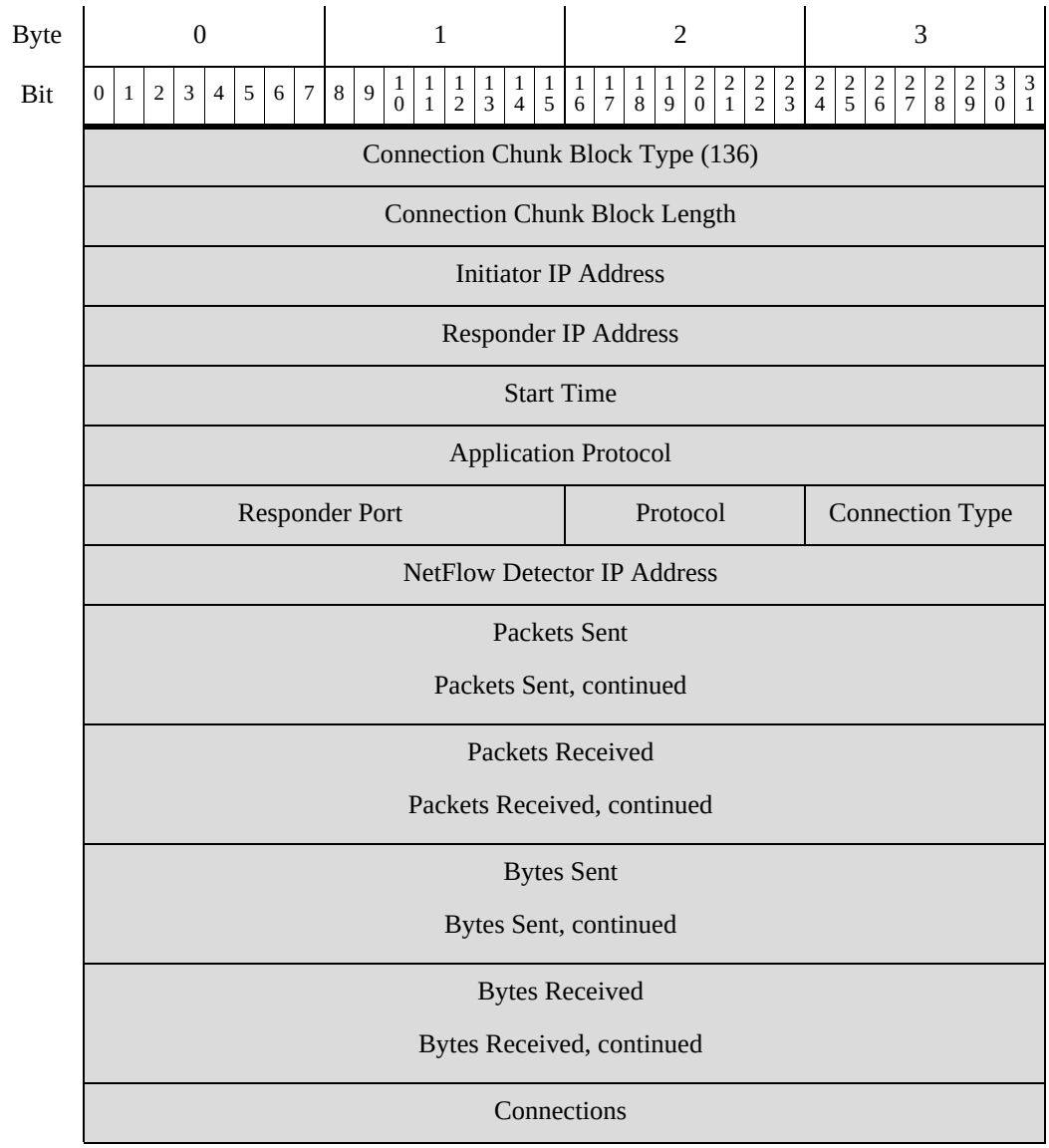
***Table B-36 Connection Chunk Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| Connection Chunk Block Type | uint32 | Initiates a Connection Chunk data block. This value is `66` for versions before 4.10.1 and a value of `119` for version 5.0. |
| Connection Chunk Block Length | uint32 | Total number of bytes in the Connection Chunk data block, including eight bytes for the connection chunk block type and length fields, plus the number of bytes in the connection chunk data that follows. |
| Initiator IP Address | uint8[4] | IP address of the host that initiated the connection, in IP address octets. |
| Responder IP Address | uint8[4] | IP address of the host responding in the connection, in IP address octets. |
| Start Time | uint32 | The starting time for the connection chunk. |
| Application ID | uint32 | Application identification number for the application protocol used in the connection. |
| Responder Port | uint16 | The port used by the responder in the connection chunk. |
| Protocol | uint8 | The protocol for the packet containing the user information. |
| Connection Type | uint8 | The type of connection. |
| Source Device IP Address | uint8[4] | IP address of the NetFlow device that detected the connection, in IP address octets. |
| Packets Sent | uint32 | The number of packets sent in the connection chunk. |
| Packets Received | uint32 | The number of packets received in the connection chunk. |
| Bytes Sent | uint32 | The number of bytes sent in the connection chunk. |
| Bytes Received | uint32 | The number of bytes received in the connection chunk. |
| Connections | uint32 | The number of sessions made in the connection chunk. |

# Connection Chunk Data Block for 5.1.1-6.0.x

The Connection Chunk data block conveys connection data. It stores connection log data that aggregates over a five-minute period. The Connection Chunk data block has a block type of 136 in the series 1 group of blocks. It supersedes block type 119.

The following diagram shows the format of the Connection Chunk data block:




The following table describes the components of the Connection Chunk data block.

***Table B-37 Connection Chunk Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| Connection Chunk Block Type | uint32 | Initiates a Connection Chunk data block. This value is always `136`. |
| Connection Chunk Block Length | uint32 | Total number of bytes in the Connection Chunk data block, including eight bytes for the connection chunk block type and length fields, plus the number of bytes in the connection chunk data that follows. |
| Initiator IP Address | uint8(4) | IP address of the initiator of this type of connection. This is used with the responder IP address to identify identical connections. |

*Table B-37 Connection Chunk Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Responder IP Address | uint8(4) | IP address of the responder to this type of connection. This is used with the initiator IP address to identify identical connections. |
| Start Time | uint32 | The starting time for the connection chunk. |
| Application Protocol | uint32 | Identification number for the protocol used in the connection. |
| Responder Port | uint16 | The port used by the responder in the connection chunk. |
| Protocol | uint8 | The protocol for the packet containing the user information. |
| Connection Type | uint8 | The type of connection. |
| NetFlow Detector IP Address | uint8[4] | IP address of the NetFlow device that detected the connection, in IP address octets. |
| Packets Sent | uint64 | The number of packets sent in the connection chunk. |
| Packets Received | uint64 | The number of packets received in the connection chunk. |
| Bytes Sent | uint64 | The number of bytes sent in the connection chunk. |
| Bytes Received | uint64 | The number of bytes received in the connection chunk. |
| Connections | uint32 | The number of connections over a five-minute period. |

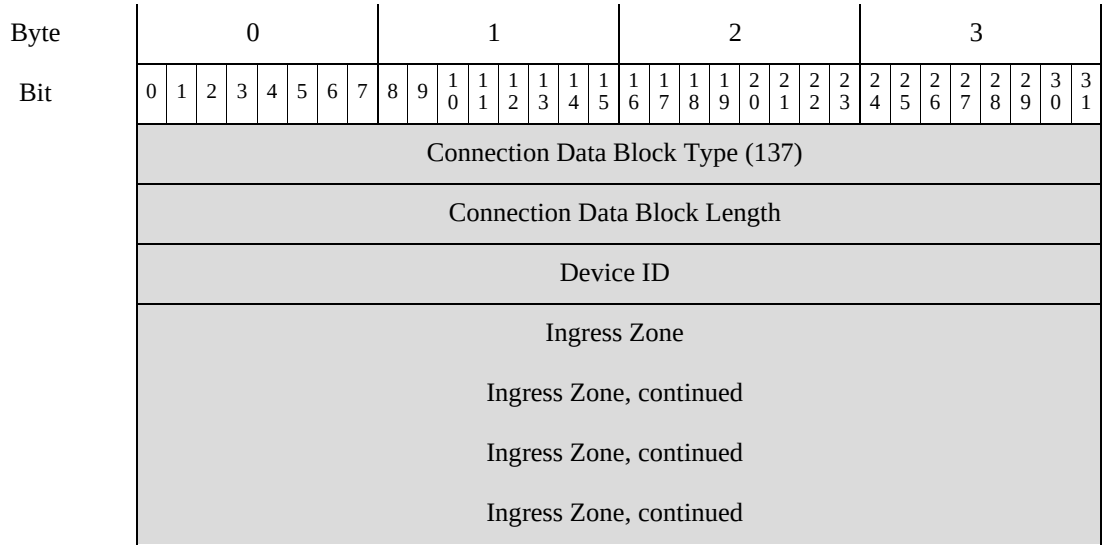
# Connection Statistics Data Block 5.1.1.x

The connection statistics data block is used in connection data messages. Changes to the connection data block between versions 5.1 and 5.1.1 include the addition of new fields to identify associated intrusion events. The connection statistics data block for version 5.1.1.x has a block type of 137. It deprecates block type 126, Connection Statistics Data Block 5.1, page B-163.

For more information on the Connection Statistics Data message, see Connection Statistics Data Message, page 4-53.

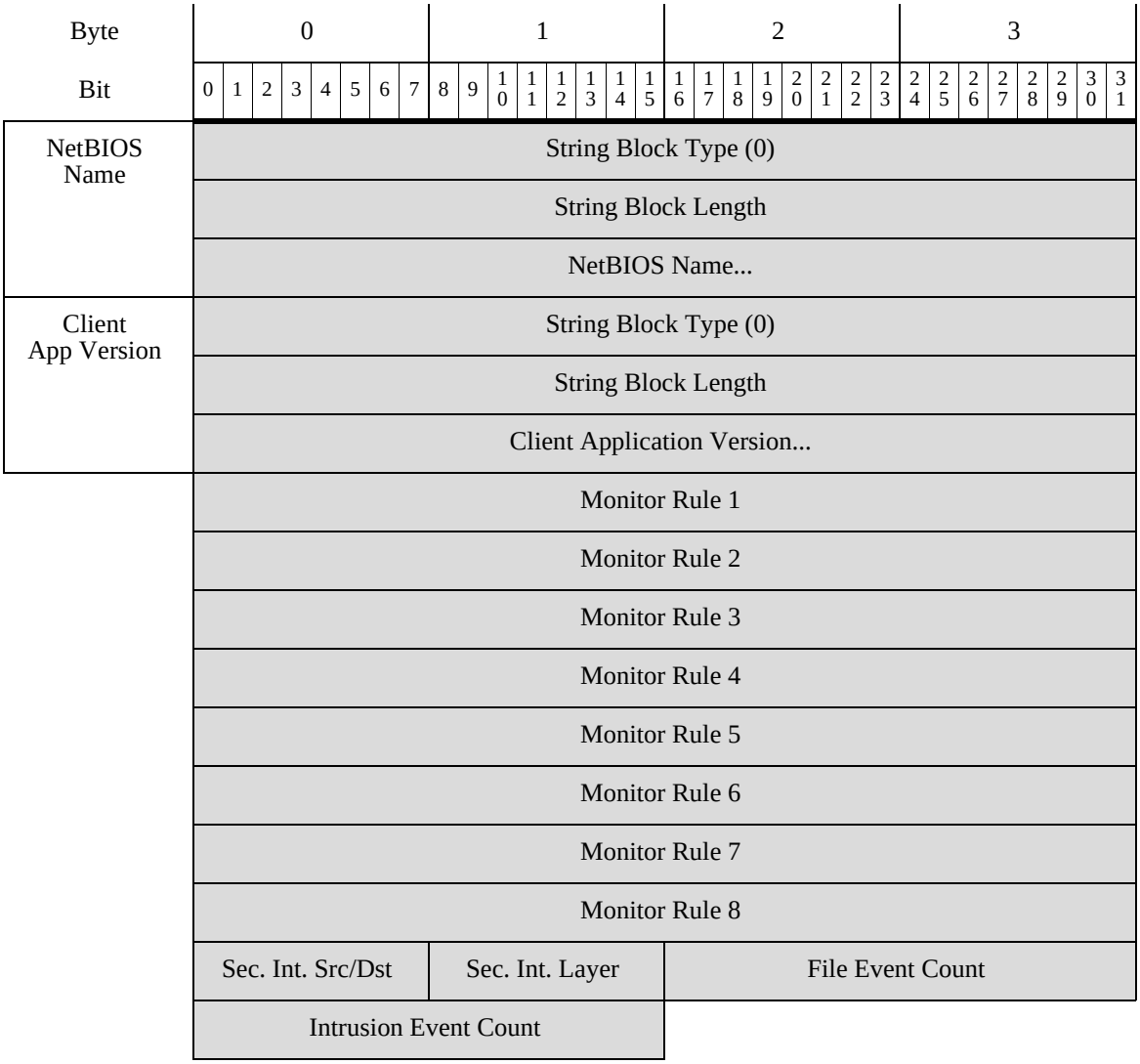
The following diagram shows the format of a Connection Statistics data block for 5.1.1:

::



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Egress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Action | | | | | | | | | | | | | | | | | Rule Reason | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Initiator Port | | | | | | | | | | | | | | | | Responder Port | | | | | | | | | | | | | | | |
| | TCP Flags | | | | | | | | | | | | | | | | Protocol | | | | | | | | NetFlow Source | | | | | | | |
| | NetFlow Source, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | NetFlow Source, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | NetFlow Source, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | NetFlow Source, continued | | | | | | | | | | | | | | | | | | | | | | | | Instance ID | | | | | | | |
| | Instance ID, cont. | | | | | | | | Connection Counter | | | | | | | | | | | | | | | | First Pkt Time | | | | | | | |
| | First Packet Timestamp, continued | | | | | | | | | | | | | | | | | | | | | | | | Last Pkt Time | | | | | | | |
| | Last Packet Timestamp, continued | | | | | | | | | | | | | | | | | | | | | | | | Initiator Tx Packets | | | | | | | |
| | Initiator Transmitted Packets, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Initiator Transmitted Packets, continued | | | | | | | | | | | | | | | | | | | | | | | | Resp. Tx Packets | | | | | | | |
| | Responder Transmitted Packets, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Responder Transmitted Packets, continued | | | | | | | | | | | | | | | | | | | | | | | | Initiator Tx Bytes | | | | | | | |
| | Initiator Transmitted Bytes, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Initiator Transmitted Bytes, continued | | | | | | | | | | | | | | | | | | | | | | | | Resp. Tx Bytes | | | | | | | |
| | Responder Transmitted Bytes, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Responder Transmitted Bytes, continued | | | | | | | | | | | | | | | | | | | | | | | | User ID | | | | | | | |
| | User ID, continued | | | | | | | | | | | | | | | | | | | | | | | | Application Prot. ID | | | | | | | |
| | Application Protocol ID, continued | | | | | | | | | | | | | | | | | | | | | | | | URL Category | | | | | | | |
| | URL Category, continued | | | | | | | | | | | | | | | | | | | | | | | | URL Reputation | | | | | | | |
| | URL Reputation, continued | | | | | | | | | | | | | | | | | | | | | | | | Client App ID | | | | | | | |
| | Client Application ID, continued | | | | | | | | | | | | | | | | | | | | | | | | Web App ID | | | | | | | |
| Client URL | Web Application ID, continued | | | | | | | | | | | | | | | | | | | | | | | | Str. Block Type (0) | | | | | | | |
| | String Block Type, continued | | | | | | | | | | | | | | | | | | | | | | | | String Block Length | | | | | | | |
| | String Block Length, continued | | | | | | | | | | | | | | | | | | | | | | | | Client App. URL... | | | | | | | |




The following table describes the fields of the Connection Statistics data block for 5.1.1.x.

***Table B-38 Connection Statistics Data Block 5.1.1.x Fields***

| Field | Data Type | Description |
|---|---|---|
| Connection Statistics Data Block Type | uint32 | Initiates a Connection Statistics data block for 5.1.1.x. The value is always `137`. |
| Connection Statistics Data Block Length | uint32 | Number of bytes in the Connection Statistics data block, including eight bytes for the connection statistics block type and length fields, plus the number of bytes in the connection data that follows. |
| Device ID | uint32 | The device that detected the connection event. |
| Ingress Zone | uint8[16] | Ingress security zone in the event that triggered the policy violation. |

*Table B-38 Connection Statistics Data Block 5.1.1.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Egress Zone | uint8[16] | Egress security zone in the event that triggered the policy violation. |
| Ingress Interface | uint8[16] | Interface for the inbound traffic. |
| Egress Interface | uint8[16] | Interface for the outbound traffic. |
| Initiator IP Address | uint8[16] | IP address of the host that initiated the session described in the connection event, in IP address octets. |
| Responder IP Address | uint8[16] | IP address of the host that responded to the initiating host, in IP address octets. |
| Policy Revision | uint8[16] | Revision number of the rule associated with the triggered correlation event, if applicable. |
| Rule ID | uint32 | Internal identifier for the rule that triggered the event, if applicable. |
| Rule Action | uint16 | The action selected in the user interface for that rule (allow, block, and so forth). |
| Rule Reason | uint16 | The reason the rule triggered the event. |
| Initiator Port | uint16 | Port used by the initiating host. |
| Responder Port | uint16 | Port used by the responding host. |
| TCP Flags | uint16 | Indicates any TCP flags for the connection event. |
| Protocol | uint8 | The IANA-specified protocol number. |
| NetFlow Source | uint8[16] | IP address of the NetFlow-enabled device that exported the data for the connection. |
| Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| First Packet Timestamp | uint32 | UNIX timestamp of the date and time the first packet was exchanged in the session. |
| Last Packet Timestamp | uint32 | UNIX timestamp of the date and time the last packet was exchanged in the session. |
| Initiator Transmitted Packets | uint64 | Number of packets transmitted by the initiating host. |
| Responder Transmitted Packets | uint64 | Number of packets transmitted by the responding host. |
| Initiator Transmitted Bytes | uint64 | Number of bytes transmitted by the initiating host. |
| Responder Transmitted Bytes | uint64 | Number of bytes transmitted by the responding host. |
| User ID | uint32 | Internal identification number for the user who last logged into the host that generated the traffic. |

*Table B-38 Connection Statistics Data Block 5.1.1.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Application Protocol ID | uint32 | Application ID of the application protocol. |
| URL Category | uint32 | The internal identification number of the URL category. |
| URL Reputation | uint32 | The internal identification number for the URL reputation. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| String Block Type | uint32 | Initiates a String data block for the client application URL. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the client application URL String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the client application URL string. |
| Client Application URL | string | URL the client application accessed, if applicable (/files/index.html, for example). |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for the client application version. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the String data block for the client application version, including eight bytes for the string block type and length, plus the number of bytes in the version. |
| Client Application Version | string | Client application version. |
| Monitor Rule 1 | uint32 | The ID of the first monitor rule associated with the connection event. |
| Monitor Rule 2 | uint32 | The ID of the second monitor rule associated with the connection event. |
| Monitor Rule 3 | uint32 | The ID of the third monitor rule associated with the connection event. |
| Monitor Rule 4 | uint32 | The ID of the fourth monitor rule associated with the connection event. |
| Monitor Rule 5 | uint32 | The ID of the fifth monitor rule associated with the connection event. |
| Monitor Rule 6 | uint32 | The ID of the sixth monitor rule associated with the connection event. |

*Table B-38 Connection Statistics Data Block 5.1.1.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Monitor Rule 7 | uint32 | The ID of the seventh monitor rule associated with the connection event. |
| Monitor Rule 8 | uint32 | The ID of the eighth monitor rule associated with the connection event. |
| Security Intelligence Source/ Destination | uint8 | Whether the source or destination IP address matched the IP block list. |
| Security Intelligence Layer | uint8 | The IP layer that matched the IP block list. |
| File Event Count | uint16 | Value used to distinguish between file events that happen during the same second. |
| Intrusion Event Count | uint16 | Value used to distinguish between intrusion events that happen during the same second. |

# Connection Statistics Data Block 5.3

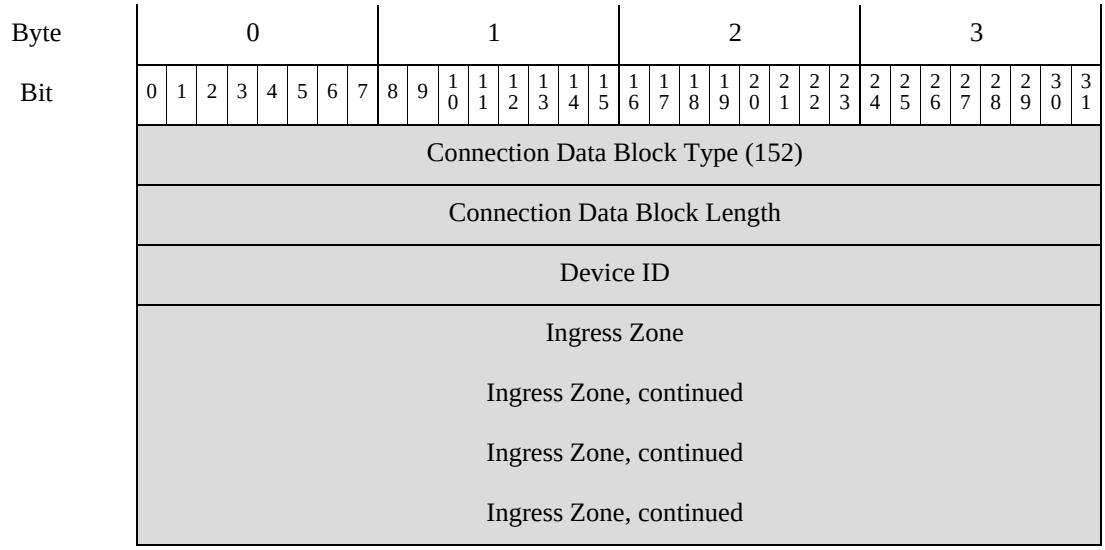
The connection statistics data block is used in connection data messages. Changes to the connection data block between versions 5.2.x and 5.3 include the addition of new fields for NetFlow information. The connection statistics data block for version 5.3 has a block type of 152 in the series 1 group of blocks. It deprecates block type 144, Connection Statistics Data Block 5.2.x, page B-169.

You request connection event records by setting the extended event flag—bit 30 in the Request Flags field—in the request message with an event version of 10 and an event code of 71. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.

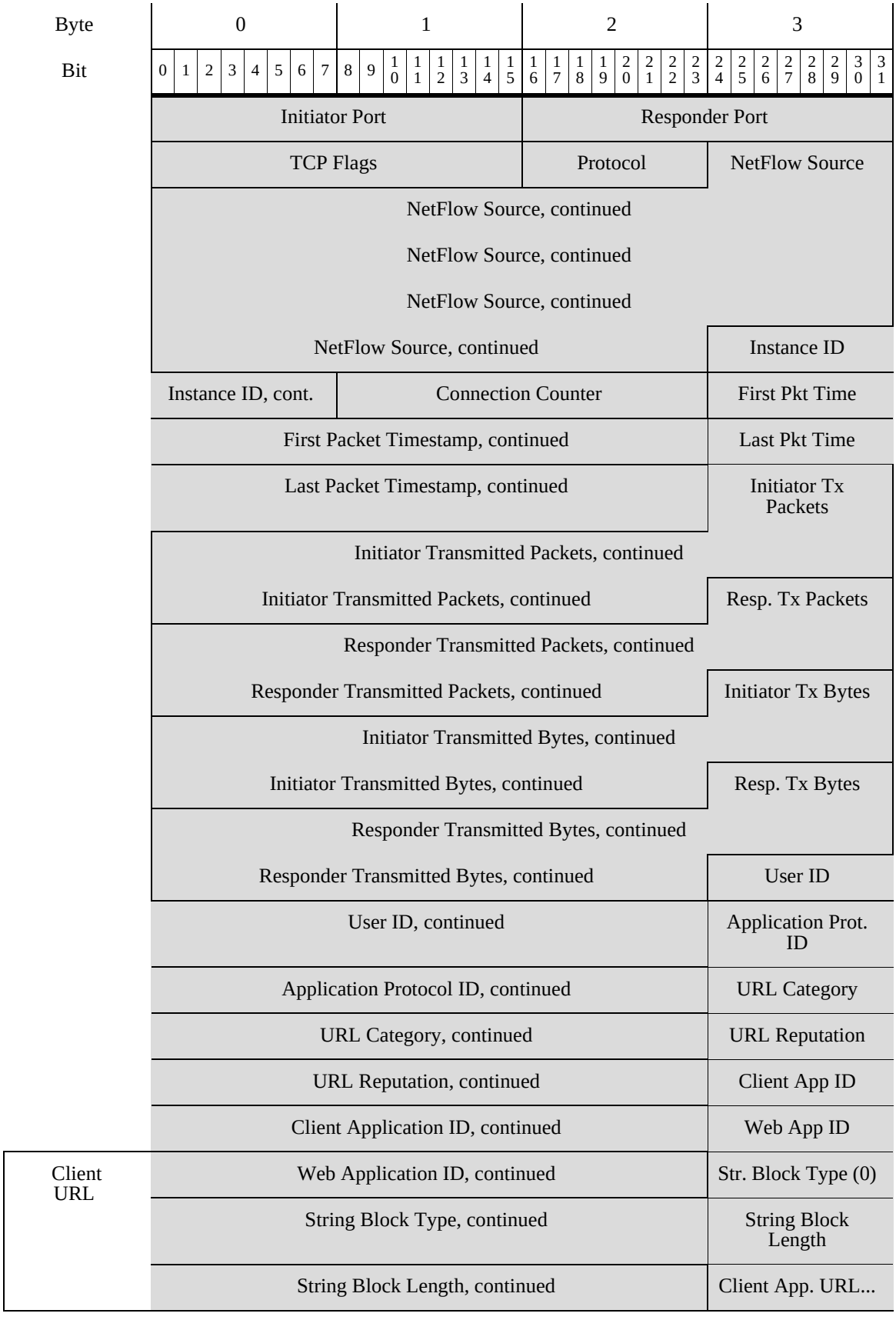
For more information on the Connection Statistics Data message, see Connection Statistics Data Message, page 4-53.

The following diagram shows the format of a Connection Statistics data block for 5.3+:

::



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Egress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Action | | | | | | | | | | | | | | | | | Rule Reason | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Initiator Port
Responder Port
TCP Flags
Protocol
NetFlow Source
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
Instance ID
Instance ID, cont.
Connection Counter
First Pkt Time
First Packet Timestamp, continued
Last Pkt Time
Last Packet Timestamp, continued
Initiator Tx Packets
Initiator Transmitted Packets, continued
Initiator Transmitted Packets, continued
Resp. Tx Packets
Responder Transmitted Packets, continued
Responder Transmitted Packets, continued
Initiator Tx Bytes
Initiator Transmitted Bytes, continued
Initiator Transmitted Bytes, continued
Resp. Tx Bytes
Responder Transmitted Bytes, continued
Responder Transmitted Bytes, continued
User ID
User ID, continued
Application Prot. ID
Application Protocol ID, continued
URL Category
URL Category, continued
URL Reputation
URL Reputation, continued
Client App ID
Client Application ID, continued
Web App ID
Client URL
Web Application ID, continued
Str. Block Type (0)
String Block Type, continued
String Block Length
String Block Length, continued
Client App. URL...

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| NetBIOS Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | NetBIOS Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client App Version | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Client Application Version... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Sec. Int. Src/Dst | | | | | | | | Sec. Int. Layer | | | | | | | | File Event Count | | | | | | | | | | | | | | | |
| | Intrusion Event Count | | | | | | | | | | | | | | | | Initiator Country | | | | | | | | | | | | | | | |
| | Responder Country | | | | | | | | | | | | | | | | IOC Number | | | | | | | | | | | | | | | |
| | Source Autonomous System | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Destination Autonomous System | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SNMP In | | | | | | | | | | | | | | | | SNMP Out | | | | | | | | | | | | | | | |
| | Source TOS | | | | | | | | Destination TOS | | | | | | | | Source Mask | | | | | | | | Destination Mask | | | | | | | |

The following table describes the fields of the Connection Statistics data block for 5.3.

*Table B-39 Connection Statistics Data Block 5.3+ Fields*

| Field | Data Type | Description |
|---|---|---|
| Connection Statistics Data Block Type | uint32 | Initiates a Connection Statistics data block for 5.3. The value is always `152`. |
| Connection Statistics Data Block Length | uint32 | Number of bytes in the Connection Statistics data block, including eight bytes for the connection statistics block type and length fields, plus the number of bytes in the connection data that follows. |
| Device ID | uint32 | The device that detected the connection event. |
| Ingress Zone | uint8[16] | Ingress security zone in the event that triggered the policy violation. |
| Egress Zone | uint8[16] | Egress security zone in the event that triggered the policy violation. |
| Ingress Interface | uint8[16] | Interface for the inbound traffic. |
| Egress Interface | uint8[16] | Interface for the outbound traffic. |
| Initiator IP Address | uint8[16] | IP address of the host that initiated the session described in the connection event, in IP address octets. |
| Responder IP Address | uint8[16] | IP address of the host that responded to the initiating host, in IP address octets. |
| Policy Revision | uint8[16] | Revision number of the rule associated with the triggered correlation event, if applicable. |
| Rule ID | uint32 | Internal identifier for the rule that triggered the event, if applicable. |
| Rule Action | uint16 | The action selected in the user interface for that rule (allow, block, and so forth). |
| Rule Reason | uint16 | The reason the rule triggered the event. |
| Initiator Port | uint16 | Port used by the initiating host. |
| Responder Port | uint16 | Port used by the responding host. |
| TCP Flags | uint16 | Indicates any TCP flags for the connection event. |
| Protocol | uint8 | The IANA-specified protocol number. |
| NetFlow Source | uint8[16] | IP address of the NetFlow-enabled device that exported the data for the connection. |
| Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| First Packet Timestamp | uint32 | UNIX timestamp of the date and time the first packet was exchanged in the session. |
| Last Packet Timestamp | uint32 | UNIX timestamp of the date and time the last packet was exchanged in the session. |

*Table B-39 Connection Statistics Data Block 5.3+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Initiator Transmitted Packets | uint64 | Number of packets transmitted by the initiating host. |
| Responder Transmitted Packets | uint64 | Number of packets transmitted by the responding host. |
| Initiator Transmitted Bytes | uint64 | Number of bytes transmitted by the initiating host. |
| Responder Transmitted Bytes | uint64 | Number of bytes transmitted by the responding host. |
| User ID | uint32 | Internal identification number for the user who last logged into the host that generated the traffic. |
| Application Protocol ID | uint32 | Application ID of the application protocol. |
| URL Category | uint32 | The internal identification number of the URL category. |
| URL Reputation | uint32 | The internal identification number for the URL reputation. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| String Block Type | uint32 | Initiates a String data block for the client application URL. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the client application URL String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the client application URL string. |
| Client Application URL | string | URL the client application accessed, if applicable (`/files/index.html`, for example). |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for the client application version. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block for the client application version, including eight bytes for the string block type and length, plus the number of bytes in the version. |
| Client Application Version | string | Client application version. |

*Table B-39 Connection Statistics Data Block 5.3+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Monitor Rule 1 | uint32 | The ID of the first monitor rule associated with the connection event. |
| Monitor Rule 2 | uint32 | The ID of the second monitor rule associated with the connection event. |
| Monitor Rule 3 | uint32 | The ID of the third monitor rule associated with the connection event. |
| Monitor Rule 4 | uint32 | The ID of the fourth monitor rule associated with the connection event. |
| Monitor Rule 5 | uint32 | The ID of the fifth monitor rule associated with the connection event. |
| Monitor Rule 6 | uint32 | The ID of the sixth monitor rule associated with the connection event. |
| Monitor Rule 7 | uint32 | The ID of the seventh monitor rule associated with the connection event. |
| Monitor Rule 8 | uint32 | The ID of the eighth monitor rule associated with the connection event. |
| Security Intelligence Source/ Destination | uint8 | Whether the source or destination IP address matched the IP block list. |
| Security Intelligence Layer | uint8 | The IP layer that matched the IP block list. |
| File Event Count | uint16 | Value used to distinguish between file events that happen during the same second. |
| Intrusion Event Count | uint16 | Value used to distinguish between intrusion events that happen during the same second. |
| Initiator Country | uint16 | Code for the country of the initiating host. |
| Responder Country | uint 16 | Code for the country of the responding host. |
| IOC Number | uint16 | ID Number of the compromise associated with this event. |
| Source Autonomous System | uint32 | Autonomous system number of the source, either origin or peer. |
| Destination Autonomous System | uint32 | Autonomous system number of the destination, either origin or peer. |
| SNMP Input | uint16 | SNMP index of the input interface. |
| SNMP Output | uint16 | SNMP index of the output interface. |
| Source TOS | uint8 | Type of Service byte setting for the incoming interface. |
| Destination TOS | uint8 | Type of Service byte setting for the outgoing interface. |

*Table B-39 Connection Statistics Data Block 5.3+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Source Mask | uint8 | Source address prefix mask. |
| Destination Mask | uint8 | Destination address prefix mask. |

# Connection Statistics Data Block 5.3.1

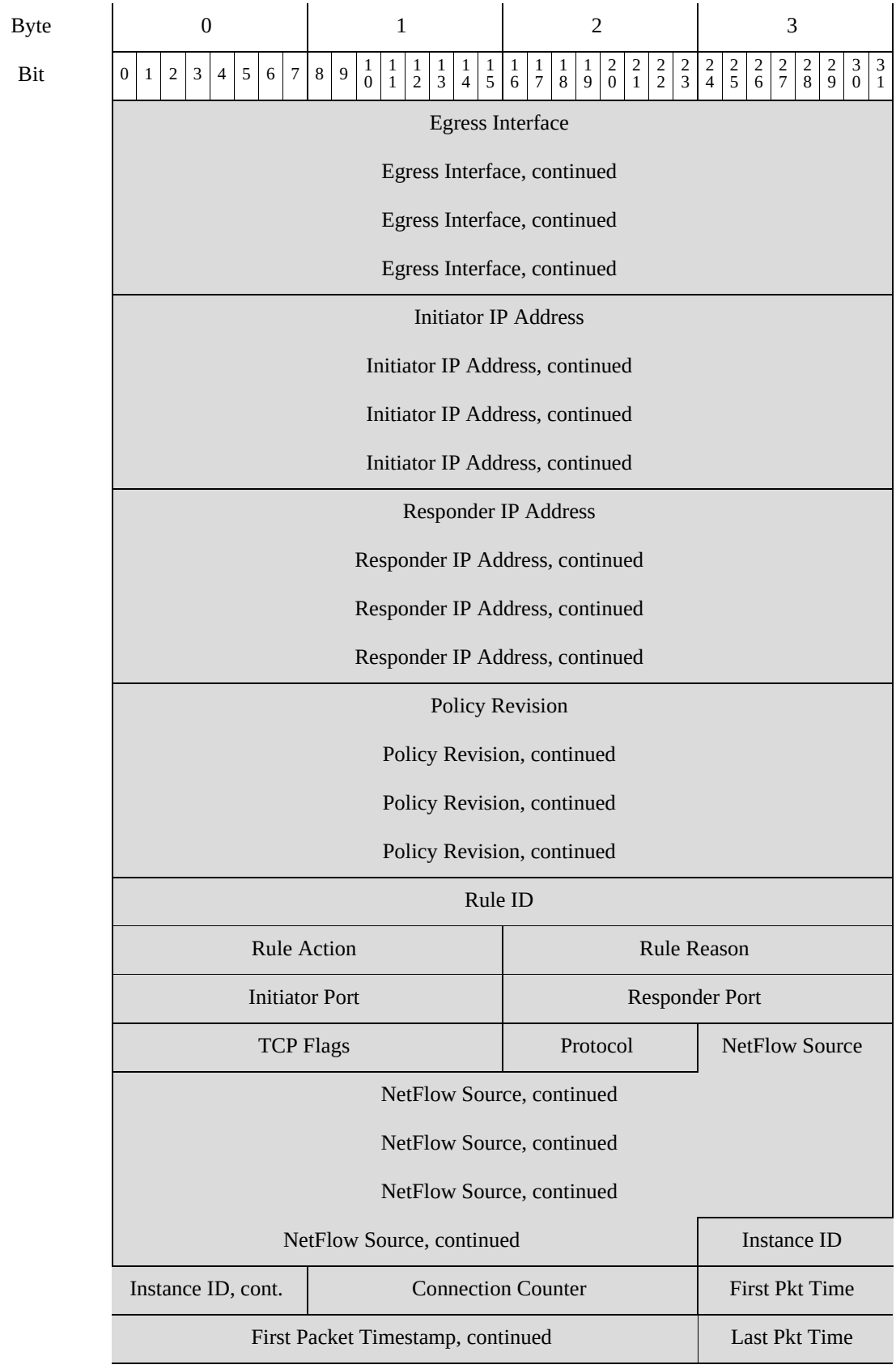
The connection statistics data block is used in connection data messages. The only changes to the connection data block between versions 5.3 and 5.3.1 is the addition of a security context field. The connection statistics data block for version 5.3.1 has a block type of 154 in the series 1 group of blocks. It deprecates block type 152, Connection Statistics Data Block 5.3, page B-184.

You request connection event records by setting the extended event flag—bit 30 in the Request Flags field—in the request message with an event version of 11 and an event code of 71. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.For more information on the Connection Statistics Data message, see Connection Statistics Data Message, page 4-53.

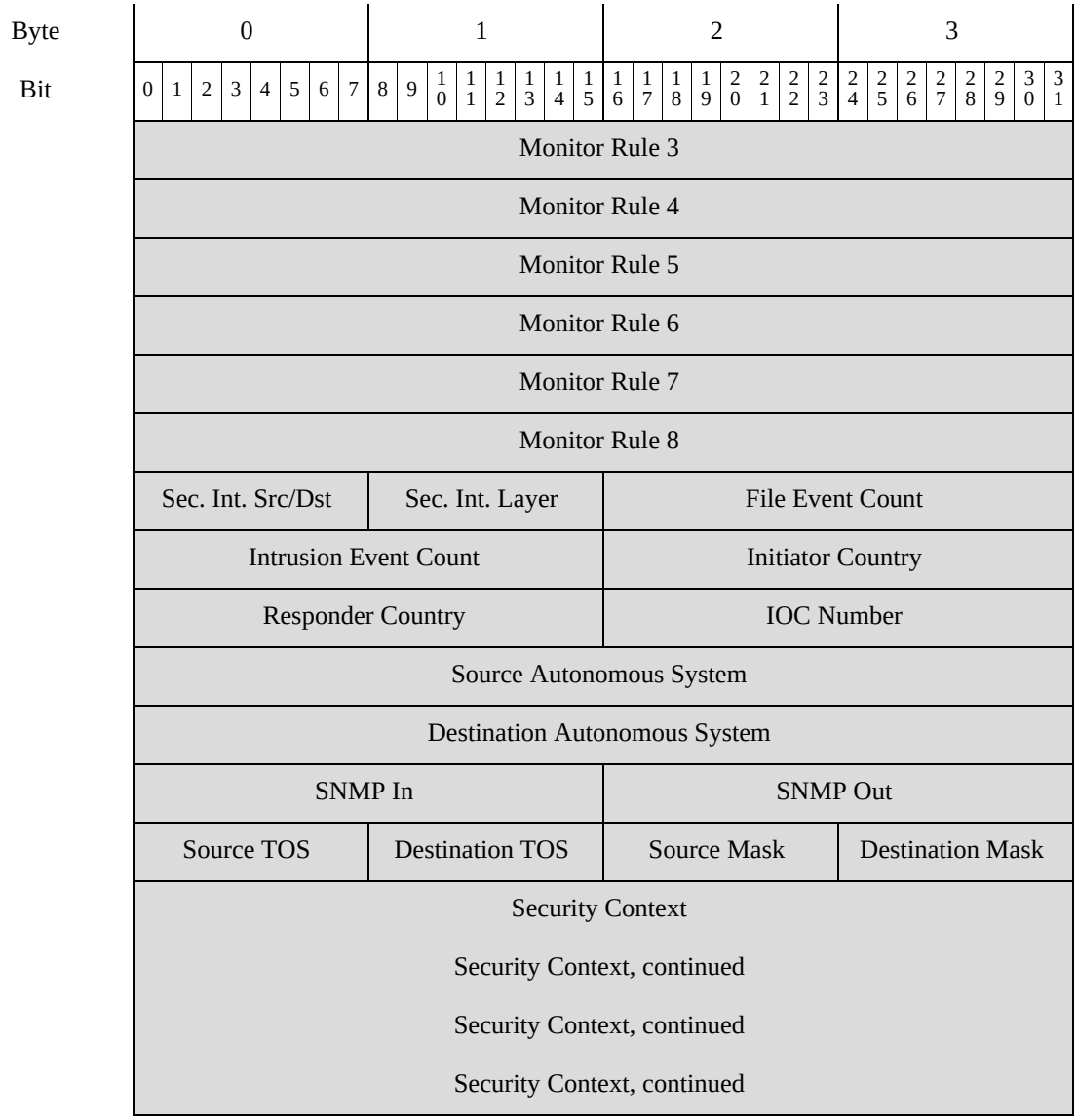
The following diagram shows the format of a Connection Statistics data block for 5.3.1:

::

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Connection Data Block Type (154) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Connection Data Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Egress Interface
Egress Interface, continued
Egress Interface, continued
Egress Interface, continued
Initiator IP Address
Initiator IP Address, continued
Initiator IP Address, continued
Initiator IP Address, continued
Responder IP Address
Responder IP Address, continued
Responder IP Address, continued
Responder IP Address, continued
Policy Revision
Policy Revision, continued
Policy Revision, continued
Policy Revision, continued
Rule ID
Rule Action
Rule Reason
Initiator Port
Responder Port
TCP Flags
Protocol
NetFlow Source
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
Instance ID
Instance ID, cont.
Connection Counter
First Pkt Time
First Packet Timestamp, continued
Last Pkt Time

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Last Packet Timestamp, continued | | | | | | | | | | | | | | | | | | | | | | | | Initiator Tx Packets | | | | | | | |
| | Initiator Transmitted Packets, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Initiator Transmitted Packets, continued | | | | | | | | | | | | | | | | | | | | | | | | Resp. Tx Packets | | | | | | | |
| | Responder Transmitted Packets, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Responder Transmitted Packets, continued | | | | | | | | | | | | | | | | | | | | | | | | Initiator Tx Bytes | | | | | | | |
| | Initiator Transmitted Bytes, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Initiator Transmitted Bytes, continued | | | | | | | | | | | | | | | | | | | | | | | | Resp. Tx Bytes | | | | | | | |
| | Responder Transmitted Bytes, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Responder Transmitted Bytes, continued | | | | | | | | | | | | | | | | | | | | | | | | User ID | | | | | | | |
| | User ID, continued | | | | | | | | | | | | | | | | | | | | | | | | Application Prot. ID | | | | | | | |
| | Application Protocol ID, continued | | | | | | | | | | | | | | | | | | | | | | | | URL Category | | | | | | | |
| | URL Category, continued | | | | | | | | | | | | | | | | | | | | | | | | URL Reputation | | | | | | | |
| | URL Reputation, continued | | | | | | | | | | | | | | | | | | | | | | | | Client App ID | | | | | | | |
| | Client Application ID, continued | | | | | | | | | | | | | | | | | | | | | | | | Web App ID | | | | | | | |
| Client URL | Web Application ID, continued | | | | | | | | | | | | | | | | | | | | | | | | Str. Block Type (0) | | | | | | | |
| | String Block Type, continued | | | | | | | | | | | | | | | | | | | | | | | | String Block Length | | | | | | | |
| | String Block Length, continued | | | | | | | | | | | | | | | | | | | | | | | | Client App. URL... | | | | | | | |
| NetBIOS Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | NetBIOS Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client App Version | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Client Application Version... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |




The following table describes the fields of the Connection Statistics data block for 5.3.1.

***Table B-40 Connection Statistics Data Block 5.3.1 Fields***

| Field | Data Type | Description |
|---|---|---|
| Connection Statistics Data Block Type | uint32 | Initiates a Connection Statistics data block for 5.3.1+. The value is always 154. |
| Connection Statistics Data Block Length | uint32 | Number of bytes in the Connection Statistics data block, including eight bytes for the connection statistics block type and length fields, plus the number of bytes in the connection data that follows. |
| Device ID | uint32 | The device that detected the connection event. |

*Table B-40 Connection Statistics Data Block 5.3.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Ingress Zone | uint8[16] | Ingress security zone in the event that triggered the policy violation. |
| Egress Zone | uint8[16] | Egress security zone in the event that triggered the policy violation. |
| Ingress Interface | uint8[16] | Interface for the inbound traffic. |
| Egress Interface | uint8[16] | Interface for the outbound traffic. |
| Initiator IP Address | uint8[16] | IP address of the host that initiated the session described in the connection event, in IP address octets. |
| Responder IP Address | uint8[16] | IP address of the host that responded to the initiating host, in IP address octets. |
| Policy Revision | uint8[16] | Revision number of the rule associated with the triggered correlation event, if applicable. |
| Rule ID | uint32 | Internal identifier for the rule that triggered the event, if applicable. |
| Rule Action | uint16 | The action selected in the user interface for that rule (allow, block, and so forth). |
| Rule Reason | uint16 | The reason the rule triggered the event. |
| Initiator Port | uint16 | Port used by the initiating host. |
| Responder Port | uint16 | Port used by the responding host. |
| TCP Flags | uint16 | Indicates any TCP flags for the connection event. |
| Protocol | uint8 | The IANA-specified protocol number. |
| NetFlow Source | uint8[16] | IP address of the NetFlow-enabled device that exported the data for the connection. |
| Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| First Packet Timestamp | uint32 | UNIX timestamp of the date and time the first packet was exchanged in the session. |
| Last Packet Timestamp | uint32 | UNIX timestamp of the date and time the last packet was exchanged in the session. |
| Initiator Transmitted Packets | uint64 | Number of packets transmitted by the initiating host. |
| Responder Transmitted Packets | uint64 | Number of packets transmitted by the responding host. |
| Initiator Transmitted Bytes | uint64 | Number of bytes transmitted by the initiating host. |
| Responder Transmitted Bytes | uint64 | Number of bytes transmitted by the responding host. |

*Table B-40 Connection Statistics Data Block 5.3.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| User ID | uint32 | Internal identification number for the user who last logged into the host that generated the traffic. |
| Application Protocol ID | uint32 | Application ID of the application protocol. |
| URL Category | uint32 | The internal identification number of the URL category. |
| URL Reputation | uint32 | The internal identification number for the URL reputation. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| String Block Type | uint32 | Initiates a String data block for the client application URL. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the client application URL String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the client application URL string. |
| Client Application URL | string | URL the client application accessed, if applicable (/files/index.html, for example). |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for the client application version. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the String data block for the client application version, including eight bytes for the string block type and length, plus the number of bytes in the version. |
| Client Application Version | string | Client application version. |
| Monitor Rule 1 | uint32 | The ID of the first monitor rule associated with the connection event. |
| Monitor Rule 2 | uint32 | The ID of the second monitor rule associated with the connection event. |
| Monitor Rule 3 | uint32 | The ID of the third monitor rule associated with the connection event. |
| Monitor Rule 4 | uint32 | The ID of the fourth monitor rule associated with the connection event. |
| Monitor Rule 5 | uint32 | The ID of the fifth monitor rule associated with the connection event. |

*Table B-40 Connection Statistics Data Block 5.3.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Monitor Rule 6 | uint32 | The ID of the sixth monitor rule associated with the connection event. |
| Monitor Rule 7 | uint32 | The ID of the seventh monitor rule associated with the connection event. |
| Monitor Rule 8 | uint32 | The ID of the eighth monitor rule associated with the connection event. |
| Security Intelligence Source/ Destination | uint8 | Whether the source or destination IP address matched the IP block list. |
| Security Intelligence Layer | uint8 | The IP layer that matched the IP block list. |
| File Event Count | uint16 | Value used to distinguish between file events that happen during the same second. |
| Intrusion Event Count | uint16 | Value used to distinguish between intrusion events that happen during the same second. |
| Initiator Country | uint16 | Code for the country of the initiating host. |
| Responder Country | uint 16 | Code for the country of the responding host. |
| IOC Number | uint16 | ID Number of the compromise associated with this event. |
| Source Autonomous System | uint32 | Autonomous system number of the source, either origin or peer. |
| Destination Autonomous System | uint32 | Autonomous system number of the destination, either origin or peer. |
| SNMP Input | uint16 | SNMP index of the input interface. |
| SNMP Output | uint16 | SNMP index of the output interface. |
| Source TOS | uint8 | Type of Service byte setting for the incoming interface. |
| Destination TOS | uint8 | Type of Service byte setting for the outgoing interface. |
| Source Mask | uint8 | Source address prefix mask. |
| Destination Mask | uint8 | Destination address prefix mask. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |

# Connection Statistics Data Block 5.4

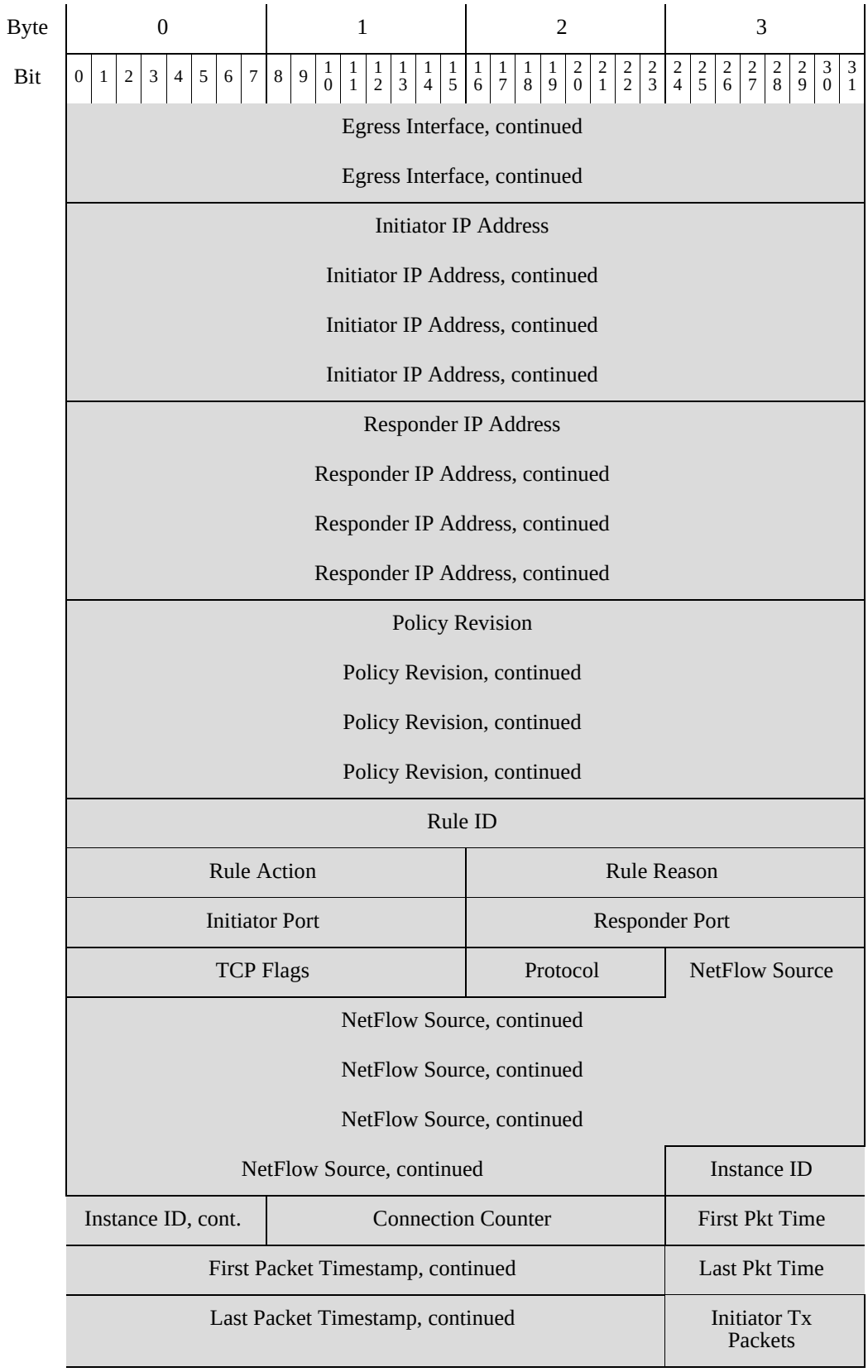
The connection statistics data block is used in connection data messages. Several new fields have been added to the Connection Statistics Data Block for 5.4. Fields have been added to support SSL connections, HTTP redirection, and network analysis policies. The connection statistics data block for version 5.4 has a block type of 155 in the series 1 group of blocks. It deprecates block type 154, Connection Statistics Data Block 5.3.1, page B-191.

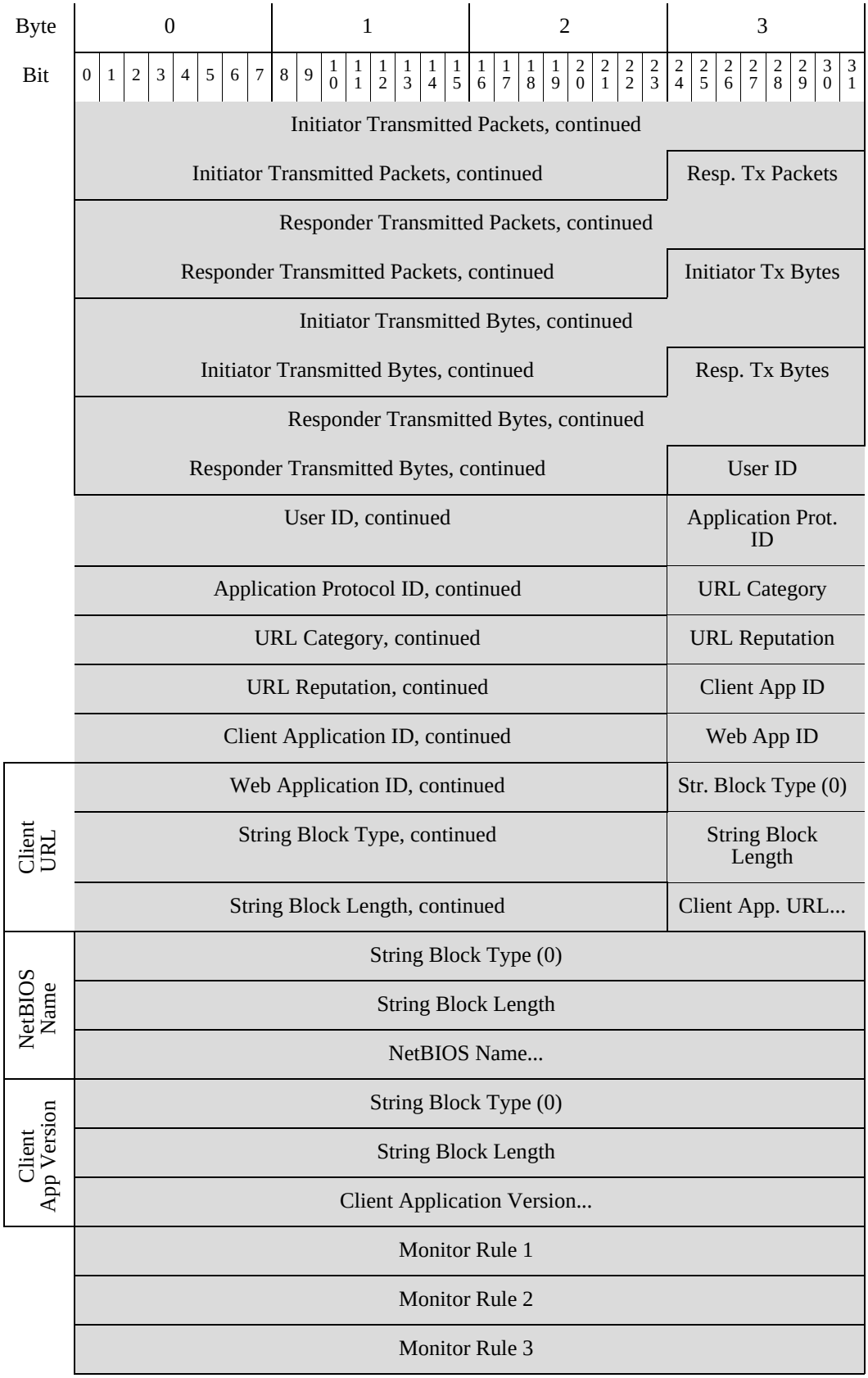
You request connection event records by setting the extended event flag—bit 30 in the Request Flags field—in the request message with an event version of 12 and an event code of 71. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.

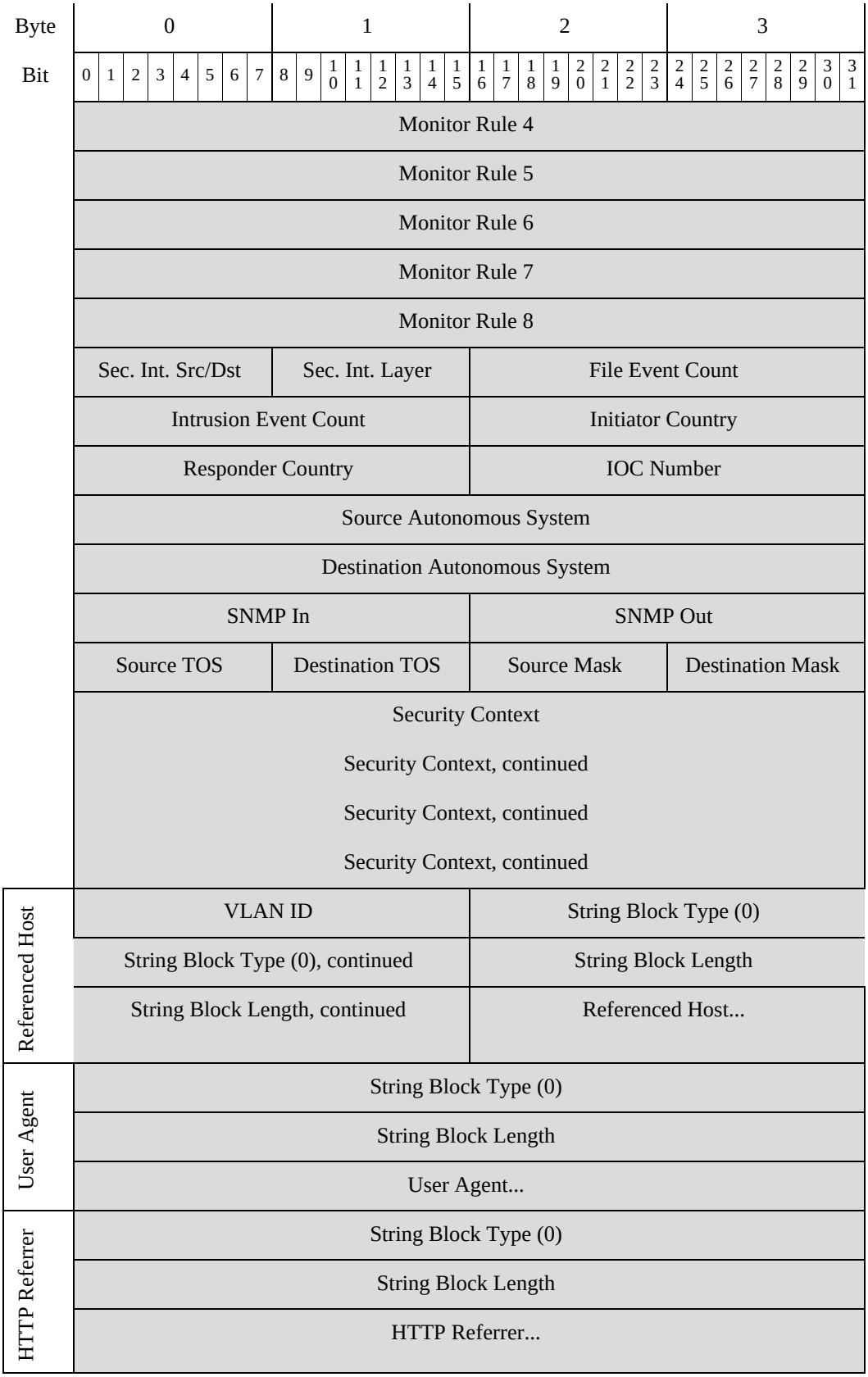
For more information on the Connection Statistics Data message, see Connection Statistics Data Message, page 4-53.

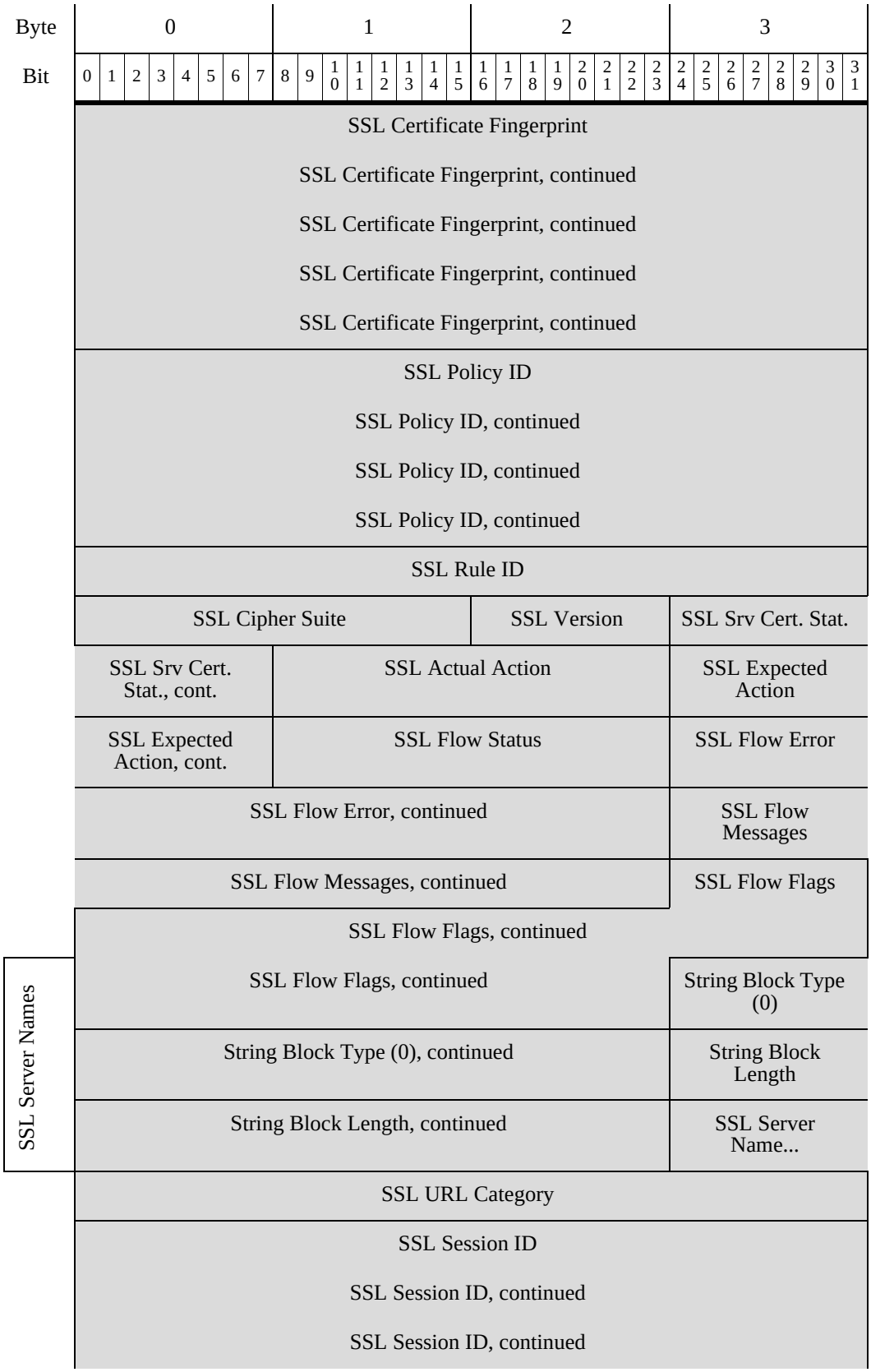
The following diagram shows the format of a Connection Statistics data block for 5.4:

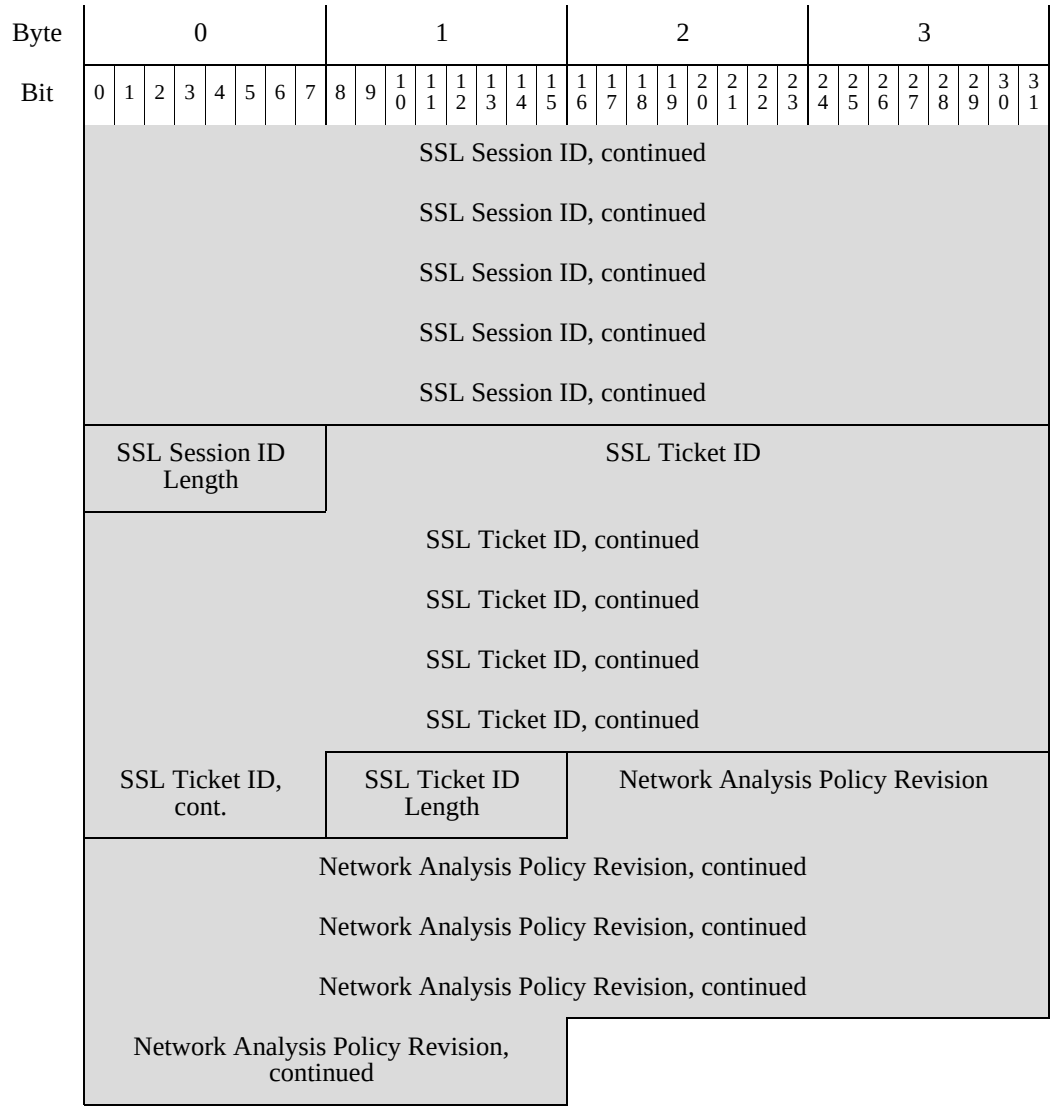
| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Connection Data Block Type (155) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Connection Data Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Egress Interface, continued
Egress Interface, continued
Initiator IP Address
Initiator IP Address, continued
Initiator IP Address, continued
Initiator IP Address, continued
Responder IP Address
Responder IP Address, continued
Responder IP Address, continued
Responder IP Address, continued
Policy Revision
Policy Revision, continued
Policy Revision, continued
Policy Revision, continued
Rule ID
Rule Action
Rule Reason
Initiator Port
Responder Port
TCP Flags
Protocol
NetFlow Source
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
Instance ID
Instance ID, cont.
Connection Counter
First Pkt Time
First Packet Timestamp, continued
Last Pkt Time
Last Packet Timestamp, continued
Initiator Tx Packets


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Initiator Transmitted Packets, continued
Initiator Transmitted Packets, continued
Resp. Tx Packets
Responder Transmitted Packets, continued
Responder Transmitted Packets, continued
Initiator Tx Bytes
Initiator Transmitted Bytes, continued
Initiator Transmitted Bytes, continued
Resp. Tx Bytes
Responder Transmitted Bytes, continued
Responder Transmitted Bytes, continued
User ID
User ID, continued
Application Prot. ID
Application Protocol ID, continued
URL Category
URL Category, continued
URL Reputation
URL Reputation, continued
Client App ID
Client Application ID, continued
Web App ID
Client URL
Web Application ID, continued
Str. Block Type (0)
String Block Type, continued
String Block Length
String Block Length, continued
Client App. URL...
NetBIOS Name
String Block Type (0)
String Block Length
NetBIOS Name...
Client App Version
String Block Type (0)
String Block Length
Client Application Version...
Monitor Rule 1
Monitor Rule 2
Monitor Rule 3


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Monitor Rule 4
Monitor Rule 5
Monitor Rule 6
Monitor Rule 7
Monitor Rule 8
Sec. Int. Src/Dst
Sec. Int. Layer
File Event Count
Intrusion Event Count
Initiator Country
Responder Country
IOC Number
Source Autonomous System
Destination Autonomous System
SNMP In
SNMP Out
Source TOS
Destination TOS
Source Mask
Destination Mask
Security Context
Security Context, continued
Security Context, continued
Security Context, continued
Referenced Host
VLAN ID
String Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
Referenced Host...
User Agent
String Block Type (0)
String Block Length
User Agent...
HTTP Referrer
String Block Type (0)
String Block Length
HTTP Referrer...


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
SSL Certificate Fingerprint
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Policy ID
SSL Policy ID, continued
SSL Policy ID, continued
SSL Policy ID, continued
SSL Rule ID
SSL Cipher Suite
SSL Version
SSL Srv Cert. Stat.
SSL Srv Cert. Stat., cont.
SSL Actual Action
SSL Expected Action
SSL Expected Action, cont.
SSL Flow Status
SSL Flow Error
SSL Flow Error, continued
SSL Flow Messages
SSL Flow Messages, continued
SSL Flow Flags
SSL Flow Flags, continued
SSL Flow Flags, continued
String Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
SSL Server Name...
SSL Server Names
SSL URL Category
SSL Session ID
SSL Session ID, continued
SSL Session ID, continued




The following table describes the fields of the Connection Statistics data block for 5.4+.

***Table B-41 Connection Statistics Data Block 5.4+ Fields***

| Field | Data Type | Description |
|---|---|---|
| Connection Statistics Data Block Type | uint32 | Initiates a Connection Statistics data block for 5.4+. The value is always `155`. |
| Connection Statistics Data Block Length | uint32 | Number of bytes in the Connection Statistics data block, including eight bytes for the connection statistics block type and length fields, plus the number of bytes in the connection data that follows. |
| Device ID | uint32 | The device that detected the connection event. |

*Table B-41 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Ingress Zone | uint8[16] | Ingress security zone in the event that triggered the policy violation. |
| Egress Zone | uint8[16] | Egress security zone in the event that triggered the policy violation. |
| Ingress Interface | uint8[16] | Interface for the inbound traffic. |
| Egress Interface | uint8[16] | Interface for the outbound traffic. |
| Initiator IP Address | uint8[16] | IP address of the host that initiated the session described in the connection event, in IP address octets. |
| Responder IP Address | uint8[16] | IP address of the host that responded to the initiating host, in IP address octets. |
| Policy Revision | uint8[16] | Revision number of the rule associated with the triggered correlation event, if applicable. |
| Rule ID | uint32 | Internal identifier for the rule that triggered the event, if applicable. |
| Rule Action | uint16 | The action selected in the user interface for that rule (allow, block, and so forth). |
| Rule Reason | uint16 | The reason the rule triggered the event. |
| Initiator Port | uint16 | Port used by the initiating host. |
| Responder Port | uint16 | Port used by the responding host. |
| TCP Flags | uint16 | Indicates any TCP flags for the connection event. |
| Protocol | uint8 | The IANA-specified protocol number. |
| NetFlow Source | uint8[16] | IP address of the NetFlow-enabled device that exported the data for the connection. |
| Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| First Packet Timestamp | uint32 | UNIX timestamp of the date and time the first packet was exchanged in the session. |
| Last Packet Timestamp | uint32 | UNIX timestamp of the date and time the last packet was exchanged in the session. |
| Initiator Transmitted Packets | uint64 | Number of packets transmitted by the initiating host. |
| Responder Transmitted Packets | uint64 | Number of packets transmitted by the responding host. |
| Initiator Transmitted Bytes | uint64 | Number of bytes transmitted by the initiating host. |
| Responder Transmitted Bytes | uint64 | Number of bytes transmitted by the responding host. |

*Table B-41 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| User ID | uint32 | Internal identification number for the user who last logged into the host that generated the traffic. |
| Application Protocol ID | uint32 | Application ID of the application protocol. |
| URL Category | uint32 | The internal identification number of the URL category. |
| URL Reputation | uint32 | The internal identification number for the URL reputation. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| String Block Type | uint32 | Initiates a String data block for the client application URL. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the client application URL String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the client application URL string. |
| Client Application URL | string | URL the client application accessed, if applicable (`/files/index.html`, for example). |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for the client application version. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block for the client application version, including eight bytes for the string block type and length, plus the number of bytes in the version. |
| Client Application Version | string | Client application version. |
| Monitor Rule 1 | uint32 | The ID of the first monitor rule associated with the connection event. |
| Monitor Rule 2 | uint32 | The ID of the second monitor rule associated with the connection event. |
| Monitor Rule 3 | uint32 | The ID of the third monitor rule associated with the connection event. |
| Monitor Rule 4 | uint32 | The ID of the fourth monitor rule associated with the connection event. |
| Monitor Rule 5 | uint32 | The ID of the fifth monitor rule associated with the connection event. |

*Table B-41 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Monitor Rule 6 | uint32 | The ID of the sixth monitor rule associated with the connection event. |
| Monitor Rule 7 | uint32 | The ID of the seventh monitor rule associated with the connection event. |
| Monitor Rule 8 | uint32 | The ID of the eighth monitor rule associated with the connection event. |
| Security Intelligence Source/ Destination | uint8 | Whether the source or destination IP address matched the IP block list. |
| Security Intelligence Layer | uint8 | The IP layer that matched the IP block list. |
| File Event Count | uint16 | Value used to distinguish between file events that happen during the same second. |
| Intrusion Event Count | uint16 | Value used to distinguish between intrusion events that happen during the same second. |
| Initiator Country | uint16 | Code for the country of the initiating host. |
| Responder Country | uint 16 | Code for the country of the responding host. |
| IOC Number | uint16 | ID Number of the compromise associated with this event. |
| Source Autonomous System | uint32 | Autonomous system number of the source, either origin or peer. |
| Destination Autonomous System | uint32 | Autonomous system number of the destination, either origin or peer. |
| SNMP Input | uint16 | SNMP index of the input interface. |
| SNMP Output | uint16 | SNMP index of the output interface. |
| Source TOS | uint8 | Type of Service byte setting for the incoming interface. |
| Destination TOS | uint8 | Type of Service byte setting for the outgoing interface. |
| Source Mask | uint8 | Source address prefix mask. |
| Destination Mask | uint8 | Destination address prefix mask. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |
| VLAN ID | uint16 | VLAN identification number that indicates which VLAN the host is a member of. |
| String Block Type | uint32 | Initiates a String data block containing the Referenced Host. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the Referenced Host String data block, including eight bytes for the block type and header fields plus the number of bytes in the Referenced Host field. |

*Table B-41 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Referenced Host | string | Host name information provided in HTTP or DNS. |
| String Block Type | uint32 | Initiates a String data block containing the User Agent. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the User Agent String data block, including eight bytes for the block type and header fields plus the number of bytes in the User Agent field. |
| User Agent | string | Information from the UserAgent header field in the session. |
| String Block Type | uint32 | Initiates a String data block containing the HTTP Referrer. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the HTTP Referrer String data block, including eight bytes for the block type and header fields plus the number of bytes in the HTTP Referrer field. |
| HTTP Referrer | string | The site from which a page originated. This is found int he Referred header information in HTTP traffic. |
| SSL Certificate Fingerprint | uint8[20] | SHA1 hash of the SSL Server certificate. |
| SSL Policy ID | uint8[16] | ID number of the SSL policy that handled the connection. |
| SSL Rule ID | uint32 | ID number of the SSL rule or default action that handled the connection. |
| SSL Cipher Suite | uint16 | Encryption suite used by the SSL connection. The value is stored in decimal format. See `www.iana.org/assignments/tls-parameters/tls-parameters.xhtml` for the cipher suite designated by the value. |
| SSL Version | uint8 | The SSL or TLS protocol version used to encrypt the connection. |
| SSL Server Certificate Status | uint16 | The status of the SSL certificate. Possible values include:<br>• `0` — Not checked — The server certificate status was not evaluated.<br>• `1` — Unknown — The server certificate status could not be determined.<br>• `2` — Valid — The server certificate is valid.<br>• `4` — Self-signed — The server certificate is self-signed.<br>• `16` — Invalid Issuer — The server certificate has an invalid issuer.<br>• `32` — Invalid Signature — The server certificate has an invalid signature.<br>• `64` — Expired — The server certificate is expired.<br>• `128` — Not valid yet — The server certificate is not yet valid.<br>• `256` — Revoked — The server certificate has been revoked. |

*Table B-41 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Actual Action | uint16 | The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |
| SSL Expected Action | uint16 | The action which should be performed on the connection based on the SSL Rule. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |

*Table B-41 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Status | uint16 | Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'No Match'<br>• 2 — 'Success'<br>• 3 — 'Uncached Session'<br>• 4 — 'Unknown Cipher Suite'<br>• 5 — 'Unsupported Cipher Suite'<br>• 6 — 'Unsupported SSL Version'<br>• 7 — 'SSL Compression Used'<br>• 8 — 'Session Undecryptable in Passive Mode'<br>• 9 — 'Handshake Error'<br>• 10 — 'Decryption Error'<br>• 11 — 'Pending Server Name Category Lookup'<br>• 12 — 'Pending Common Name Category Lookup'<br>• 13 — 'Internal Error'<br>• 14 — 'Network Parameters Unavailable'<br>• 15 — 'Invalid Server Certificate Handle'<br>• 16 — 'Server Certificate Fingerprint Unavailable'<br>• 17 — 'Cannot Cache Subject DN'<br>• 18 — 'Cannot Cache Issuer DN'<br>• 19 — 'Unknown SSL Version'<br>• 20 — 'External Certificate List Unavailable'<br>• 21 — 'External Certificate Fingerprint Unavailable'<br>• 22 — 'Internal Certificate List Invalid'<br>• 23 — 'Internal Certificate List Unavailable'<br>• 24 — 'Internal Certificate Unavailable'<br>• 25 — 'Internal Certificate Fingerprint Unavailable'<br>• 26 — 'Server Certificate Validation Unavailable'<br>• 27 — 'Server Certificate Validation Failure'<br>• 28 — 'Invalid Action' |
| SSL Flow Error | uint32 | Detailed SSL error code. These values may be needed for support purposes. |

*Table B-41 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Messages | uint32 | The messages exchanged between client and server during the SSL handshake. See http://tools.ietf.org/html/rfc5246 for more information.<br>• `0x00000001` — NSE_MT__HELLO_REQUEST<br>• `0x00000002` — NSE_MT__CLIENT_ALERT<br>• `0x00000004` — NSE_MT__SERVER_ALERT<br>• `0x00000008` — NSE_MT__CLIENT_HELLO<br>• `0x00000010` — NSE_MT__SERVER_HELLO<br>• `0x00000020` — NSE_MT__SERVER_CERTIFICATE<br>• `0x00000040` — NSE_MT__SERVER_KEY_EXCHANGE<br>• `0x00000080` — NSE_MT__CERTIFICATE_REQUEST<br>• `0x00000100` — NSE_MT__SERVER_HELLO_DONE<br>• `0x00000200` — NSE_MT__CLIENT_CERTIFICATE<br>• `0x00000400` — NSE_MT__CLIENT_KEY_EXCHANGE<br>• `0x00000800` — NSE_MT__CERTIFICATE_VERIFY<br>• `0x00001000` — NSE_MT__CLIENT_CHANGE_CIPHER_SPEC<br>• `0x00002000` — NSE_MT__CLIENT_FINISHED<br>• `0x00004000` — NSE_MT__SERVER_CHANGE_CIPHER_SPEC<br>• `0x00008000` — NSE_MT__SERVER_FINISHED<br>• `0x00010000` — NSE_MT__NEW_SESSION_TICKET<br>• `0x00020000` — NSE_MT__HANDSHAKE_OTHER<br>• `0x00040000` — NSE_MT__APP_DATA_FROM_CLIENT<br>• `0x00080000` — NSE_MT__APP_DATA_FROM_SERVER |
| SSL Flow Flags | uint64 | The debugging level flags for an encrypted connection. Possible values include:<br>• `0x00000001` — NSE_FLOW__VALID - must be set for other fields to be valid<br>• `0x00000002` — NSE_FLOW__INITIALIZED - internal structures ready for processing<br>• `0x00000004` — NSE_FLOW__INTERCEPT - SSL session has been intercepted |
| String Block Type | uint32 | Initiates a String data block containing the SSL Server Name. This value is always `0`. |

*Table B-41 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the SSL Server Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the SSL Server Name field. |
| SSL Server Name | string | Name provided in the server name indication in the SSL Client Hello. |
| SSL URL Category | uint32 | Category of the flow as identified from the server name and certificate common name. |
| SSL Session ID | uint8[32] | Value of the session ID used during the SSL handshake when the client and server agree to do session reuse |
| SSL Session ID Length | uint8 | Length of the SSL Session ID. While the session ID cannot exceed 32 bytes, it may be less than 32 bytes. |
| SSL Ticket ID | uint8[20] | Hash of the session ticket used when the client and server agree to use a session ticket. |
| SSL Ticket ID Length | uint8 | Length of the SSL Ticket ID. While the ticket ID cannot exceed 20 bytes, it may be less than 20 bytes. |
| Network Analysis Policy revision | uint8[16] | Revision of the Network Analysis Policy associated with the connection event. |

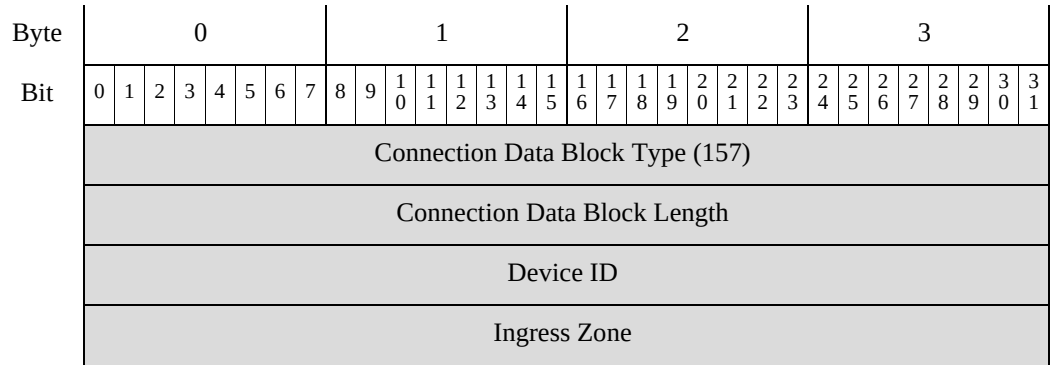
# Connection Statistics Data Block 5.4.1

The connection statistics data block is used in connection data messages. Several new fields have been added to the Connection Statistics Data Block for 5.4. Fields have been added to support SSL connections, HTTP redirection, and network analysis policies. The connection statistics data block for version 5.4+ has a block type of 157 in the series 1 group of blocks. It deprecates block type 155, Connection Statistics Data Block 5.3.1, page B-191.

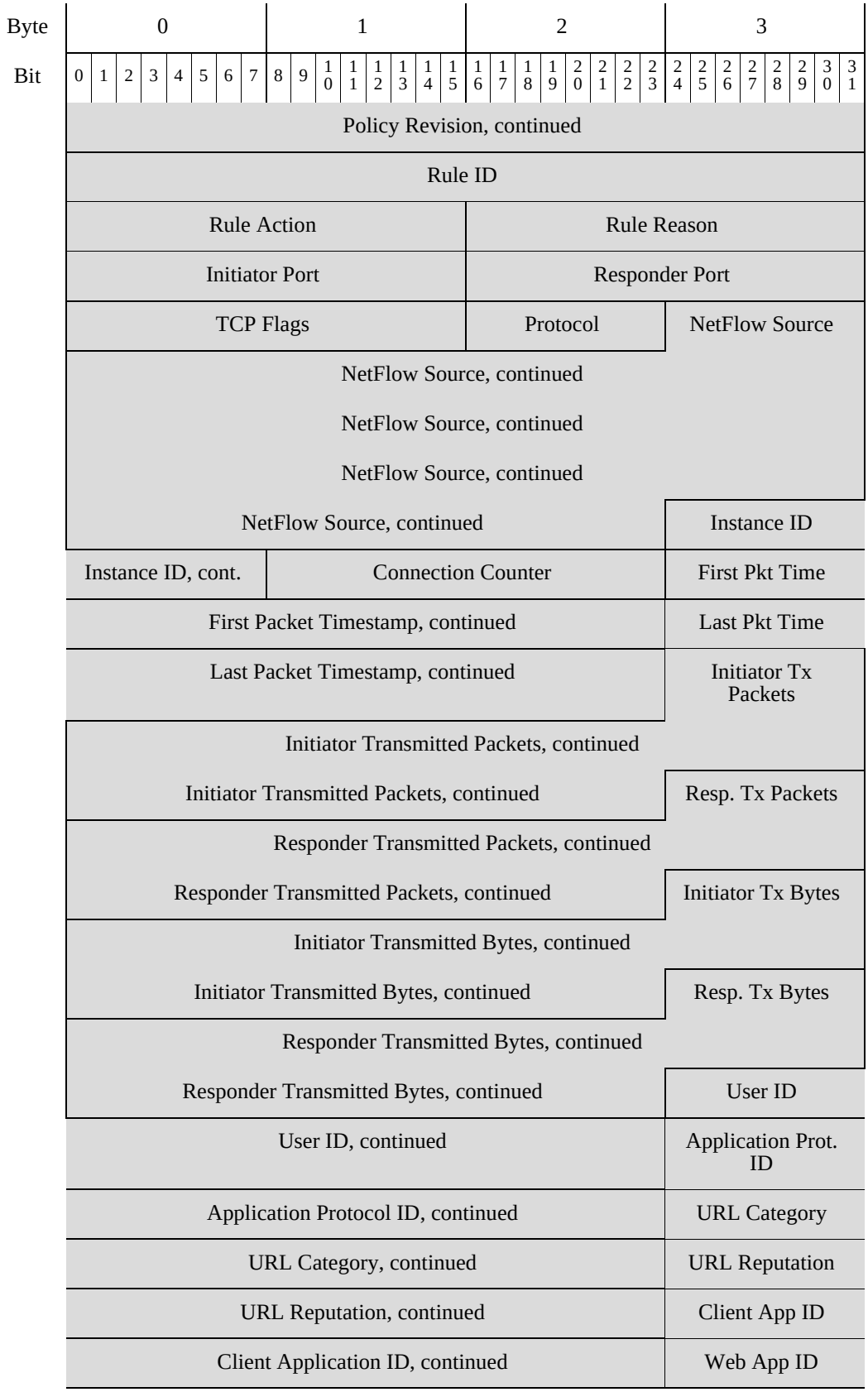
You request connection event records by setting the extended event flag—bit 30 in the Request Flags field—in the request message with an event version of 12 and an event code of 71. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.

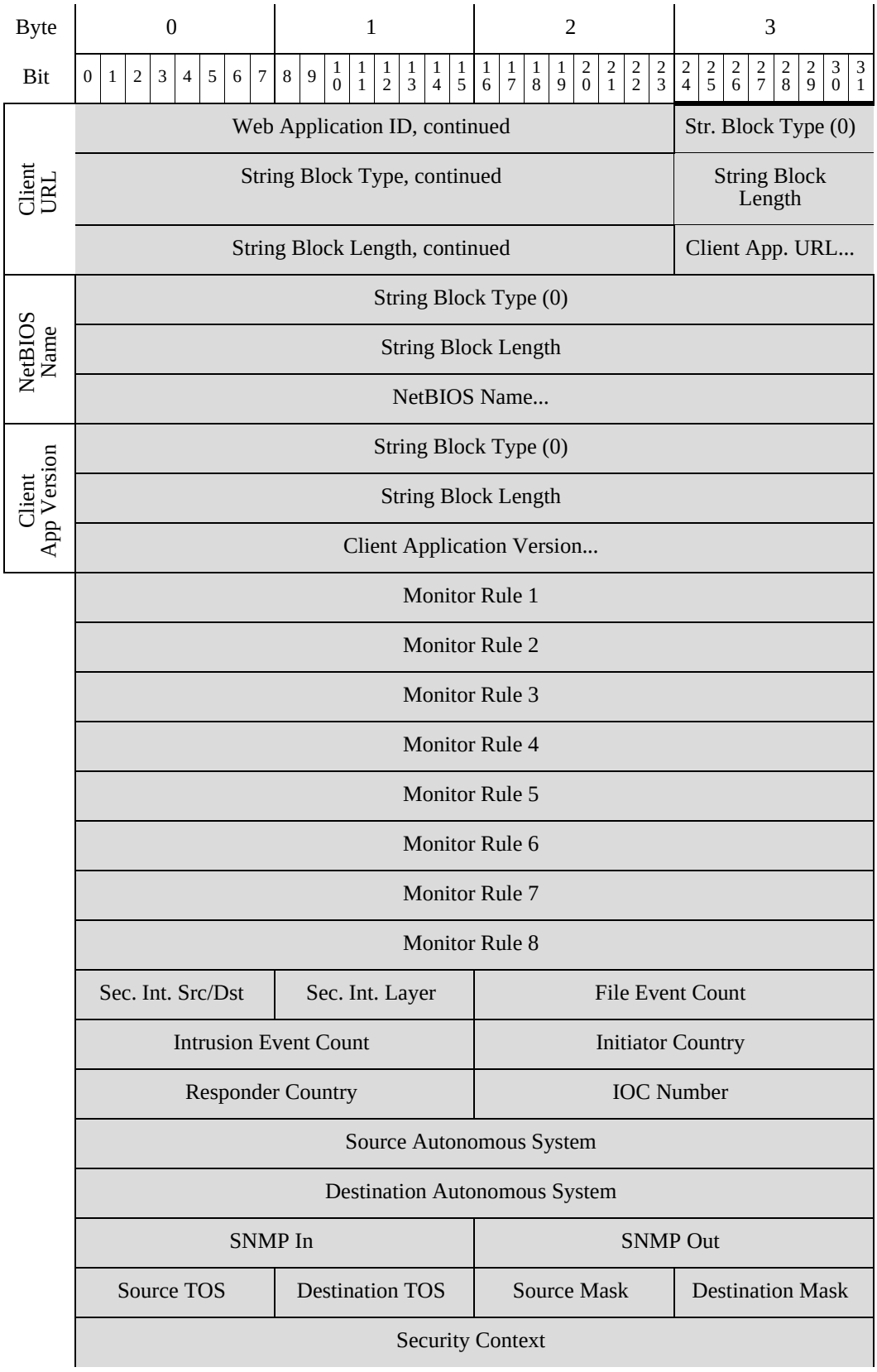
For more information on the Connection Statistics Data message, see Connection Statistics Data Message, page 4-53.

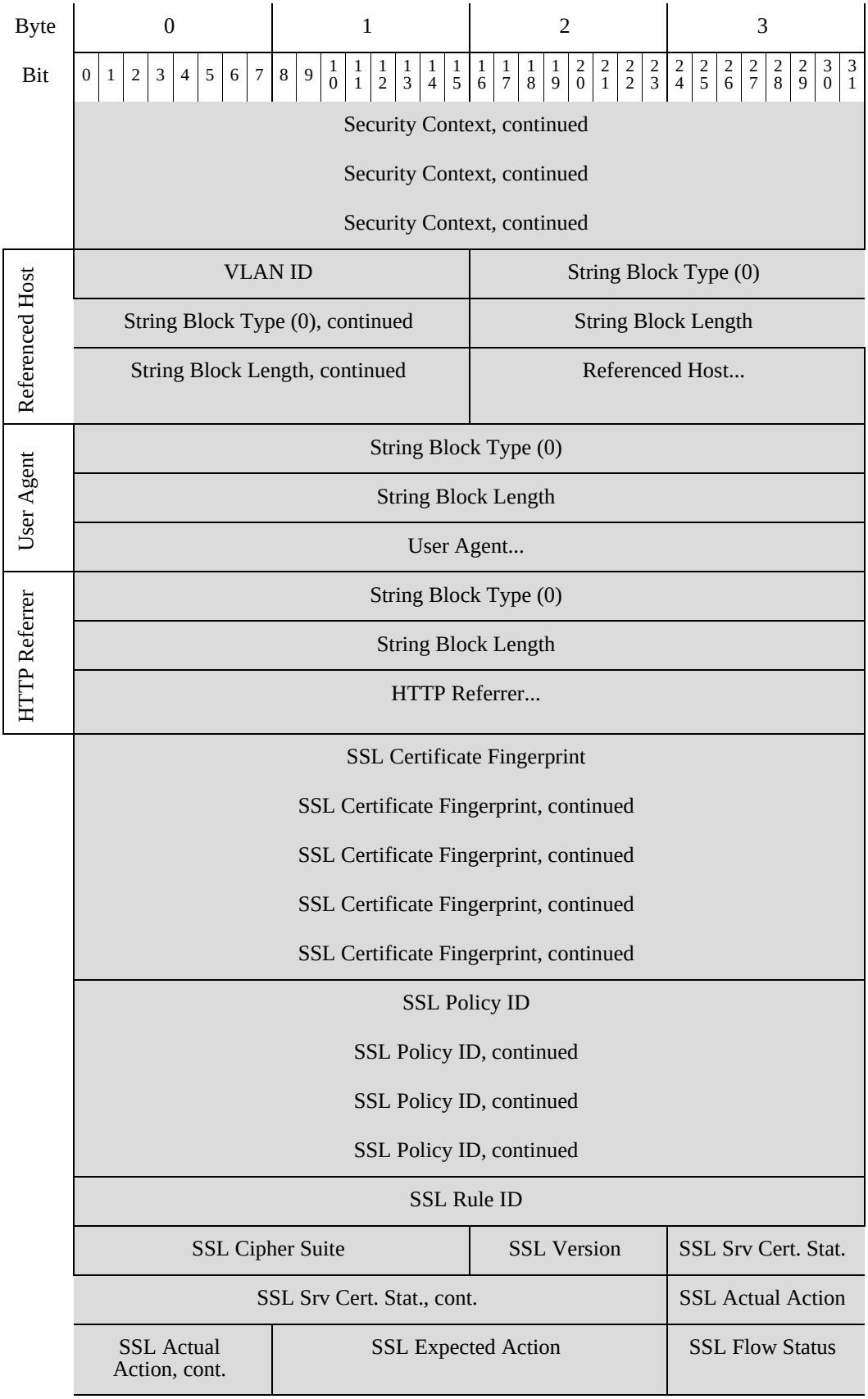
The following diagram shows the format of a Connection Statistics data block for 5.4+:

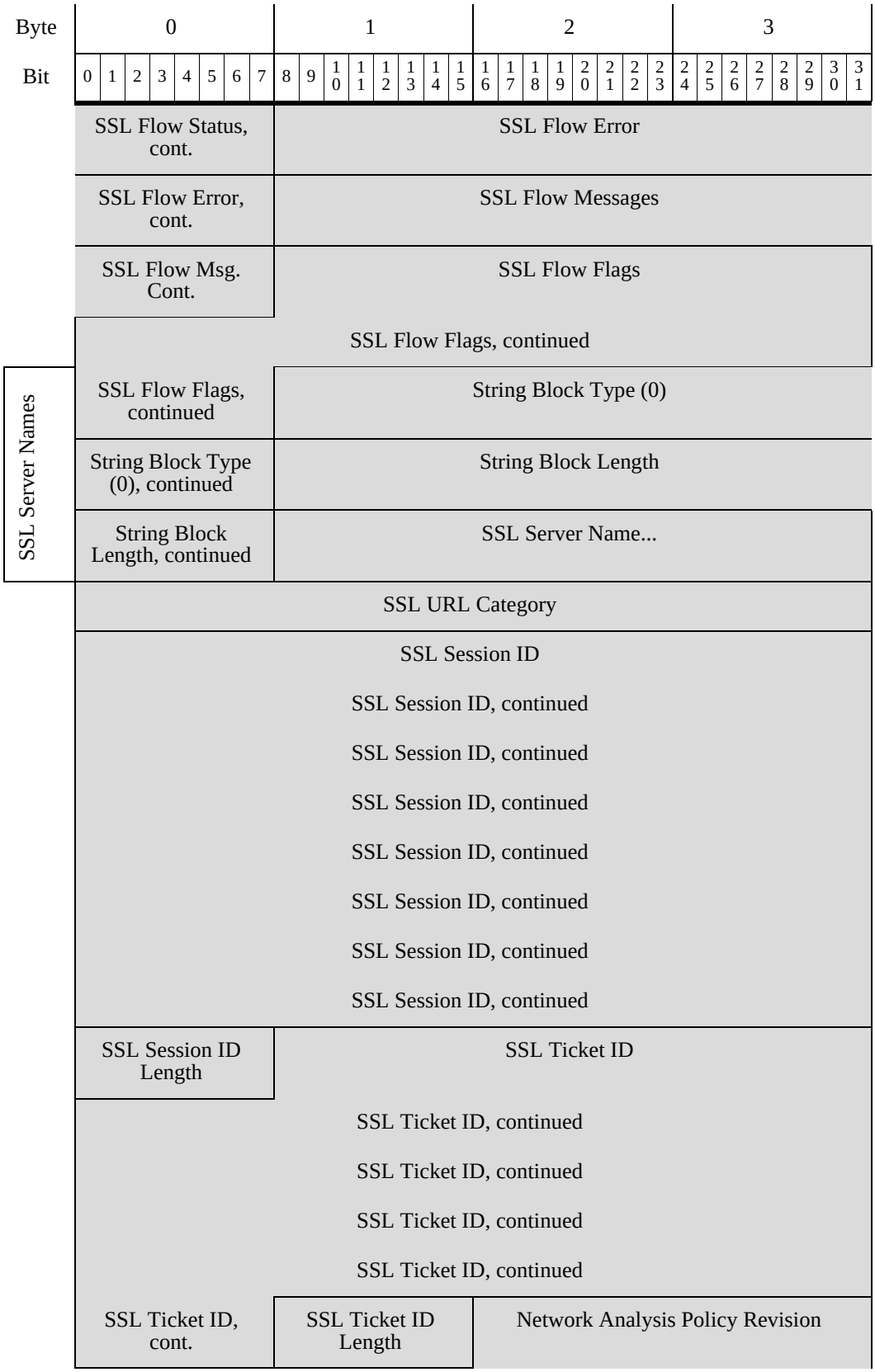


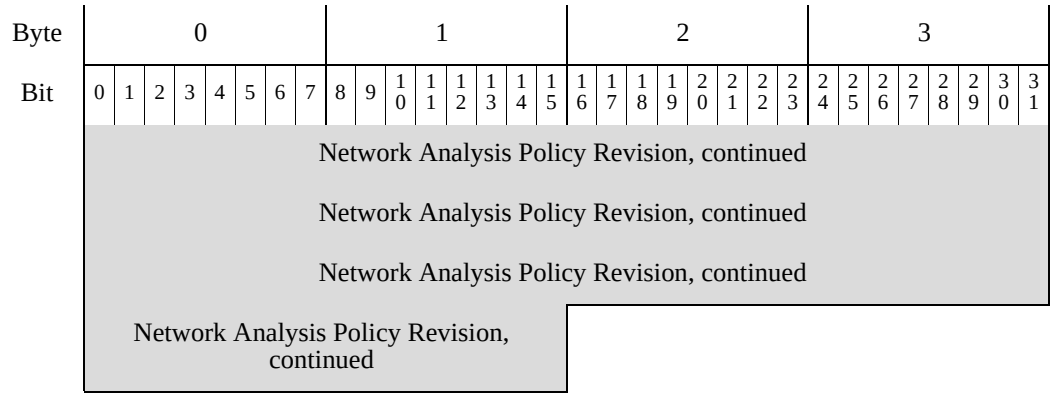
| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Policy Revision, continued
Rule ID
Rule Action
Rule Reason
Initiator Port
Responder Port
TCP Flags
Protocol
NetFlow Source
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
Instance ID
Instance ID, cont.
Connection Counter
First Pkt Time
First Packet Timestamp, continued
Last Pkt Time
Last Packet Timestamp, continued
Initiator Tx Packets
Initiator Transmitted Packets, continued
Initiator Transmitted Packets, continued
Resp. Tx Packets
Responder Transmitted Packets, continued
Responder Transmitted Packets, continued
Initiator Tx Bytes
Initiator Transmitted Bytes, continued
Initiator Transmitted Bytes, continued
Resp. Tx Bytes
Responder Transmitted Bytes, continued
Responder Transmitted Bytes, continued
User ID
User ID, continued
Application Prot. ID
Application Protocol ID, continued
URL Category
URL Category, continued
URL Reputation
URL Reputation, continued
Client App ID
Client Application ID, continued
Web App ID


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Client URL
Web Application ID, continued
Str. Block Type (0)
String Block Type, continued
String Block Length
String Block Length, continued
Client App. URL...
NetBIOS Name
String Block Type (0)
String Block Length
NetBIOS Name...
Client App Version
String Block Type (0)
String Block Length
Client Application Version...
Monitor Rule 1
Monitor Rule 2
Monitor Rule 3
Monitor Rule 4
Monitor Rule 5
Monitor Rule 6
Monitor Rule 7
Monitor Rule 8
Sec. Int. Src/Dst
Sec. Int. Layer
File Event Count
Intrusion Event Count
Initiator Country
Responder Country
IOC Number
Source Autonomous System
Destination Autonomous System
SNMP In
SNMP Out
Source TOS
Destination TOS
Source Mask
Destination Mask
Security Context


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Security Context, continued
Security Context, continued
Security Context, continued
Referenced Host
VLAN ID
String Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
Referenced Host...
User Agent
String Block Type (0)
String Block Length
User Agent...
HTTP Referrer
String Block Type (0)
String Block Length
HTTP Referrer...
SSL Certificate Fingerprint
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Policy ID
SSL Policy ID, continued
SSL Policy ID, continued
SSL Policy ID, continued
SSL Rule ID
SSL Cipher Suite
SSL Version
SSL Srv Cert. Stat.
SSL Srv Cert. Stat., cont.
SSL Actual Action
SSL Actual Action, cont.
SSL Expected Action
SSL Flow Status


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
SSL Flow Status, cont.
SSL Flow Error
SSL Flow Error, cont.
SSL Flow Messages
SSL Flow Msg. Cont.
SSL Flow Flags
SSL Flow Flags, continued
SSL Server Names
SSL Flow Flags, continued
String Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
SSL Server Name...
SSL URL Category
SSL Session ID
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID Length
SSL Ticket ID
SSL Ticket ID, continued
SSL Ticket ID, continued
SSL Ticket ID, continued
SSL Ticket ID, continued
SSL Ticket ID, cont.
SSL Ticket ID Length
Network Analysis Policy Revision




The following table describes the fields of the Connection Statistics data block for 5.4+.

***Table B-42 Connection Statistics Data Block 5.4+ Fields***

| Field | Data Type | Description |
|---|---|---|
| Connection Statistics Data Block Type | uint32 | Initiates a Connection Statistics data block for 5.4+. The value is always `157`. |
| Connection Statistics Data Block Length | uint32 | Number of bytes in the Connection Statistics data block, including eight bytes for the connection statistics block type and length fields, plus the number of bytes in the connection data that follows. |
| Device ID | uint32 | The device that detected the connection event. |
| Ingress Zone | uint8[16] | Ingress security zone in the event that triggered the policy violation. |
| Egress Zone | uint8[16] | Egress security zone in the event that triggered the policy violation. |
| Ingress Interface | uint8[16] | Interface for the inbound traffic. |
| Egress Interface | uint8[16] | Interface for the outbound traffic. |
| Initiator IP Address | uint8[16] | IP address of the host that initiated the session described in the connection event, in IP address octets. |
| Responder IP Address | uint8[16] | IP address of the host that responded to the initiating host, in IP address octets. |
| Policy Revision | uint8[16] | Revision number of the rule associated with the triggered correlation event, if applicable. |
| Rule ID | uint32 | Internal identifier for the rule that triggered the event, if applicable. |
| Rule Action | uint16 | The action selected in the user interface for that rule (allow, block, and so forth). |
| Rule Reason | uint16 | The reason the rule triggered the event. |
| Initiator Port | uint16 | Port used by the initiating host. |
| Responder Port | uint16 | Port used by the responding host. |

*Table B-42 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| TCP Flags | uint16 | Indicates any TCP flags for the connection event. |
| Protocol | uint8 | The IANA-specified protocol number. |
| NetFlow Source | uint8[16] | IP address of the NetFlow-enabled device that exported the data for the connection. |
| Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| First Packet Timestamp | uint32 | UNIX timestamp of the date and time the first packet was exchanged in the session. |
| Last Packet Timestamp | uint32 | UNIX timestamp of the date and time the last packet was exchanged in the session. |
| Initiator Transmitted Packets | uint64 | Number of packets transmitted by the initiating host. |
| Responder Transmitted Packets | uint64 | Number of packets transmitted by the responding host. |
| Initiator Transmitted Bytes | uint64 | Number of bytes transmitted by the initiating host. |
| Responder Transmitted Bytes | uint64 | Number of bytes transmitted by the responding host. |
| User ID | uint32 | Internal identification number for the user who last logged into the host that generated the traffic. |
| Application Protocol ID | uint32 | Application ID of the application protocol. |
| URL Category | uint32 | The internal identification number of the URL category. |
| URL Reputation | uint32 | The internal identification number for the URL reputation. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| String Block Type | uint32 | Initiates a String data block for the client application URL. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the client application URL String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the client application URL string. |
| Client Application URL | string | URL the client application accessed, if applicable (`/files/index.html`, for example). |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always `0`. |

*Table B-42 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for the client application version. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block for the client application version, including eight bytes for the string block type and length, plus the number of bytes in the version. |
| Client Application Version | string | Client application version. |
| Monitor Rule 1 | uint32 | The ID of the first monitor rule associated with the connection event. |
| Monitor Rule 2 | uint32 | The ID of the second monitor rule associated with the connection event. |
| Monitor Rule 3 | uint32 | The ID of the third monitor rule associated with the connection event. |
| Monitor Rule 4 | uint32 | The ID of the fourth monitor rule associated with the connection event. |
| Monitor Rule 5 | uint32 | The ID of the fifth monitor rule associated with the connection event. |
| Monitor Rule 6 | uint32 | The ID of the sixth monitor rule associated with the connection event. |
| Monitor Rule 7 | uint32 | The ID of the seventh monitor rule associated with the connection event. |
| Monitor Rule 8 | uint32 | The ID of the eighth monitor rule associated with the connection event. |
| Security Intelligence Source/ Destination | uint8 | Whether the source or destination IP address matched the IP block list. |
| Security Intelligence Layer | uint8 | The IP layer that matched the IP block list. |
| File Event Count | uint16 | Value used to distinguish between file events that happen during the same second. |
| Intrusion Event Count | uint16 | Value used to distinguish between intrusion events that happen during the same second. |
| Initiator Country | uint16 | Code for the country of the initiating host. |
| Responder Country | uint 16 | Code for the country of the responding host. |
| IOC Number | uint16 | ID Number of the compromise associated with this event. |

*Table B-42 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Source Autonomous System | uint32 | Autonomous system number of the source, either origin or peer. |
| Destination Autonomous System | uint32 | Autonomous system number of the destination, either origin or peer. |
| SNMP Input | uint16 | SNMP index of the input interface. |
| SNMP Output | uint16 | SNMP index of the output interface. |
| Source TOS | uint8 | Type of Service byte setting for the incoming interface. |
| Destination TOS | uint8 | Type of Service byte setting for the outgoing interface. |
| Source Mask | uint8 | Source address prefix mask. |
| Destination Mask | uint8 | Destination address prefix mask. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |
| VLAN ID | uint16 | VLAN identification number that indicates which VLAN the host is a member of. |
| String Block Type | uint32 | Initiates a String data block containing the Referenced Host. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the Referenced Host String data block, including eight bytes for the block type and header fields plus the number of bytes in the Referenced Host field. |
| Referenced Host | string | Host name information provided in HTTP or DNS. |
| String Block Type | uint32 | Initiates a String data block containing the User Agent. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the User Agent String data block, including eight bytes for the block type and header fields plus the number of bytes in the User Agent field. |
| User Agent | string | Information from the UserAgent header field in the session. |
| String Block Type | uint32 | Initiates a String data block containing the HTTP Referrer. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the HTTP Referrer String data block, including eight bytes for the block type and header fields plus the number of bytes in the HTTP Referrer field. |
| HTTP Referrer | string | The site from which a page originated. This is found int he Referred header information in HTTP traffic. |
| SSL Certificate Fingerprint | uint8[20] | SHA1 hash of the SSL Server certificate. |
| SSL Policy ID | uint8[16] | ID number of the SSL policy that handled the connection. |
| SSL Rule ID | uint32 | ID number of the SSL rule or default action that handled the connection. |

*Table B-42 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Cipher Suite | uint16 | Encryption suite used by the SSL connection. The value is stored in decimal format. See `www.iana.org/assignments/tls-parameters/tls-parameters.xhtml` for the cipher suite designated by the value. |
| SSL Version | uint8 | The SSL or TLS protocol version used to encrypt the connection. |
| SSL Server Certificate Status | uint32 | The status of the SSL certificate. Possible values include:<br>• `0` — Not checked — The server certificate status was not evaluated.<br>• `1` — Unknown — The server certificate status could not be determined.<br>• `2` — Valid — The server certificate is valid.<br>• `4` — Self-signed — The server certificate is self-signed.<br>• `16` — Invalid Issuer — The server certificate has an invalid issuer.<br>• `32` — Invalid Signature — The server certificate has an invalid signature.<br>• `64` — Expired — The server certificate is expired.<br>• `128` — Not valid yet — The server certificate is not yet valid.<br>• `256` — Revoked — The server certificate has been revoked. |
| SSL Actual Action | uint16 | The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include:<br>• `0` — 'Unknown'<br>• `1` — 'Do Not Decrypt'<br>• `2` — 'Block'<br>• `3` — 'Block With Reset'<br>• `4` — 'Decrypt (Known Key)'<br>• `5` — 'Decrypt (Replace Key)'<br>• `6` — 'Decrypt (Resign)' |
| SSL Expected Action | uint16 | The action which should be performed on the connection based on the SSL Rule. Possible values include:<br>• `0` — 'Unknown'<br>• `1` — 'Do Not Decrypt'<br>• `2` — 'Block'<br>• `3` — 'Block With Reset'<br>• `4` — 'Decrypt (Known Key)'<br>• `5` — 'Decrypt (Replace Key)'<br>• `6` — 'Decrypt (Resign)' |

*Table B-42 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Status | uint16 | Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'No Match'<br>• 2 — 'Success'<br>• 3 — 'Uncached Session'<br>• 4 — 'Unknown Cipher Suite'<br>• 5 — 'Unsupported Cipher Suite'<br>• 6 — 'Unsupported SSL Version'<br>• 7 — 'SSL Compression Used'<br>• 8 — 'Session Undecryptable in Passive Mode'<br>• 9 — 'Handshake Error'<br>• 10 — 'Decryption Error'<br>• 11 — 'Pending Server Name Category Lookup'<br>• 12 — 'Pending Common Name Category Lookup'<br>• 13 — 'Internal Error'<br>• 14 — 'Network Parameters Unavailable'<br>• 15 — 'Invalid Server Certificate Handle'<br>• 16 — 'Server Certificate Fingerprint Unavailable'<br>• 17 — 'Cannot Cache Subject DN'<br>• 18 — 'Cannot Cache Issuer DN'<br>• 19 — 'Unknown SSL Version'<br>• 20 — 'External Certificate List Unavailable'<br>• 21 — 'External Certificate Fingerprint Unavailable'<br>• 22 — 'Internal Certificate List Invalid'<br>• 23 — 'Internal Certificate List Unavailable'<br>• 24 — 'Internal Certificate Unavailable'<br>• 25 — 'Internal Certificate Fingerprint Unavailable'<br>• 26 — 'Server Certificate Validation Unavailable'<br>• 27 — 'Server Certificate Validation Failure'<br>• 28 — 'Invalid Action' |
| SSL Flow Error | uint32 | Detailed SSL error code. These values may be needed for support purposes. |

*Table B-42 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Messages | uint32 | The messages exchanged between client and server during the SSL handshake. See http://tools.ietf.org/html/rfc5246 for more information.<br>• `0x00000001` — NSE_MT__HELLO_REQUEST<br>• `0x00000002` — NSE_MT__CLIENT_ALERT<br>• `0x00000004` — NSE_MT__SERVER_ALERT<br>• `0x00000008` — NSE_MT__CLIENT_HELLO<br>• `0x00000010` — NSE_MT__SERVER_HELLO<br>• `0x00000020` — NSE_MT__SERVER_CERTIFICATE<br>• `0x00000040` — NSE_MT__SERVER_KEY_EXCHANGE<br>• `0x00000080` — NSE_MT__CERTIFICATE_REQUEST<br>• `0x00000100` — NSE_MT__SERVER_HELLO_DONE<br>• `0x00000200` — NSE_MT__CLIENT_CERTIFICATE<br>• `0x00000400` — NSE_MT__CLIENT_KEY_EXCHANGE<br>• `0x00000800` — NSE_MT__CERTIFICATE_VERIFY<br>• `0x00001000` — NSE_MT__CLIENT_CHANGE_CIPHER_SPEC<br>• `0x00002000` — NSE_MT__CLIENT_FINISHED<br>• `0x00004000` — NSE_MT__SERVER_CHANGE_CIPHER_SPEC<br>• `0x00008000` — NSE_MT__SERVER_FINISHED<br>• `0x00010000` — NSE_MT__NEW_SESSION_TICKET<br>• `0x00020000` — NSE_MT__HANDSHAKE_OTHER<br>• `0x00040000` — NSE_MT__APP_DATA_FROM_CLIENT<br>• `0x00080000` — NSE_MT__APP_DATA_FROM_SERVER |
| SSL Flow Flags | uint64 | The debugging level flags for an encrypted connection. Possible values include:<br>• `0x00000001` — NSE_FLOW__VALID - must be set for other fields to be valid<br>• `0x00000002` — NSE_FLOW__INITIALIZED - internal structures ready for processing<br>• `0x00000004` — NSE_FLOW__INTERCEPT - SSL session has been intercepted |
| String Block Type | uint32 | Initiates a String data block containing the SSL Server Name. This value is always `0`. |

*Table B-42 Connection Statistics Data Block 5.4+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the SSL Server Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the SSL Server Name field. |
| SSL Server Name | string | Name provided in the server name indication in the SSL Client Hello. |
| SSL URL Category | uint32 | Category of the flow as identified from the server name and certificate common name. |
| SSL Session ID | uint8[32] | Value of the session ID used during the SSL handshake when the client and server agree to do session reuse |
| SSL Session ID Length | uint8 | Length of the SSL Session ID. While the session ID cannot exceed 32 bytes, it may be less than 32 bytes. |
| SSL Ticket ID | uint8[20] | Hash of the session ticket used when the client and server agree to use a session ticket. |
| SSL Ticket ID Length | uint8 | Length of the SSL Ticket ID. While the ticket ID cannot exceed 20 bytes, it may be less than 20 bytes. |
| Network Analysis Policy revision | uint8[16] | Revision of the Network Analysis Policy associated with the connection event. |

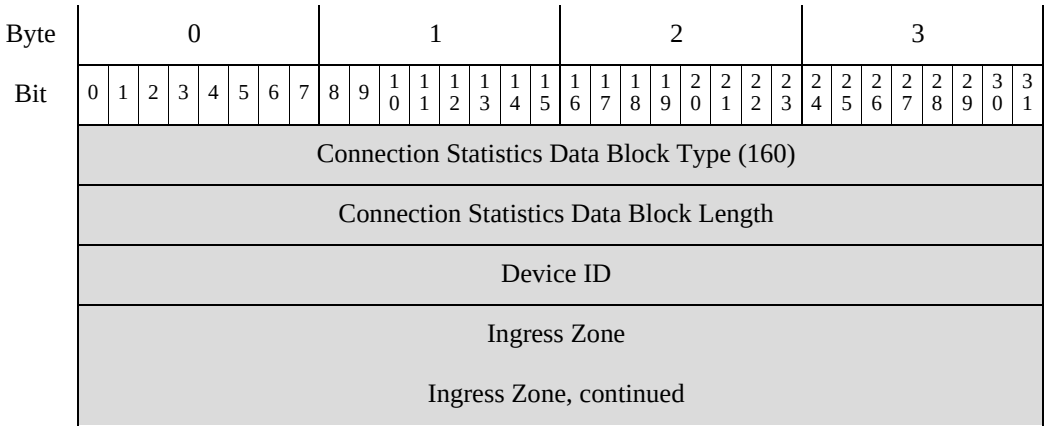
# Connection Statistics Data Block 6.0.x

The connection statistics data block is used in connection data messages. Several new fields have been added to the Connection Statistics Data Block for 6.0. Fields have been added to support ISE Integration and Multiple Network Maps. The connection statistics data block for version 6.0.x has a block type of 160 in the series 1 group of blocks. It supersedes block type 157, Connection Statistics Data Block 5.4.1, page B-211. New fields have been added to support DNS lookup and Security Intelligence.

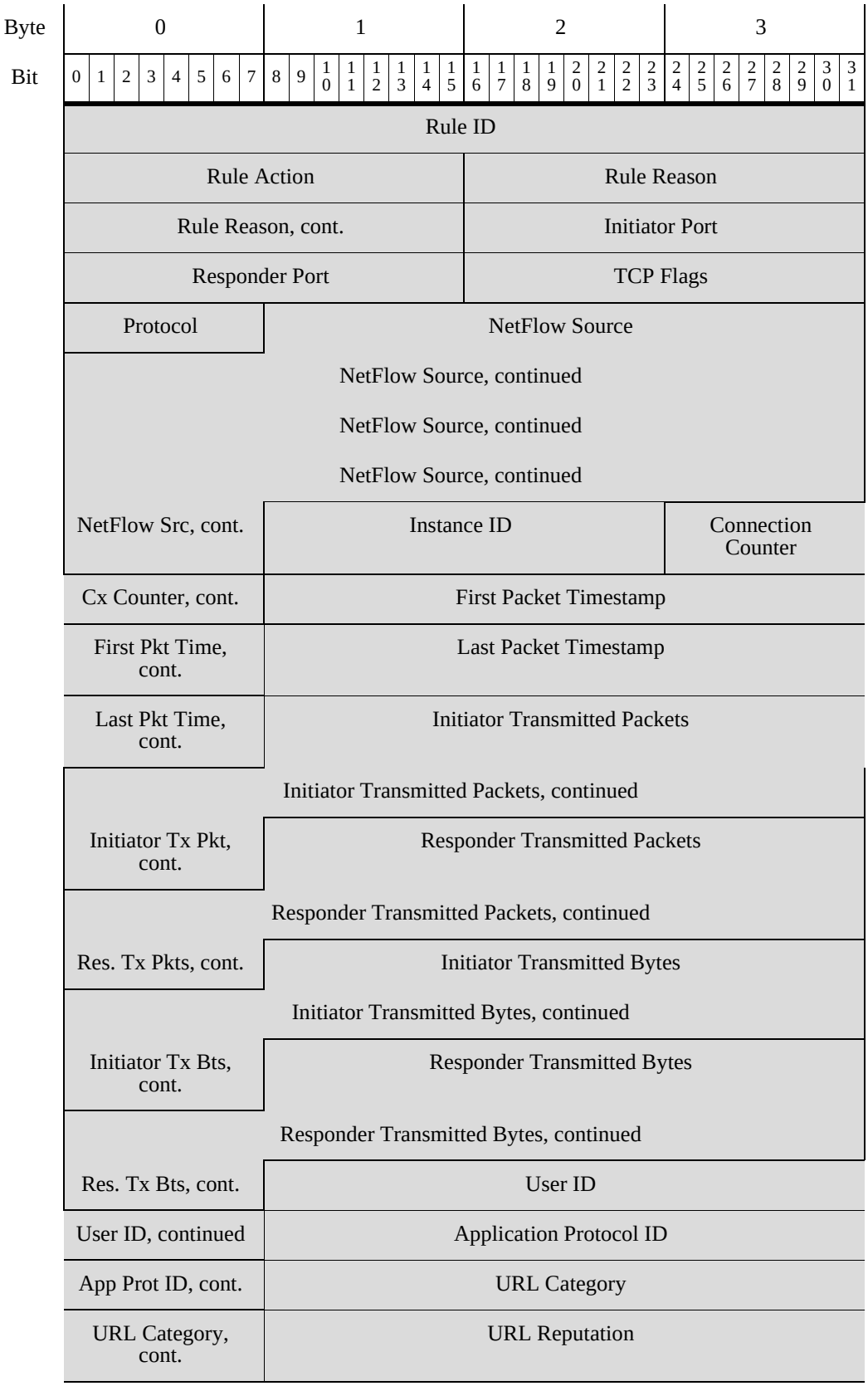
You request connection event records by setting the extended event flag—bit 30 in the Request Flags field—in the request message with an event version of 13 and an event code of 71. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.

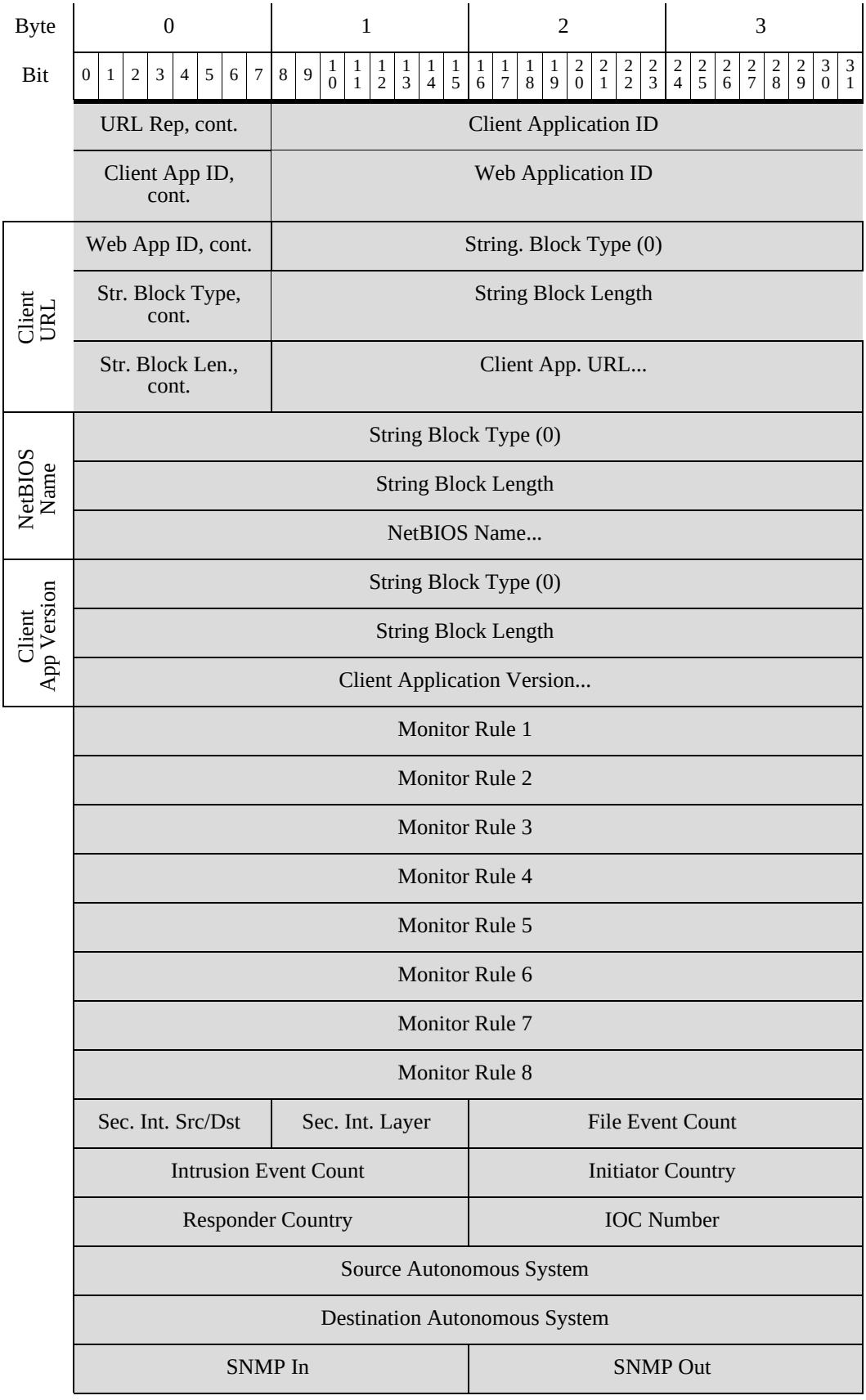
The following diagram shows the format of a Connection Statistics data block for 6.0.x:

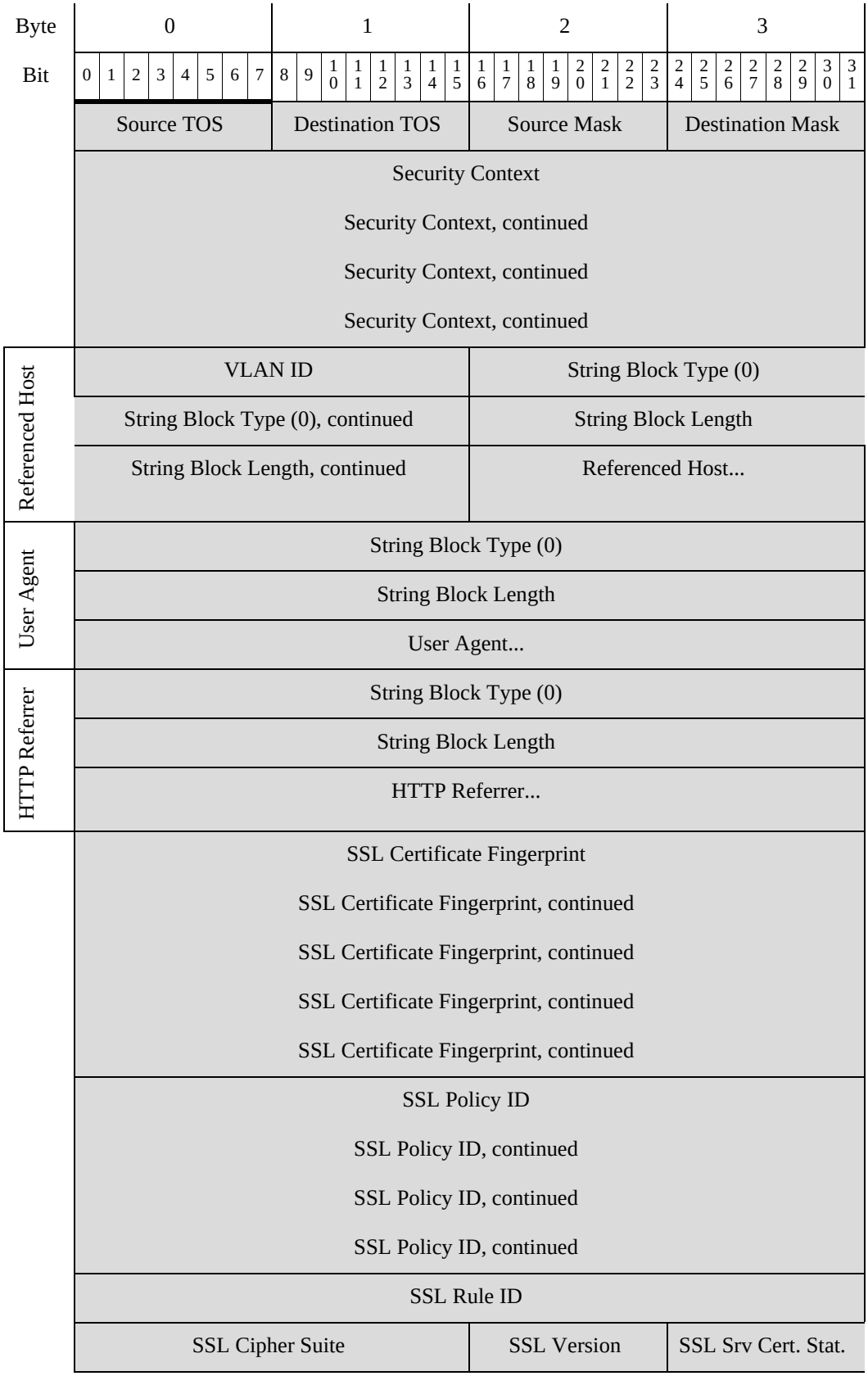
7

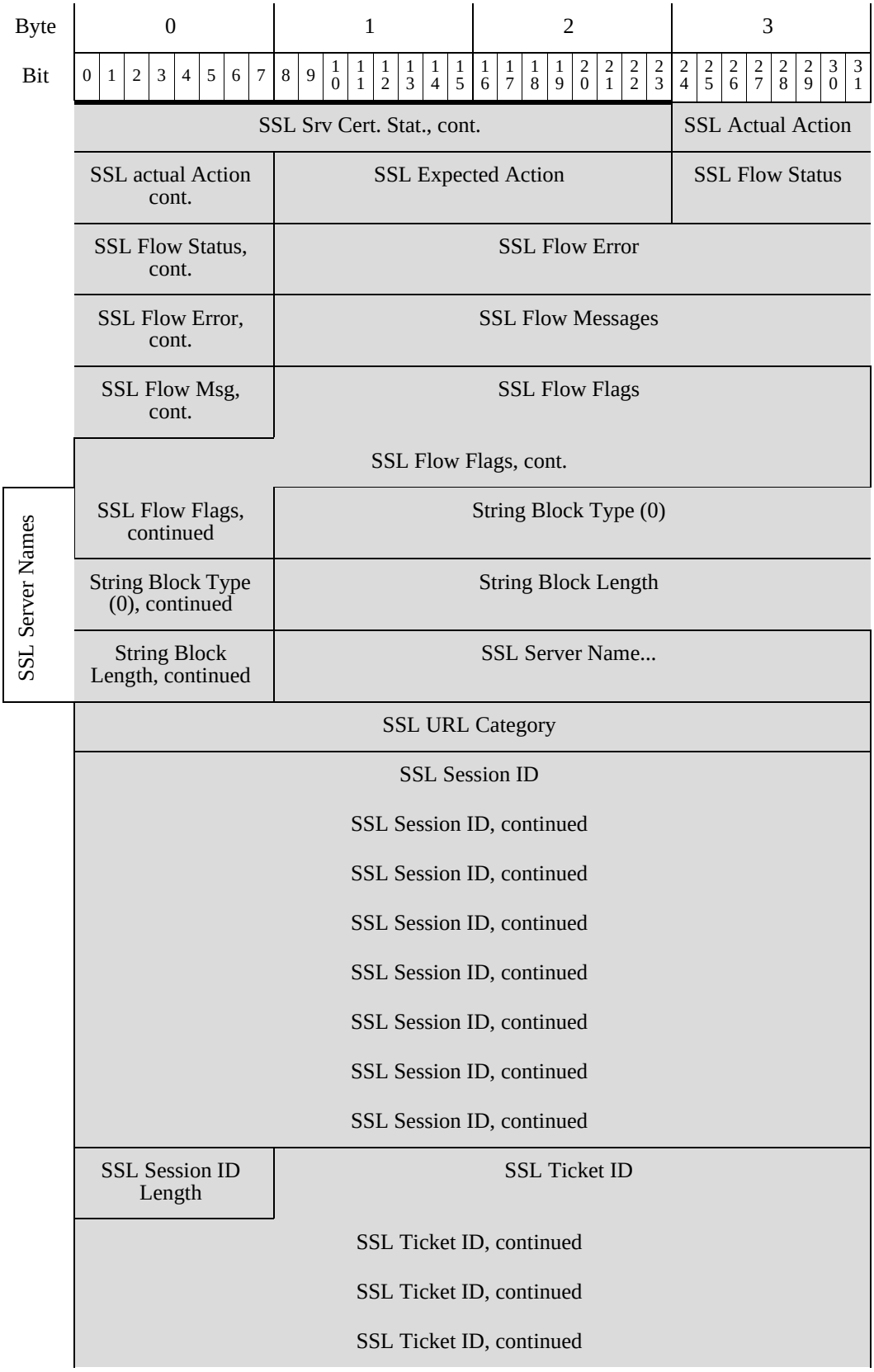


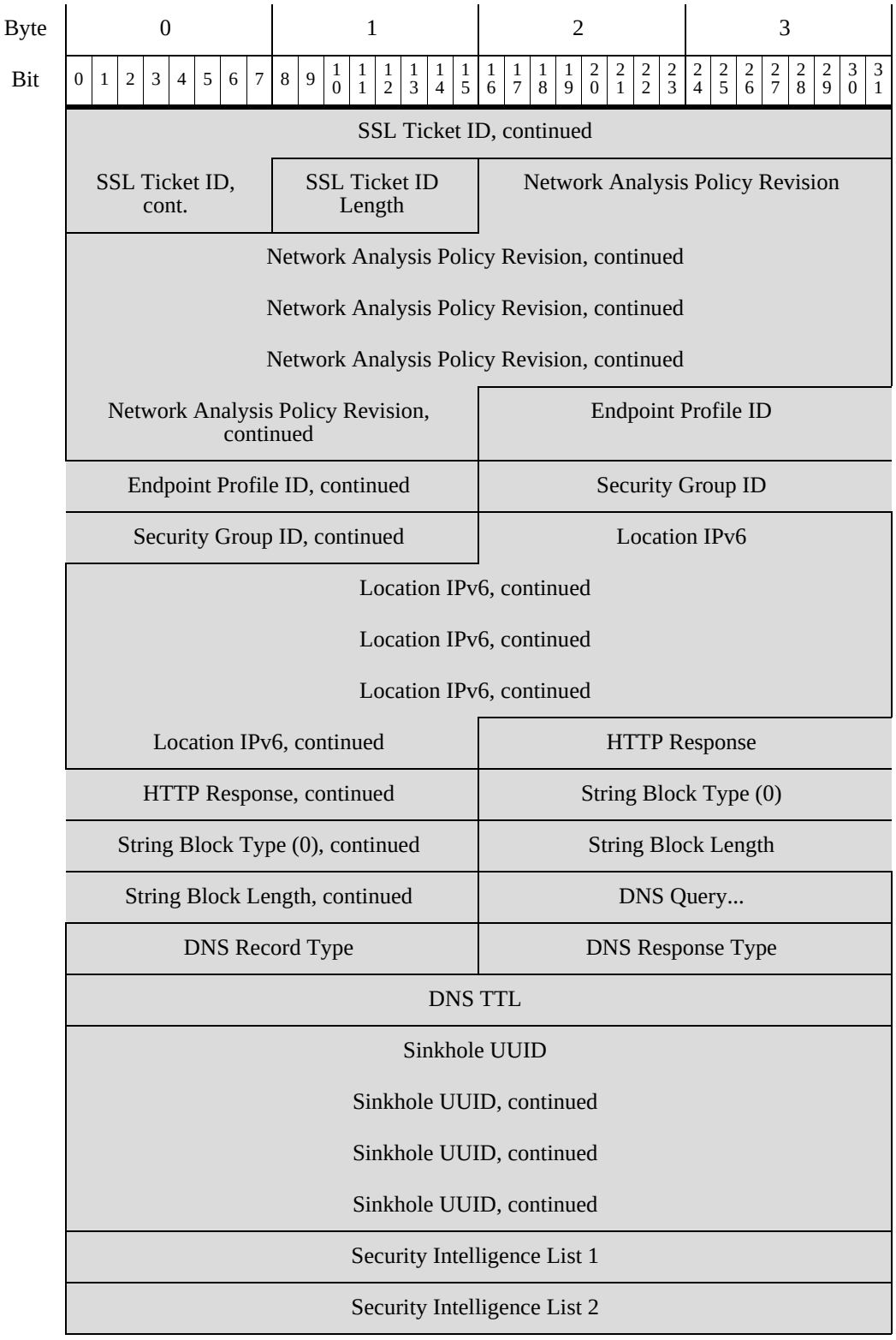
| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Rule ID
Rule Action
Rule Reason
Rule Reason, cont.
Initiator Port
Responder Port
TCP Flags
Protocol
NetFlow Source
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
NetFlow Src, cont.
Instance ID
Connection Counter
Cx Counter, cont.
First Packet Timestamp
First Pkt Time, cont.
Last Packet Timestamp
Last Pkt Time, cont.
Initiator Transmitted Packets
Initiator Transmitted Packets, continued
Initiator Tx Pkt, cont.
Responder Transmitted Packets
Responder Transmitted Packets, continued
Res. Tx Pkts, cont.
Initiator Transmitted Bytes
Initiator Transmitted Bytes, continued
Initiator Tx Bts, cont.
Responder Transmitted Bytes
Responder Transmitted Bytes, continued
Res. Tx Bts, cont.
User ID
User ID, continued
Application Protocol ID
App Prot ID, cont.
URL Category
URL Category, cont.
URL Reputation


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
URL Rep, cont.
Client Application ID
Client App ID, cont.
Web Application ID
Client URL
Web App ID, cont.
String. Block Type (0)
Str. Block Type, cont.
String Block Length
Str. Block Len., cont.
Client App. URL...
NetBIOS Name
String Block Type (0)
String Block Length
NetBIOS Name...
Client App Version
String Block Type (0)
String Block Length
Client Application Version...
Monitor Rule 1
Monitor Rule 2
Monitor Rule 3
Monitor Rule 4
Monitor Rule 5
Monitor Rule 6
Monitor Rule 7
Monitor Rule 8
Sec. Int. Src/Dst
Sec. Int. Layer
File Event Count
Intrusion Event Count
Initiator Country
Responder Country
IOC Number
Source Autonomous System
Destination Autonomous System
SNMP In
SNMP Out


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Source TOS
Destination TOS
Source Mask
Destination Mask
Security Context
Security Context, continued
Security Context, continued
Security Context, continued
Referenced Host
VLAN ID
String Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
Referenced Host...
User Agent
String Block Type (0)
String Block Length
User Agent...
HTTP Referrer
String Block Type (0)
String Block Length
HTTP Referrer...
SSL Certificate Fingerprint
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Policy ID
SSL Policy ID, continued
SSL Policy ID, continued
SSL Policy ID, continued
SSL Rule ID
SSL Cipher Suite
SSL Version
SSL Srv Cert. Stat.


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
SSL Srv Cert. Stat., cont.
SSL Actual Action
SSL actual Action cont.
SSL Expected Action
SSL Flow Status
SSL Flow Status, cont.
SSL Flow Error
SSL Flow Error, cont.
SSL Flow Messages
SSL Flow Msg, cont.
SSL Flow Flags
SSL Flow Flags, cont.
SSL Server Names
SSL Flow Flags, continued
String Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
SSL Server Name...
SSL URL Category
SSL Session ID
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID Length
SSL Ticket ID
SSL Ticket ID, continued
SSL Ticket ID, continued
SSL Ticket ID, continued


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
SSL Ticket ID, continued
SSL Ticket ID, cont.
SSL Ticket ID Length
Network Analysis Policy Revision
Network Analysis Policy Revision, continued
Network Analysis Policy Revision, continued
Network Analysis Policy Revision, continued
Network Analysis Policy Revision, continued
Endpoint Profile ID
Endpoint Profile ID, continued
Security Group ID
Security Group ID, continued
Location IPv6
Location IPv6, continued
Location IPv6, continued
Location IPv6, continued
Location IPv6, continued
HTTP Response
HTTP Response, continued
String Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
DNS Query...
DNS Record Type
DNS Response Type
DNS TTL
Sinkhole UUID
Sinkhole UUID, continued
Sinkhole UUID, continued
Sinkhole UUID, continued
Security Intelligence List 1
Security Intelligence List 2

The following table describes the fields of the Connection Statistics data block for 6.0.x.

***Table B-43 Connection Statistics Data Block 6.0.x Fields***

| Field | Data Type | Description |
|---|---|---|
| Connection Statistics Data Block Type | uint32 | Initiates a Connection Statistics data block for 6.0+. The value is always `160`. |
| Connection Statistics Data Block Length | uint32 | Number of bytes in the Connection Statistics data block, including eight bytes for the connection statistics block type and length fields, plus the number of bytes in the connection data that follows. |
| Device ID | uint32 | The device that detected the connection event. |
| Ingress Zone | uint8[16] | Ingress security zone in the event that triggered the policy violation. |
| Egress Zone | uint8[16] | Egress security zone in the event that triggered the policy violation. |
| Ingress Interface | uint8[16] | Interface for the inbound traffic. |
| Egress Interface | uint8[16] | Interface for the outbound traffic. |
| Initiator IP Address | uint8[16] | IP address of the host that initiated the session described in the connection event, in IP address octets. |
| Responder IP Address | uint8[16] | IP address of the host that responded to the initiating host, in IP address octets. |
| Policy Revision | uint8[16] | Revision number of the rule associated with the triggered correlation event, if applicable. |
| Rule ID | uint32 | Internal identifier for the rule that triggered the event, if applicable. |
| Rule Action | uint16 | The action selected in the user interface for that rule (allow, block, and so forth). |
| Rule Reason | uint32 | The reason the rule triggered the event. |
| Initiator Port | uint16 | Port used by the initiating host. |
| Responder Port | uint16 | Port used by the responding host. |
| TCP Flags | uint16 | Indicates any TCP flags for the connection event. |
| Protocol | uint8 | The IANA-specified protocol number. |
| NetFlow Source | uint8[16] | IP address of the NetFlow-enabled device that exported the data for the connection. |
| Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| First Packet Timestamp | uint32 | UNIX timestamp of the date and time the first packet was exchanged in the session. |
| Last Packet Timestamp | uint32 | UNIX timestamp of the date and time the last packet was exchanged in the session. |

*Table B-43 Connection Statistics Data Block 6.0.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Initiator Transmitted Packets | uint64 | Number of packets transmitted by the initiating host. |
| Responder Transmitted Packets | uint64 | Number of packets transmitted by the responding host. |
| Initiator Transmitted Bytes | uint64 | Number of bytes transmitted by the initiating host. |
| Responder Transmitted Bytes | uint64 | Number of bytes transmitted by the responding host. |
| User ID | uint32 | Internal identification number for the user who last logged into the host that generated the traffic. |
| Application Protocol ID | uint32 | Application ID of the application protocol. |
| URL Category | uint32 | The internal identification number of the URL category. |
| URL Reputation | uint32 | The internal identification number for the URL reputation. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| String Block Type | uint32 | Initiates a String data block for the client application URL. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the client application URL String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the client application URL string. |
| Client Application URL | string | URL the client application accessed, if applicable (`/files/index.html`, for example). |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for the client application version. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block for the client application version, including eight bytes for the string block type and length, plus the number of bytes in the version. |
| Client Application Version | string | Client application version. |

*Table B-43 Connection Statistics Data Block 6.0.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Monitor Rule 1 | uint32 | The ID of the first monitor rule associated with the connection event. |
| Monitor Rule 2 | uint32 | The ID of the second monitor rule associated with the connection event. |
| Monitor Rule 3 | uint32 | The ID of the third monitor rule associated with the connection event. |
| Monitor Rule 4 | uint32 | The ID of the fourth monitor rule associated with the connection event. |
| Monitor Rule 5 | uint32 | The ID of the fifth monitor rule associated with the connection event. |
| Monitor Rule 6 | uint32 | The ID of the sixth monitor rule associated with the connection event. |
| Monitor Rule 7 | uint32 | The ID of the seventh monitor rule associated with the connection event. |
| Monitor Rule 8 | uint32 | The ID of the eighth monitor rule associated with the connection event. |
| Security Intelligence Source/ Destination | uint8 | Whether the source or destination IP address matched the IP block list. |
| Security Intelligence Layer | uint8 | The IP layer that matched the IP block list. |
| File Event Count | uint16 | Value used to distinguish between file events that happen during the same second. |
| Intrusion Event Count | uint16 | Value used to distinguish between intrusion events that happen during the same second. |
| Initiator Country | uint16 | Code for the country of the initiating host. |
| Responder Country | uint 16 | Code for the country of the responding host. |
| IOC Number | uint16 | ID Number of the compromise associated with this event. |
| Source Autonomous System | uint32 | Autonomous system number of the source, either origin or peer. |
| Destination Autonomous System | uint32 | Autonomous system number of the destination, either origin or peer. |
| SNMP Input | uint16 | SNMP index of the input interface. |
| SNMP Output | uint16 | SNMP index of the output interface. |
| Source TOS | uint8 | Type of Service byte setting for the incoming interface. |
| Destination TOS | uint8 | Type of Service byte setting for the outgoing interface. |
| Source Mask | uint8 | Source address prefix mask. |

*Table B-43 Connection Statistics Data Block 6.0.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Destination Mask | uint8 | Destination address prefix mask. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |
| VLAN ID | uint16 | VLAN identification number that indicates which VLAN the host is a member of. |
| String Block Type | uint32 | Initiates a String data block containing the Referenced Host. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Referenced Host String data block, including eight bytes for the block type and header fields plus the number of bytes in the Referenced Host field. |
| Referenced Host | string | Host name information provided in HTTP or DNS. |
| String Block Type | uint32 | Initiates a String data block containing the User Agent. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the User Agent String data block, including eight bytes for the block type and header fields plus the number of bytes in the User Agent field. |
| User Agent | string | Information from the UserAgent header field in the session. |
| String Block Type | uint32 | Initiates a String data block containing the HTTP Referrer. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the HTTP Referrer String data block, including eight bytes for the block type and header fields plus the number of bytes in the HTTP Referrer field. |
| HTTP Referrer | string | The site from which a page originated. This is found int he Referred header information in HTTP traffic. |
| SSL Certificate Fingerprint | uint8[20] | SHA1 hash of the SSL Server certificate. |
| SSL Policy ID | uint8[16] | ID number of the SSL policy that handled the connection. |
| SSL Rule ID | uint32 | ID number of the SSL rule or default action that handled the connection. |
| SSL Cipher Suite | uint16 | Encryption suite used by the SSL connection. The value is stored in decimal format. See `www.iana.org/assignments/tls-parameters/tls-parameters.xhtml` for the cipher suite designated by the value. |
| SSL Version | uint8 | The SSL or TLS protocol version used to encrypt the connection. |

*Table B-43 Connection Statistics Data Block 6.0.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Server Certificate Status | uint32 | The status of the SSL certificate. Possible values include:<br>• 0 — Not checked — The server certificate status was not evaluated.<br>• 1 — Unknown — The server certificate status could not be determined.<br>• 2 — Valid — The server certificate is valid.<br>• 4 — Self-signed — The server certificate is self-signed.<br>• 16 — Invalid Issuer — The server certificate has an invalid issuer.<br>• 32 — Invalid Signature — The server certificate has an invalid signature.<br>• 64 — Expired — The server certificate is expired.<br>• 128 — Not valid yet — The server certificate is not yet valid.<br>• 256 — Revoked — The server certificate has been revoked. |
| SSL Actual Action | uint16 | The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |
| SSL Expected Action | uint16 | The action which should be performed on the connection based on the SSL Rule. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |

*Table B-43 Connection Statistics Data Block 6.0.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Status | uint16 | Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'No Match'<br>• 2 — 'Success'<br>• 3 — 'Uncached Session'<br>• 4 — 'Unknown Cipher Suite'<br>• 5 — 'Unsupported Cipher Suite'<br>• 6 — 'Unsupported SSL Version'<br>• 7 — 'SSL Compression Used'<br>• 8 — 'Session Undecryptable in Passive Mode'<br>• 9 — 'Handshake Error'<br>• 10 — 'Decryption Error'<br>• 11 — 'Pending Server Name Category Lookup'<br>• 12 — 'Pending Common Name Category Lookup'<br>• 13 — 'Internal Error'<br>• 14 — 'Network Parameters Unavailable'<br>• 15 — 'Invalid Server Certificate Handle'<br>• 16 — 'Server Certificate Fingerprint Unavailable'<br>• 17 — 'Cannot Cache Subject DN'<br>• 18 — 'Cannot Cache Issuer DN'<br>• 19 — 'Unknown SSL Version'<br>• 20 — 'External Certificate List Unavailable'<br>• 21 — 'External Certificate Fingerprint Unavailable'<br>• 22 — 'Internal Certificate List Invalid'<br>• 23 — 'Internal Certificate List Unavailable'<br>• 24 — 'Internal Certificate Unavailable'<br>• 25 — 'Internal Certificate Fingerprint Unavailable'<br>• 26 — 'Server Certificate Validation Unavailable'<br>• 27 — 'Server Certificate Validation Failure'<br>• 28 — 'Invalid Action' |
| SSL Flow Error | uint32 | Detailed SSL error code. These values may be needed for support purposes. |

*Table B-43 Connection Statistics Data Block 6.0.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Messages | uint32 | The messages exchanged between client and server during the SSL handshake. See http://tools.ietf.org/html/rfc5246 for more information.<br>• `0x00000001` — NSE_MT__HELLO_REQUEST<br>• `0x00000002` — NSE_MT__CLIENT_ALERT<br>• `0x00000004` — NSE_MT__SERVER_ALERT<br>• `0x00000008` — NSE_MT__CLIENT_HELLO<br>• `0x00000010` — NSE_MT__SERVER_HELLO<br>• `0x00000020` — NSE_MT__SERVER_CERTIFICATE<br>• `0x00000040` — NSE_MT__SERVER_KEY_EXCHANGE<br>• `0x00000080` — NSE_MT__CERTIFICATE_REQUEST<br>• `0x00000100` — NSE_MT__SERVER_HELLO_DONE<br>• `0x00000200` — NSE_MT__CLIENT_CERTIFICATE<br>• `0x00000400` — NSE_MT__CLIENT_KEY_EXCHANGE<br>• `0x00000800` — NSE_MT__CERTIFICATE_VERIFY<br>• `0x00001000` — NSE_MT__CLIENT_CHANGE_CIPHER_SPEC<br>• `0x00002000` — NSE_MT__CLIENT_FINISHED<br>• `0x00004000` — NSE_MT__SERVER_CHANGE_CIPHER_SPEC<br>• `0x00008000` — NSE_MT__SERVER_FINISHED<br>• `0x00010000` — NSE_MT__NEW_SESSION_TICKET<br>• `0x00020000` — NSE_MT__HANDSHAKE_OTHER<br>• `0x00040000` — NSE_MT__APP_DATA_FROM_CLIENT<br>• `0x00080000` — NSE_MT__APP_DATA_FROM_SERVER |
| SSL Flow Flags | uint64 | The debugging level flags for an encrypted connection. Possible values include:<br>• `0x00000001` — NSE_FLOW__VALID - must be set for other fields to be valid<br>• `0x00000002` — NSE_FLOW__INITIALIZED - internal structures ready for processing<br>• `0x00000004` — NSE_FLOW__INTERCEPT - SSL session has been intercepted |
| String Block Type | uint32 | Initiates a String data block containing the SSL Server Name. This value is always `0`. |

*Table B-43 Connection Statistics Data Block 6.0.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the SSL Server Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the SSL Server Name field. |
| SSL Server Name | string | Name provided in the server name indication in the SSL Client Hello. |
| SSL URL Category | uint32 | Category of the flow as identified from the server name and certificate common name. |
| SSL Session ID | uint8[32] | Value of the session ID used during the SSL handshake when the client and server agree to do session reuse |
| SSL Session ID Length | uint8 | Length of the SSL Session ID. While the session ID cannot exceed 32 bytes, it may be less than 32 bytes. |
| SSL Ticket ID | uint8[20] | Hash of the session ticket used when the client and server agree to use a session ticket. |
| SSL Ticket ID Length | uint8 | Length of the SSL Ticket ID. While the ticket ID cannot exceed 20 bytes, it may be less than 20 bytes. |
| Network Analysis Policy revision | uint8[16] | Revision of the Network Analysis Policy associated with the connection event. |
| Endpoint Profile ID | uint32 | ID number of the type of device used by the connection endpoint as identified by ISE. This is unique for each DC and resolved in metadata. |
| Security Group ID | uint32 | ID number assigned to the user by ISE based on policy. |
| Location IPv6 | uint8[16] | IP address of the interface communicating with ISE. Can be IPv4 or IPv6. |
| HTTP Response | uint32 | Response code of the HTTP Request. |
| String Block Type | uint32 | Initiates a String data block for the DNS query. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the DNS query string. |
| DNS Query | string | The content of the query sent to the DNS server. |
| DNS Record Type | uint16 | The numerical value for the type of DNS record. |

*Table B-43 Connection Statistics Data Block 6.0.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| DNS Response Type | uint16 | 0 — NoError — No Error<br>1 — FormErr — Format Error<br>2 — ServFail — Server Failure<br>3 — NXDomain — Non-Existent Domain<br>4 — NotImp — Not Implemented<br>5 — Refused — Query Refused<br>6 — YXDomain — Name Exists when it should not<br>7 — YXRRSet — RR Set Exists when it should not<br>8 — NXRRSet — RR Set that should exist does not<br>9 — NotAuth — Not Authorized<br>10 — NotZone — Name not contained in zone<br>16 — BADSIG — TSIG Signature Failure<br>17 — BADKEY — Key not recognized<br>18 — BADTIME — Signature out of time window<br>19 — BADMODE — Bad TKEY Mode<br>20 — BADNAME — Duplicate key name<br>21 — BADALG — Algorithm not supported<br>22 — BADTRUNC — Bad Truncation<br>3841 — NXDOMAIN — NXDOMAIN response from firewall<br>3842 — SINKHOLE — Sinkhole response from firewall |
| DNS TTL | uint32 | The time to live for the DNS response, in seconds. |
| Sinkhole UUID | uin8[16] | Revision UUID associated with this sinkhole object. |
| Security Intelligence List 1 | uint32 | Security Intelligence List associated with the event. This maps to a Security Intelligence list in associated metadata. There may be two Security Intelligence lists associated with the connection. |
| Security Intelligence List 2 | uint32 | Security Intelligence List associated with the event. This maps to a Security Intelligence list in associated metadata. There may be two Security Intelligence lists associated with the connection. |

# Connection Statistics Data Block 6.1.x

The connection statistics data block is used in connection data messages. Several new fields have been added to the Connection Statistics Data Block for 6.1.x. Fields have been added to support ISE Integration and Multiple Network Maps. The connection statistics data block for version 6.1+ has a block type of 163 in the series 1 group of blocks. It supersedes block type 160, Connection Statistics Data Block 6.0.x, page B-224. New fields have been added to support DNS lookup and Security Intelligence. It is superseded by block type 168, Connection Statistics Data Block 7.1+, page 4-118,

You request connection event records by setting the extended event flag—bit 30 in the Request Flags field—in the request message with an event version of 13 and an event code of 71. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.

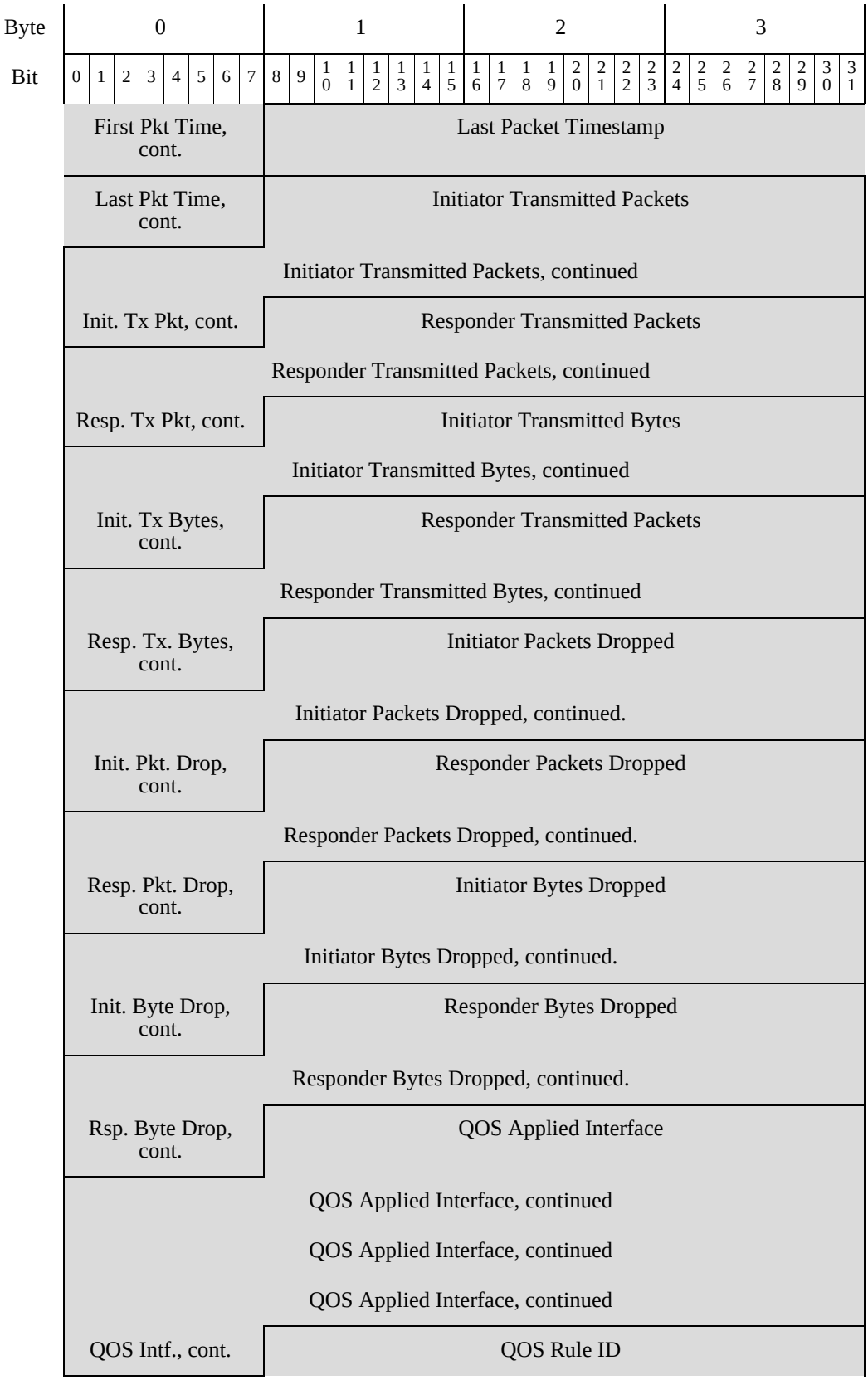
For more information on the Connection Statistics Data message, see Connection Statistics Data Message, page 4-53.

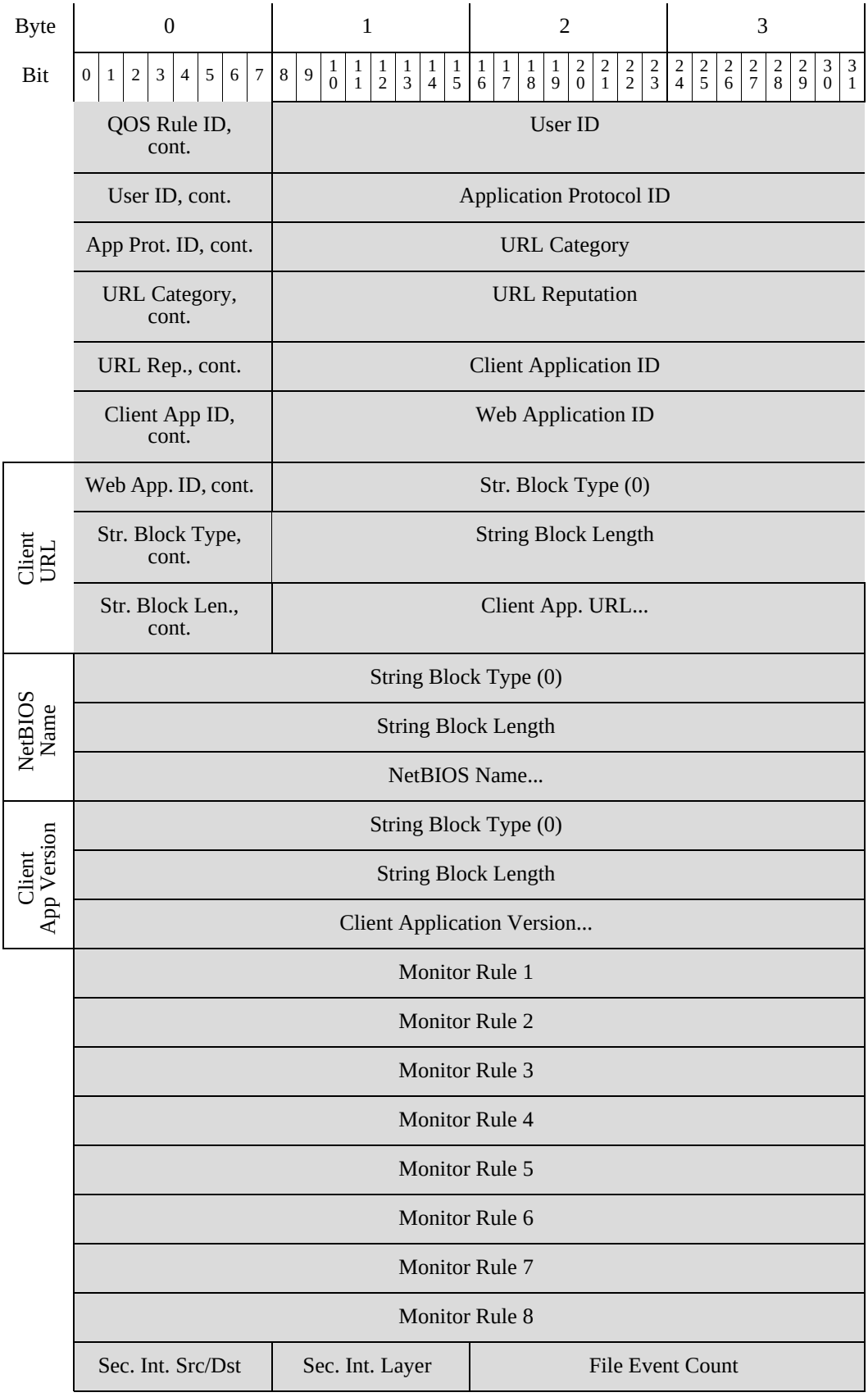
The following diagram shows the format of a Connection Statistics data block for 6.1+:

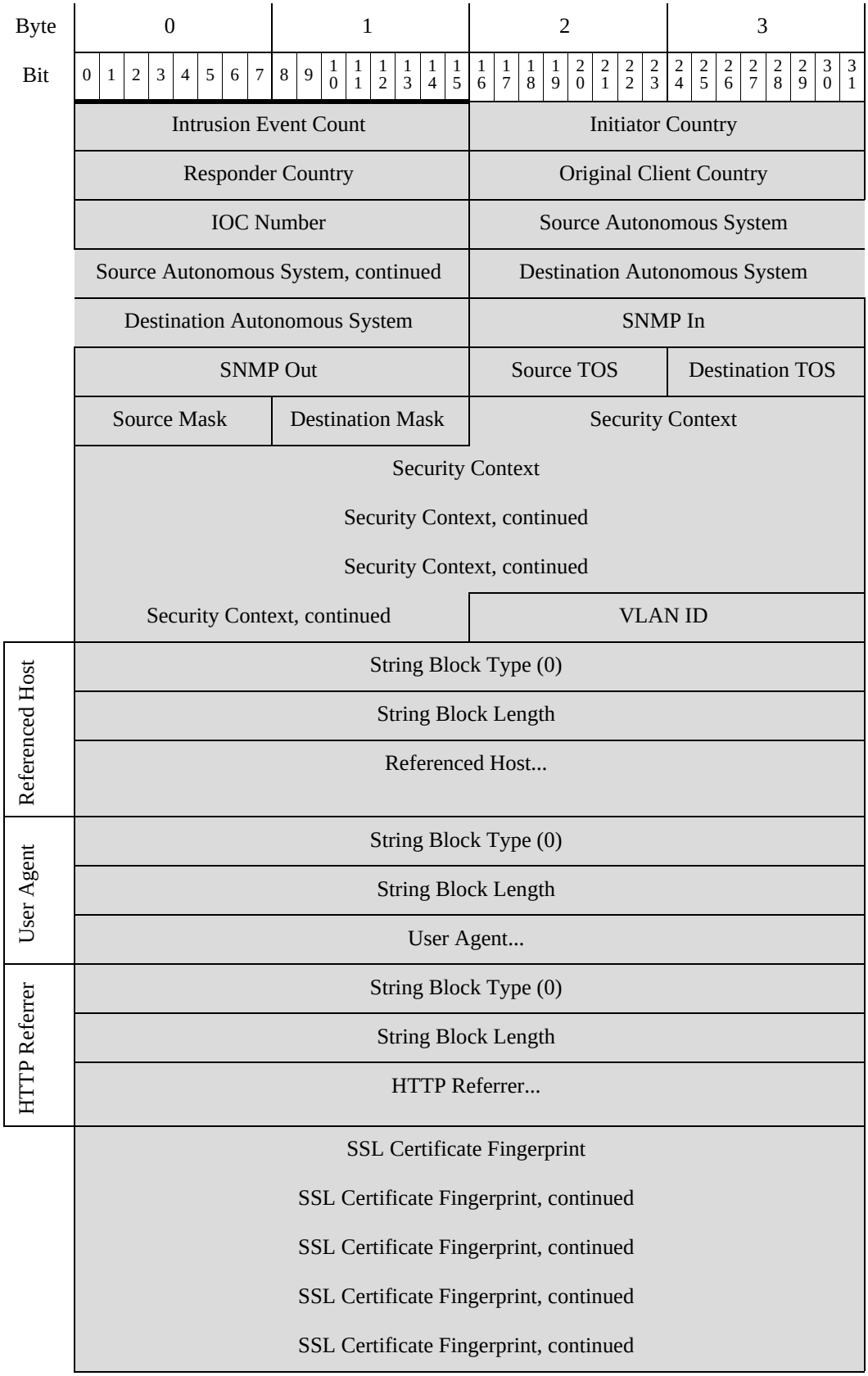
7

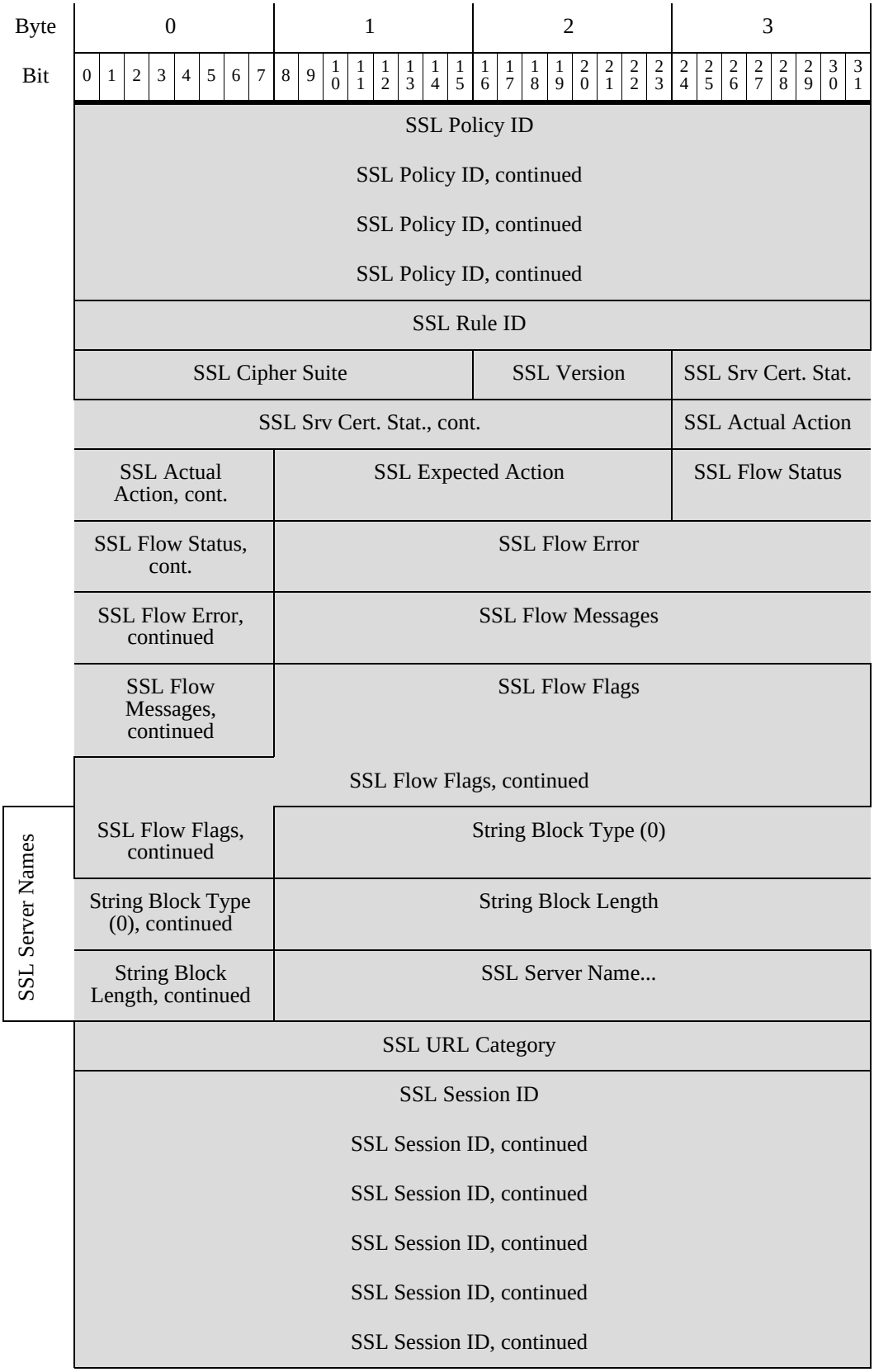
| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Connection Statistics Data Block Type (163) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Connection Statistics Data Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

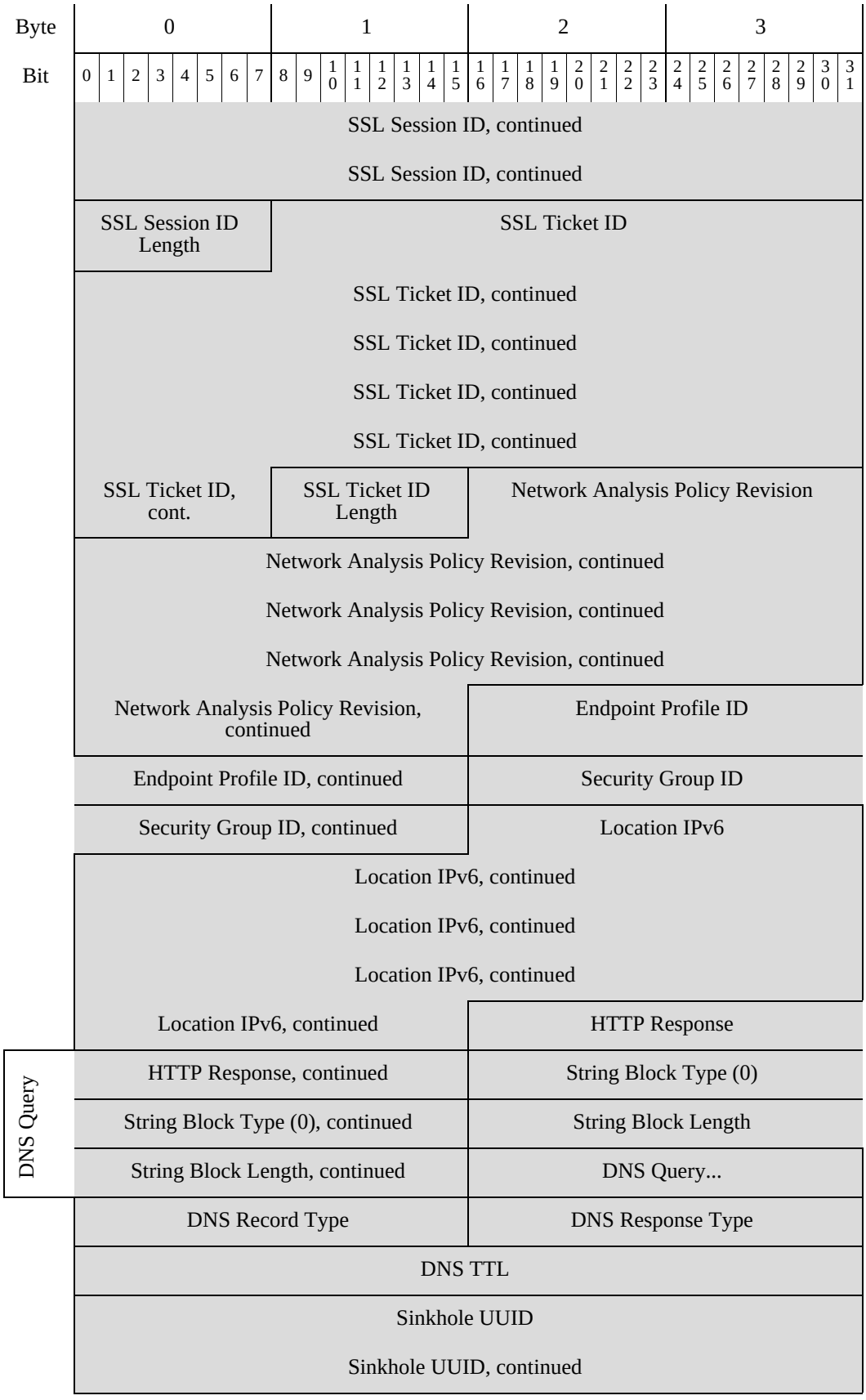
| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Original Client IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Original Client IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Original Client IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Original Client IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Tunnel Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Action | | | | | | | | | | | | | | | | | Rule Reason | | | | | | | | | | | | | | | |
| Rule Reason, cont. | | | | | | | | | | | | | | | | | Initiator Port | | | | | | | | | | | | | | | |
| Responder Port | | | | | | | | | | | | | | | | | TCP Flags | | | | | | | | | | | | | | | |
| Protocol | | | | | | | | | NetFlow Source | | | | | | | | | | | | | | | | | | | | | | | |
| NetFlow Source, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NetFlow Source, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NetFlow Source, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NetFlow Src., cont. | | | | | | | | | Instance ID | | | | | | | | | | | | | | | | Connection Counter | | | | | | | |
| Cx Ctr, cont. | | | | | | | | | First Packet Timestamp | | | | | | | | | | | | | | | | | | | | | | | |

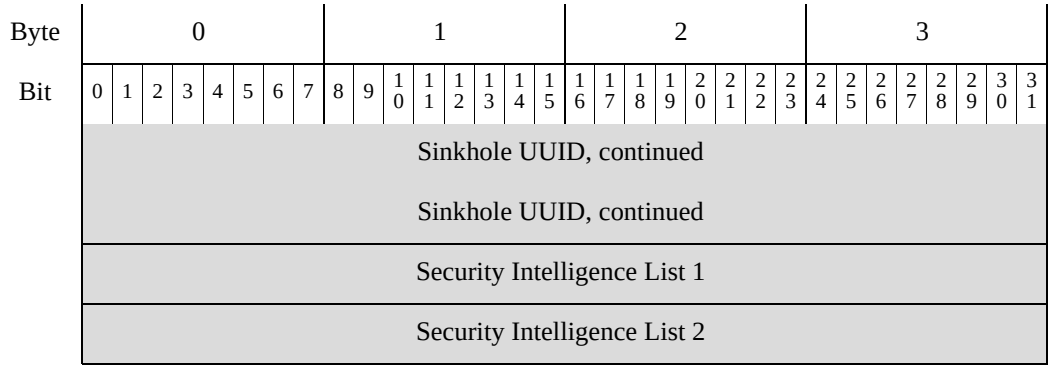

Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
First Pkt Time, cont.
Last Packet Timestamp
Last Pkt Time, cont.
Initiator Transmitted Packets
Initiator Transmitted Packets, continued
Init. Tx Pkt, cont.
Responder Transmitted Packets
Responder Transmitted Packets, continued
Resp. Tx Pkt, cont.
Initiator Transmitted Bytes
Initiator Transmitted Bytes, continued
Init. Tx Bytes, cont.
Responder Transmitted Packets
Responder Transmitted Bytes, continued
Resp. Tx. Bytes, cont.
Initiator Packets Dropped
Initiator Packets Dropped, continued.
Init. Pkt. Drop, cont.
Responder Packets Dropped
Responder Packets Dropped, continued.
Resp. Pkt. Drop, cont.
Initiator Bytes Dropped
Initiator Bytes Dropped, continued.
Init. Byte Drop, cont.
Responder Bytes Dropped
Responder Bytes Dropped, continued.
Rsp. Byte Drop, cont.
QOS Applied Interface
QOS Applied Interface, continued
QOS Applied Interface, continued
QOS Applied Interface, continued
QOS Intf., cont.
QOS Rule ID


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
QOS Rule ID, cont.
User ID
User ID, cont.
Application Protocol ID
App Prot. ID, cont.
URL Category
URL Category, cont.
URL Reputation
URL Rep., cont.
Client Application ID
Client App ID, cont.
Web Application ID
Client URL
Web App. ID, cont.
Str. Block Type (0)
Str. Block Type, cont.
String Block Length
Str. Block Len., cont.
Client App. URL...
NetBIOS Name
String Block Type (0)
String Block Length
NetBIOS Name...
Client App Version
String Block Type (0)
String Block Length
Client Application Version...
Monitor Rule 1
Monitor Rule 2
Monitor Rule 3
Monitor Rule 4
Monitor Rule 5
Monitor Rule 6
Monitor Rule 7
Monitor Rule 8
Sec. Int. Src/Dst
Sec. Int. Layer
File Event Count


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Intrusion Event Count
Initiator Country
Responder Country
Original Client Country
IOC Number
Source Autonomous System
Source Autonomous System, continued
Destination Autonomous System
Destination Autonomous System
SNMP In
SNMP Out
Source TOS
Destination TOS
Source Mask
Destination Mask
Security Context
Security Context
Security Context, continued
Security Context, continued
Security Context, continued
VLAN ID
Referenced Host
String Block Type (0)
String Block Length
Referenced Host...
User Agent
String Block Type (0)
String Block Length
User Agent...
HTTP Referrer
String Block Type (0)
String Block Length
HTTP Referrer...
SSL Certificate Fingerprint
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
SSL Policy ID
SSL Policy ID, continued
SSL Policy ID, continued
SSL Policy ID, continued
SSL Rule ID
SSL Cipher Suite
SSL Version
SSL Srv Cert. Stat.
SSL Srv Cert. Stat., cont.
SSL Actual Action
SSL Actual Action, cont.
SSL Expected Action
SSL Flow Status
SSL Flow Status, cont.
SSL Flow Error
SSL Flow Error, continued
SSL Flow Messages
SSL Flow Messages, continued
SSL Flow Flags
SSL Flow Flags, continued
SSL Server Names
SSL Flow Flags, continued
String Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
SSL Server Name...
SSL URL Category
SSL Session ID
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID Length
SSL Ticket ID
SSL Ticket ID, continued
SSL Ticket ID, continued
SSL Ticket ID, continued
SSL Ticket ID, continued
SSL Ticket ID, cont.
SSL Ticket ID Length
Network Analysis Policy Revision
Network Analysis Policy Revision, continued
Network Analysis Policy Revision, continued
Network Analysis Policy Revision, continued
Network Analysis Policy Revision, continued
Endpoint Profile ID
Endpoint Profile ID, continued
Security Group ID
Security Group ID, continued
Location IPv6
Location IPv6, continued
Location IPv6, continued
Location IPv6, continued
Location IPv6, continued
HTTP Response
DNS Query
HTTP Response, continued
String Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
DNS Query...
DNS Record Type
DNS Response Type
DNS TTL
Sinkhole UUID
Sinkhole UUID, continued




The following table describes the fields of the Connection Statistics data block for 6.1.x.

***Table B-44 Connection Statistics Data Block 6.1+ Fields***

| Field | Data Type | Description |
|---|---|---|
| Connection Statistics Data Block Type | uint32 | Initiates a Connection Statistics data block for 6.1.x. The value is always `163`. |
| Connection Statistics Data Block Length | uint32 | Number of bytes in the Connection Statistics data block, including eight bytes for the connection statistics block type and length fields, plus the number of bytes in the connection data that follows. |
| Device ID | uint32 | The device that detected the connection event. |
| Ingress Zone | uint8[16] | Ingress security zone in the event that triggered the policy violation. |
| Egress Zone | uint8[16] | Egress security zone in the event that triggered the policy violation. |
| Ingress Interface | uint8[16] | Interface for the inbound traffic. |
| Egress Interface | uint8[16] | Interface for the outbound traffic. |
| Initiator IP Address | uint8[16] | IP address of the host that initiated the session described in the connection event, in IP address octets. |
| Responder IP Address | uint8[16] | IP address of the host that responded to the initiating host, in IP address octets. |
| Original Client IP Address | uint8[16] | IP address of the host behind the proxy that originated the request, in IP address octets. |
| Policy Revision | uint8[16] | Revision number of the rule associated with the triggered correlation event, if applicable. |
| Rule ID | uint32 | Internal identifier for the rule that triggered the event, if applicable. |
| Tunnel Rule ID | uint32 | Internal identifier for the tunnel rule that triggered the event, if applicable. |
| Rule Action | uint16 | The action selected in the user interface for that rule (allow, block, and so forth). |

*Table B-44 Connection Statistics Data Block 6.1+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Rule Reason | uint32 | The reason the rule triggered the event. |
| Initiator Port | uint16 | Port used by the initiating host. |
| Responder Port | uint16 | Port used by the responding host. |
| TCP Flags | uint16 | Indicates any TCP flags for the connection event. |
| Protocol | uint8 | The IANA-specified protocol number. |
| NetFlow Source | uint8[16] | IP address of the NetFlow-enabled device that exported the data for the connection. |
| Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| First Packet Timestamp | uint32 | UNIX timestamp of the date and time the first packet was exchanged in the session. |
| Last Packet Timestamp | uint32 | UNIX timestamp of the date and time the last packet was exchanged in the session. |
| Initiator Transmitted Packets | uint64 | Number of packets transmitted by the initiating host. |
| Responder Transmitted Packets | uint64 | Number of packets transmitted by the responding host. |
| Initiator Transmitted Bytes | uint64 | Number of bytes transmitted by the initiating host. |
| Responder Transmitted Bytes | uint64 | Number of bytes transmitted by the responding host. |
| Initiator Packets Dropped | uint64 | Number of packets dropped from the session initiator due to rate limiting. |
| Responder Packets Dropped | uint64 | Number of packets dropped from the session responder due to rate limiting. |
| Initiator Bytes Dropped | uint64 | Number of bytes dropped from the session initiator due to rate limiting. |
| Responder Bytes Dropped | uint64 | Number of bytes dropped from the session responders due to rate limiting. |
| QOS Applied Interface | uint8[16] | For rate-limited connections, the name of the interface on which rate limiting is applied. |
| QOS Rule ID | uint32 | Internal ID number of the Quality of Service rule applied to the connection, if applicable. |
| User ID | uint32 | Internal identification number for the user who last logged into the host that generated the traffic. |
| Application Protocol ID | uint32 | Application ID of the application protocol. |

*Table B-44 Connection Statistics Data Block 6.1+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| URL Category | uint32 | The internal identification number of the URL category. |
| URL Reputation | uint32 | The internal identification number for the URL reputation. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| String Block Type | uint32 | Initiates a String data block for the client application URL. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the client application URL String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the client application URL string. |
| Client Application URL | string | URL the client application accessed, if applicable (`/files/index.html`, for example). |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for the client application version. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block for the client application version, including eight bytes for the string block type and length, plus the number of bytes in the version. |
| Client Application Version | string | Client application version. |
| Monitor Rule 1 | uint32 | The ID of the first monitor rule associated with the connection event. |
| Monitor Rule 2 | uint32 | The ID of the second monitor rule associated with the connection event. |
| Monitor Rule 3 | uint32 | The ID of the third monitor rule associated with the connection event. |
| Monitor Rule 4 | uint32 | The ID of the fourth monitor rule associated with the connection event. |
| Monitor Rule 5 | uint32 | The ID of the fifth monitor rule associated with the connection event. |
| Monitor Rule 6 | uint32 | The ID of the sixth monitor rule associated with the connection event. |
| Monitor Rule 7 | uint32 | The ID of the seventh monitor rule associated with the connection event. |

*Table B-44 Connection Statistics Data Block 6.1+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Monitor Rule 8 | uint32 | The ID of the eighth monitor rule associated with the connection event. |
| Security Intelligence Source/ Destination | uint8 | Whether the source or destination IP address matched the IP block list. |
| Security Intelligence Layer | uint8 | The IP layer that matched the IP block list. |
| File Event Count | uint16 | Value used to distinguish between file events that happen during the same second. |
| Intrusion Event Count | uint16 | Value used to distinguish between intrusion events that happen during the same second. |
| Initiator Country | uint16 | Code for the country of the initiating host. |
| Responder Country | uint 16 | Code for the country of the responding host. |
| Original Client Country | uint 16 | Code for the country of the host behind the proxy which originated the request. |
| IOC Number | uint16 | ID Number of the compromise associated with this event. |
| Source Autonomous System | uint32 | Autonomous system number of the source, either origin or peer. |
| Destination Autonomous System | uint32 | Autonomous system number of the destination, either origin or peer. |
| SNMP Input | uint16 | SNMP index of the input interface. |
| SNMP Output | uint16 | SNMP index of the output interface. |
| Source TOS | uint8 | Type of Service byte setting for the incoming interface. |
| Destination TOS | uint8 | Type of Service byte setting for the outgoing interface. |
| Source Mask | uint8 | Source address prefix mask. |
| Destination Mask | uint8 | Destination address prefix mask. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |
| VLAN ID | uint16 | VLAN identification number that indicates which VLAN the host is a member of. |
| String Block Type | uint32 | Initiates a String data block containing the Referenced Host. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Referenced Host String data block, including eight bytes for the block type and header fields plus the number of bytes in the Referenced Host field. |
| Referenced Host | string | Host name information provided in HTTP or DNS. |

*Table B-44 Connection Statistics Data Block 6.1+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Type | uint32 | Initiates a String data block containing the User Agent. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the User Agent String data block, including eight bytes for the block type and header fields plus the number of bytes in the User Agent field. |
| User Agent | string | Information from the UserAgent header field in the session. |
| String Block Type | uint32 | Initiates a String data block containing the HTTP Referrer. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the HTTP Referrer String data block, including eight bytes for the block type and header fields plus the number of bytes in the HTTP Referrer field. |
| HTTP Referrer | string | The site from which a page originated. This is found int he Referred header information in HTTP traffic. |
| SSL Certificate Fingerprint | uint8[20] | SHA1 hash of the SSL Server certificate. |
| SSL Policy ID | uint8[16] | ID number of the SSL policy that handled the connection. |
| SSL Rule ID | uint32 | ID number of the SSL rule or default action that handled the connection. |
| SSL Cipher Suite | uint16 | Encryption suite used by the SSL connection. The value is stored in decimal format. See `www.iana.org/assignments/tls-parameters/tls-parameters.xhtml` for the cipher suite designated by the value. |
| SSL Version | uint8 | The SSL or TLS protocol version used to encrypt the connection. |
| SSL Server Certificate Status | uint32 | The status of the SSL certificate. Possible values include:<br>• `0` — Not checked — The server certificate status was not evaluated.<br>• `1` — Unknown — The server certificate status could not be determined.<br>• `2` — Valid — The server certificate is valid.<br>• `4` — Self-signed — The server certificate is self-signed.<br>• `16` — Invalid Issuer — The server certificate has an invalid issuer.<br>• `32` — Invalid Signature — The server certificate has an invalid signature.<br>• `64` — Expired — The server certificate is expired.<br>• `128` — Not valid yet — The server certificate is not yet valid.<br>• `256` — Revoked — The server certificate has been revoked. |

*Table B-44 Connection Statistics Data Block 6.1+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Actual Action | uint16 | The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |
| SSL Expected Action | uint16 | The action which should be performed on the connection based on the SSL Rule. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |

*Table B-44 Connection Statistics Data Block 6.1+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Status | uint16 | Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'No Match'<br>• 2 — 'Success'<br>• 3 — 'Uncached Session'<br>• 4 — 'Unknown Cipher Suite'<br>• 5 — 'Unsupported Cipher Suite'<br>• 6 — 'Unsupported SSL Version'<br>• 7 — 'SSL Compression Used'<br>• 8 — 'Session Undecryptable in Passive Mode'<br>• 9 — 'Handshake Error'<br>• 10 — 'Decryption Error'<br>• 11 — 'Pending Server Name Category Lookup'<br>• 12 — 'Pending Common Name Category Lookup'<br>• 13 — 'Internal Error'<br>• 14 — 'Network Parameters Unavailable'<br>• 15 — 'Invalid Server Certificate Handle'<br>• 16 — 'Server Certificate Fingerprint Unavailable'<br>• 17 — 'Cannot Cache Subject DN'<br>• 18 — 'Cannot Cache Issuer DN'<br>• 19 — 'Unknown SSL Version'<br>• 20 — 'External Certificate List Unavailable'<br>• 21 — 'External Certificate Fingerprint Unavailable'<br>• 22 — 'Internal Certificate List Invalid'<br>• 23 — 'Internal Certificate List Unavailable'<br>• 24 — 'Internal Certificate Unavailable'<br>• 25 — 'Internal Certificate Fingerprint Unavailable'<br>• 26 — 'Server Certificate Validation Unavailable'<br>• 27 — 'Server Certificate Validation Failure'<br>• 28 — 'Invalid Action' |
| SSL Flow Error | uint32 | Detailed SSL error code. These values may be needed for support purposes. |

*Table B-44 Connection Statistics Data Block 6.1+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Messages | uint32 | The messages exchanged between client and server during the SSL handshake. See http://tools.ietf.org/html/rfc5246 for more information.<br>• `0x00000001` — NSE_MT__HELLO_REQUEST<br>• `0x00000002` — NSE_MT__CLIENT_ALERT<br>• `0x00000004` — NSE_MT__SERVER_ALERT<br>• `0x00000008` — NSE_MT__CLIENT_HELLO<br>• `0x00000010` — NSE_MT__SERVER_HELLO<br>• `0x00000020` — NSE_MT__SERVER_CERTIFICATE<br>• `0x00000040` — NSE_MT__SERVER_KEY_EXCHANGE<br>• `0x00000080` — NSE_MT__CERTIFICATE_REQUEST<br>• `0x00000100` — NSE_MT__SERVER_HELLO_DONE<br>• `0x00000200` — NSE_MT__CLIENT_CERTIFICATE<br>• `0x00000400` — NSE_MT__CLIENT_KEY_EXCHANGE<br>• `0x00000800` — NSE_MT__CERTIFICATE_VERIFY<br>• `0x00001000` — NSE_MT__CLIENT_CHANGE_CIPHER_SPEC<br>• `0x00002000` — NSE_MT__CLIENT_FINISHED<br>• `0x00004000` — NSE_MT__SERVER_CHANGE_CIPHER_SPEC<br>• `0x00008000` — NSE_MT__SERVER_FINISHED<br>• `0x00010000` — NSE_MT__NEW_SESSION_TICKET<br>• `0x00020000` — NSE_MT__HANDSHAKE_OTHER<br>• `0x00040000` — NSE_MT__APP_DATA_FROM_CLIENT<br>• `0x00080000` — NSE_MT__APP_DATA_FROM_SERVER |
| SSL Flow Flags | uint64 | The debugging level flags for an encrypted connection. Possible values include:<br>• `0x00000001` — NSE_FLOW__VALID - must be set for other fields to be valid<br>• `0x00000002` — NSE_FLOW__INITIALIZED - internal structures ready for processing<br>• `0x00000004` — NSE_FLOW__INTERCEPT - SSL session has been intercepted |
| String Block Type | uint32 | Initiates a String data block containing the SSL Server Name. This value is always `0`. |

*Table B-44 Connection Statistics Data Block 6.1+ Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the SSL Server Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the SSL Server Name field. |
| SSL Server Name | string | Name provided in the server name indication in the SSL Client Hello. |
| SSL URL Category | uint32 | Category of the flow as identified from the server name and certificate common name. |
| SSL Session ID | uint8[32] | Value of the session ID used during the SSL handshake when the client and server agree to do session reuse |
| SSL Session ID Length | uint8 | Length of the SSL Session ID. While the session ID cannot exceed 32 bytes, it may be less than 32 bytes. |
| SSL Ticket ID | uint8[20] | Hash of the session ticket used when the client and server agree to use a session ticket. |
| SSL Ticket ID Length | uint8 | Length of the SSL Ticket ID. While the ticket ID cannot exceed 20 bytes, it may be less than 20 bytes. |
| Network Analysis Policy revision | uint8[16] | Revision of the Network Analysis Policy associated with the connection event. |
| Endpoint Profile ID | uint32 | ID number of the type of device used by the connection endpoint as identified by ISE. This is unique for each DC and resolved in metadata. |
| Security Group ID | uint32 | ID number assigned to the user by ISE based on policy. |
| Location IPv6 | uint8[16] | IP address of the interface communicating with ISE. Can be IPv4 or IPv6. |
| HTTP Response | uint32 | Response code of the HTTP Request. |
| String Block Type | uint32 | Initiates a String data block for the DNS query. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the DNS query string. |
| DNS Query | string | The content of the query sent to the DNS server. |
| DNS Record Type | uint16 | The numerical value for the type of DNS record. |
| DNS Response Type | uint16 | The numerical value for the type of DNS response. |
| DNS TTL | uint32 | The time to live for the DNS response, in seconds. |
| Sinkhole UUID | uin8[16] | Revision UUID associated with this sinkhole object. |
| Security Intelligence List 1 | uint32 | Security Intelligence List associated with the event. This maps to a Security Intelligence list in associated metadata. There may be two Security Intelligence lists associated with the connection. |
| Security Intelligence List 2 | uint32 | Security Intelligence List associated with the event. This maps to a Security Intelligence list in associated metadata. There may be two Security Intelligence lists associated with the connection. |

# Connection Statistics Data Block 6.2-6.7.x

The connection statistics data block is used in connection data messages. A third Security Intelligence field has been added to Connection Statistics Data Block for 6.2-6.7.x. The connection statistics data block for version 6.2-6.7.x has a block type of 168 in the series 1 group of blocks. It supersedes block type 163, Connection Statistics Data Block 6.1.x, page B-239. It is superseded by block type 173.

You request connection event records by setting the extended event flag—bit 30 in the Request Flags field—in the request message with an event version of 15 and an event code of 71. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.

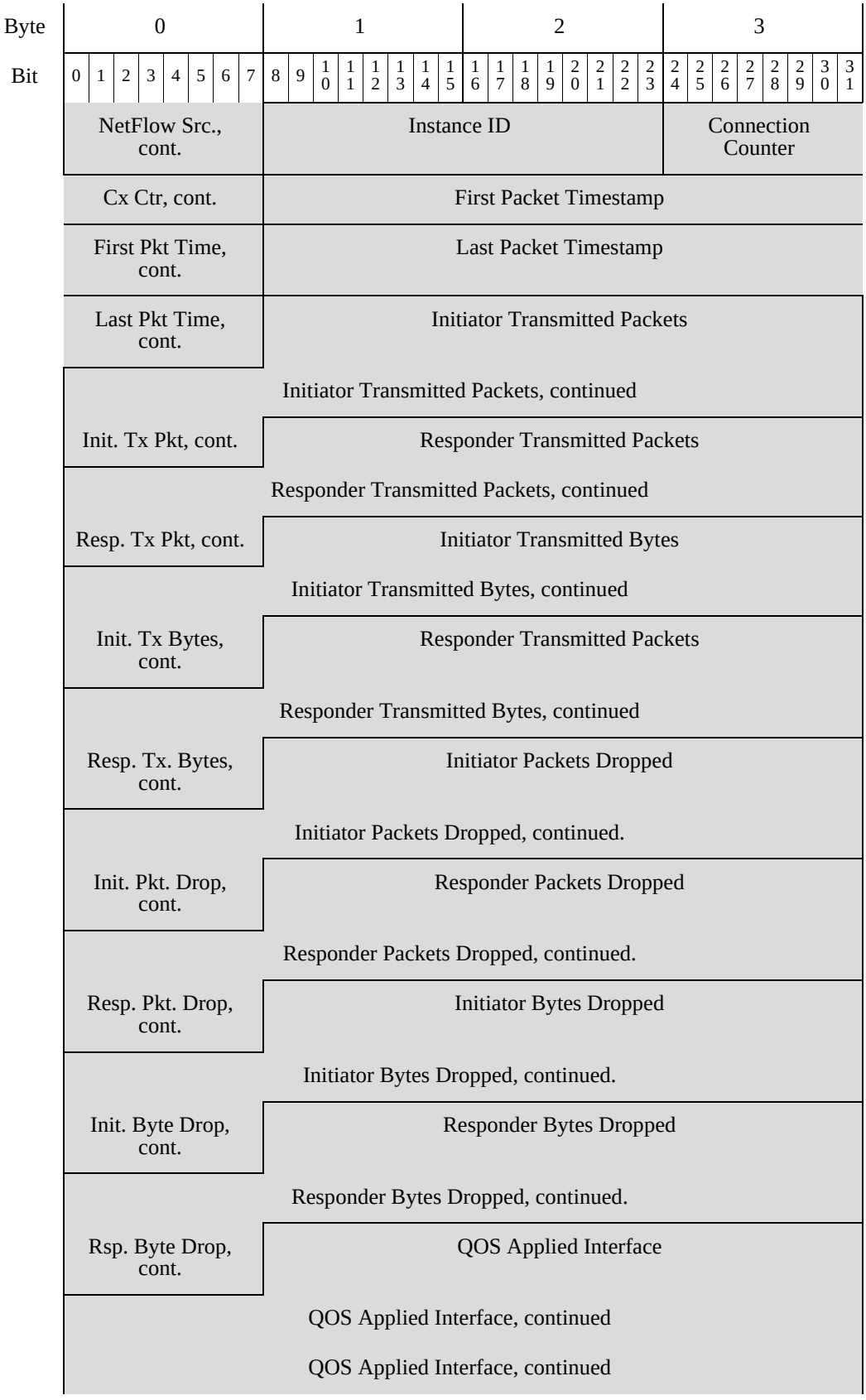
For more information on the Connection Statistics Data message, see Connection Statistics Data Message, page 4-53.

The following diagram shows the format of a Connection Statistics data block for 6.2-6.7.x:

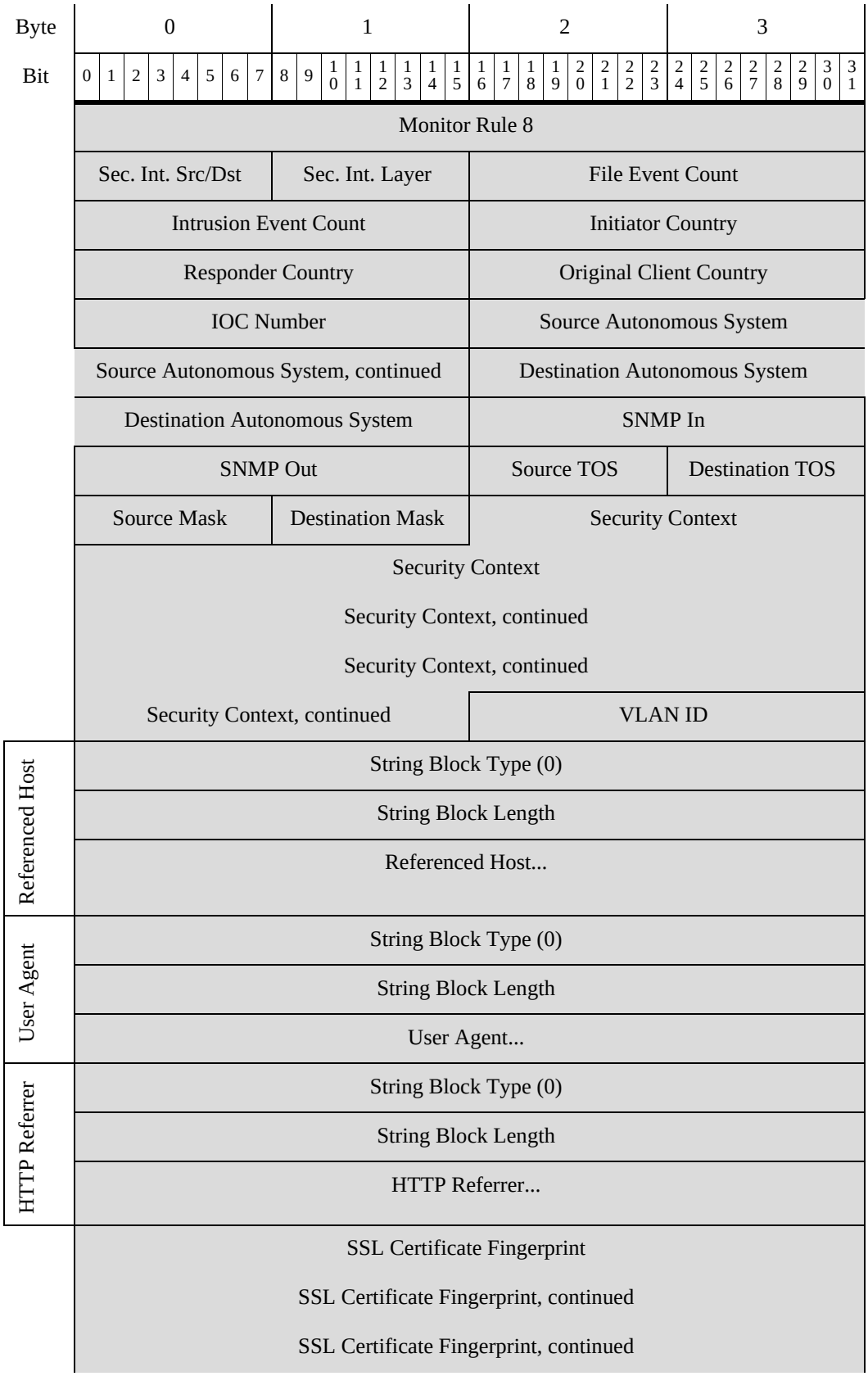
7

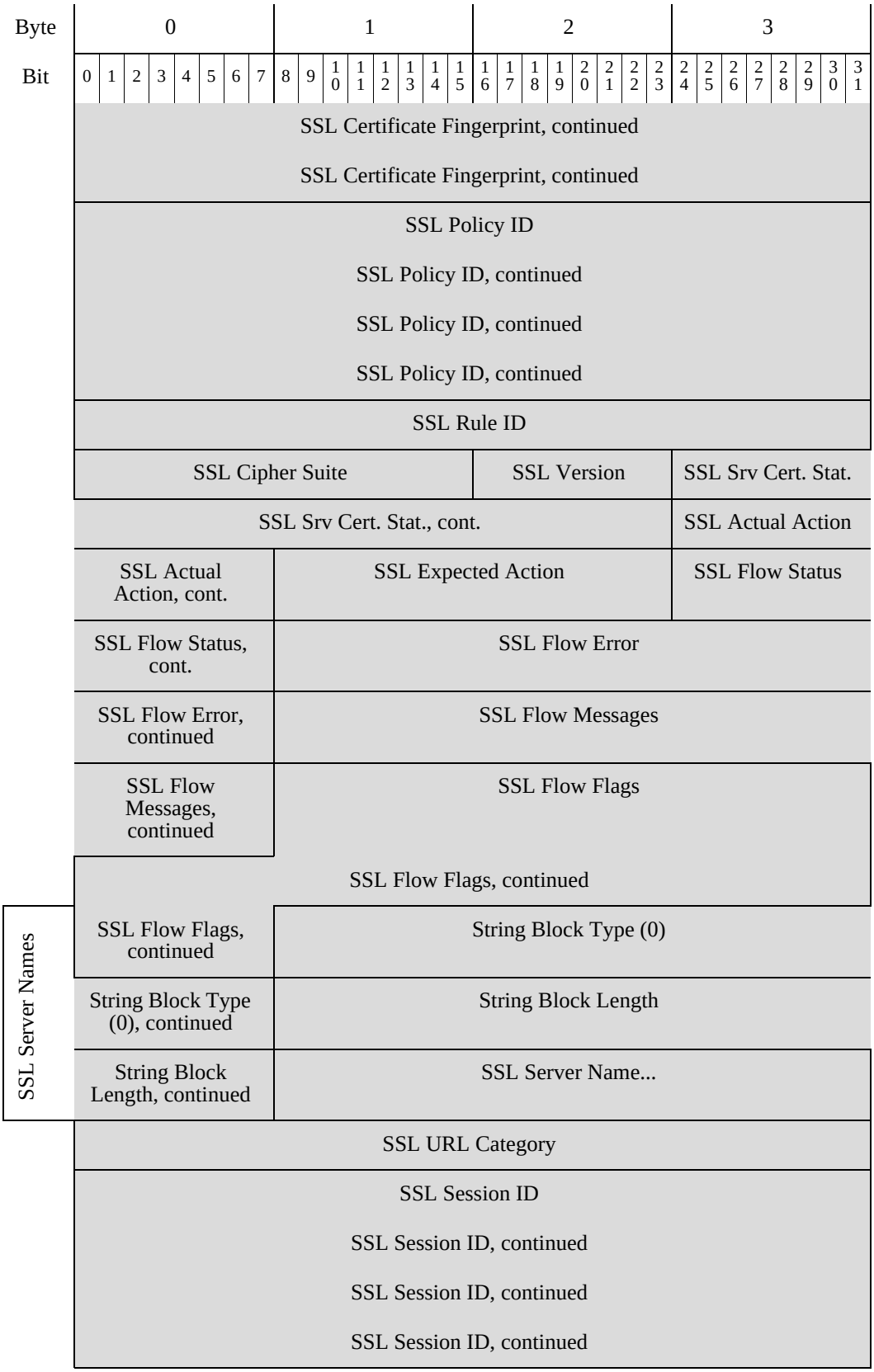
| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Connection Statistics Data Block Type (168) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Connection Statistics Data Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Zone, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

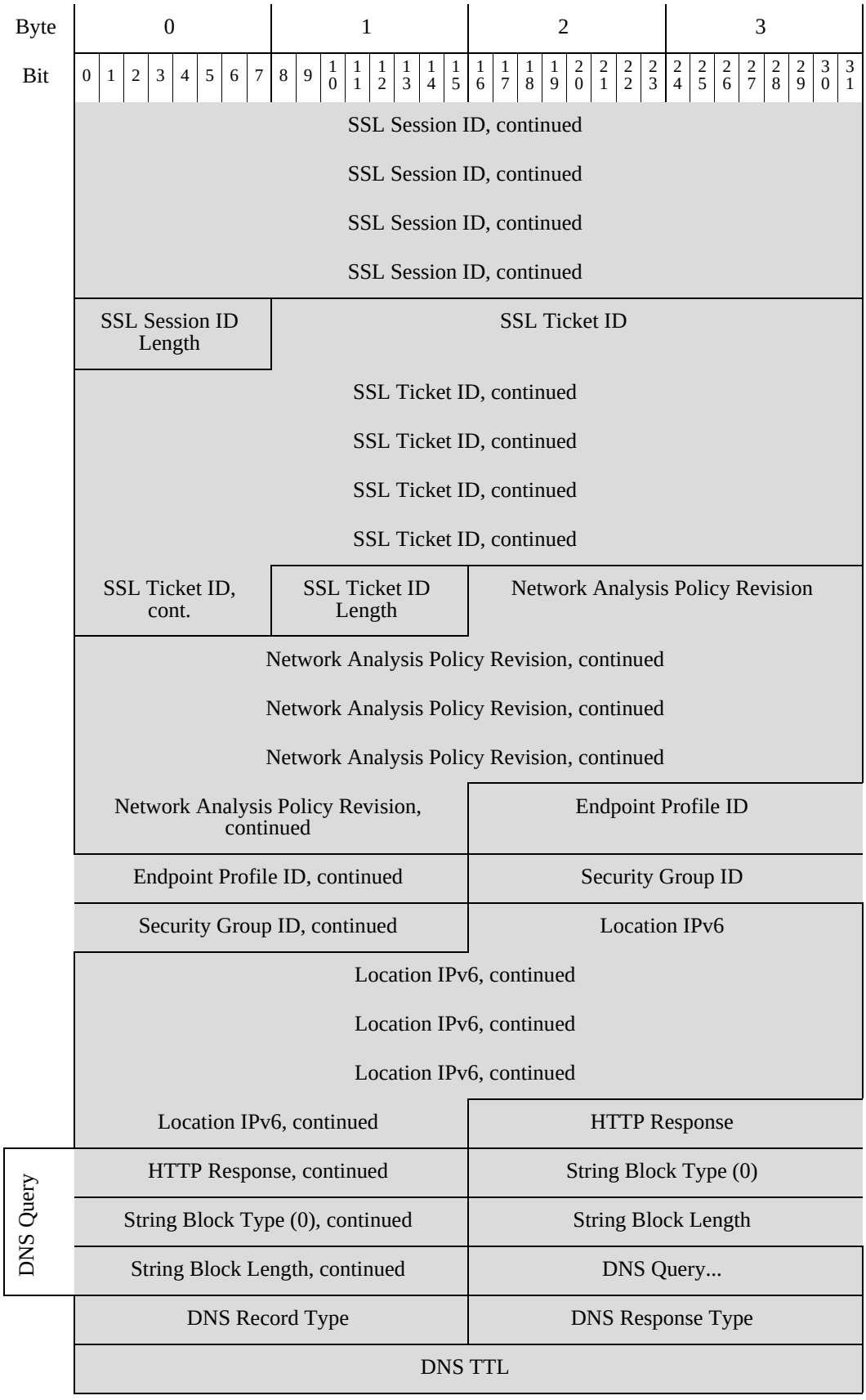
| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Original Client IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Original Client IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Original Client IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Original Client IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Tunnel Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule Action | | | | | | | | | | | | | | | | | Rule Reason | | | | | | | | | | | | | | | |
| Rule Reason, cont. | | | | | | | | | | | | | | | | | Initiator Port | | | | | | | | | | | | | | | |
| Responder Port | | | | | | | | | | | | | | | | | TCP Flags | | | | | | | | | | | | | | | |
| Protocol | | | | | | | | | NetFlow Source | | | | | | | | | | | | | | | | | | | | | | | |
| NetFlow Source, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NetFlow Source, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NetFlow Source, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

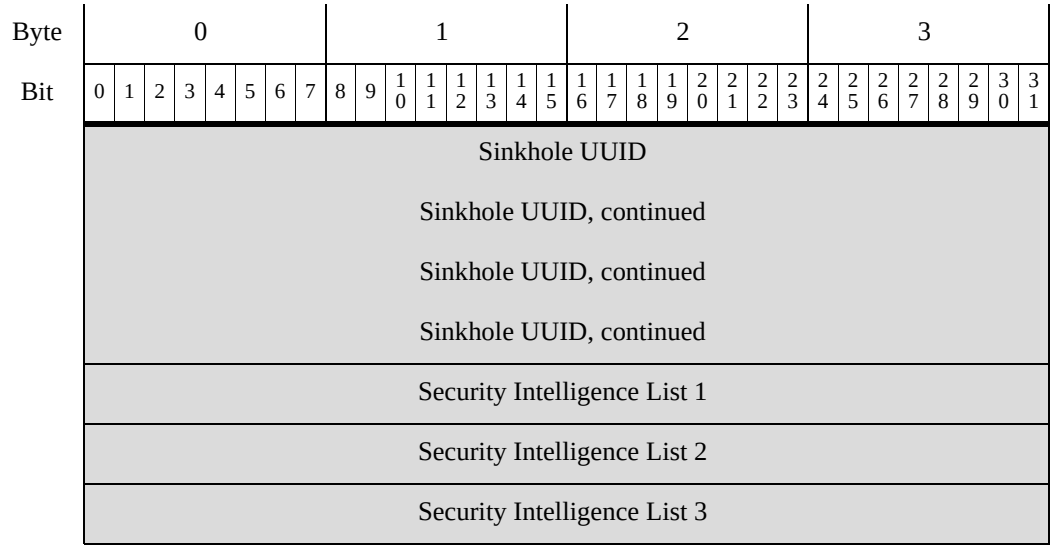

Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
NetFlow Src., cont.
Instance ID
Connection Counter
Cx Ctr, cont.
First Packet Timestamp
First Pkt Time, cont.
Last Packet Timestamp
Last Pkt Time, cont.
Initiator Transmitted Packets
Initiator Transmitted Packets, continued
Init. Tx Pkt, cont.
Responder Transmitted Packets
Responder Transmitted Packets, continued
Resp. Tx Pkt, cont.
Initiator Transmitted Bytes
Initiator Transmitted Bytes, continued
Init. Tx Bytes, cont.
Responder Transmitted Packets
Responder Transmitted Bytes, continued
Resp. Tx. Bytes, cont.
Initiator Packets Dropped
Initiator Packets Dropped, continued.
Init. Pkt. Drop, cont.
Responder Packets Dropped
Responder Packets Dropped, continued.
Resp. Pkt. Drop, cont.
Initiator Bytes Dropped
Initiator Bytes Dropped, continued.
Init. Byte Drop, cont.
Responder Bytes Dropped
Responder Bytes Dropped, continued.
Rsp. Byte Drop, cont.
QOS Applied Interface
QOS Applied Interface, continued
QOS Applied Interface, continued

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | QOS Applied Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | QOS Intf., cont. | | | | | | | | QOS Rule ID | | | | | | | | | | | | | | | | | | | | | | | |
| | QOS Rule ID, cont. | | | | | | | | User ID | | | | | | | | | | | | | | | | | | | | | | | |
| | User ID, cont. | | | | | | | | Application Protocol ID | | | | | | | | | | | | | | | | | | | | | | | |
| | App Prot. ID, cont. | | | | | | | | URL Category | | | | | | | | | | | | | | | | | | | | | | | |
| | URL Category, cont. | | | | | | | | URL Reputation | | | | | | | | | | | | | | | | | | | | | | | |
| | URL Rep., cont. | | | | | | | | Client Application ID | | | | | | | | | | | | | | | | | | | | | | | |
| | Client App ID, cont. | | | | | | | | Web Application ID | | | | | | | | | | | | | | | | | | | | | | | |
| Client URL | Web App. ID, cont. | | | | | | | | Str. Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | |
| Client URL | Str. Block Type, cont. | | | | | | | | String Block Length | | | | | | | | | | | | | | | | | | | | | | | |
| Client URL | Str. Block Len., cont. | | | | | | | | Client App. URL... | | | | | | | | | | | | | | | | | | | | | | | |
| NetBIOS Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NetBIOS Name | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NetBIOS Name | NetBIOS Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client App Version | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client App Version | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client App Version | Client Application Version... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Monitor Rule 7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Monitor Rule 8
Sec. Int. Src/Dst
Sec. Int. Layer
File Event Count
Intrusion Event Count
Initiator Country
Responder Country
Original Client Country
IOC Number
Source Autonomous System
Source Autonomous System, continued
Destination Autonomous System
Destination Autonomous System
SNMP In
SNMP Out
Source TOS
Destination TOS
Source Mask
Destination Mask
Security Context
Security Context
Security Context, continued
Security Context, continued
Security Context, continued
VLAN ID
Referenced Host
String Block Type (0)
String Block Length
Referenced Host...
User Agent
String Block Type (0)
String Block Length
User Agent...
HTTP Referrer
String Block Type (0)
String Block Length
HTTP Referrer...
SSL Certificate Fingerprint
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Policy ID
SSL Policy ID, continued
SSL Policy ID, continued
SSL Policy ID, continued
SSL Rule ID
SSL Cipher Suite
SSL Version
SSL Srv Cert. Stat.
SSL Srv Cert. Stat., cont.
SSL Actual Action
SSL Actual Action, cont.
SSL Expected Action
SSL Flow Status
SSL Flow Status, cont.
SSL Flow Error
SSL Flow Error, continued
SSL Flow Messages
SSL Flow Messages, continued
SSL Flow Flags
SSL Flow Flags, continued
SSL Server Names
SSL Flow Flags, continued
String Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
SSL Server Name...
SSL URL Category
SSL Session ID
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID Length
SSL Ticket ID
SSL Ticket ID, continued
SSL Ticket ID, continued
SSL Ticket ID, continued
SSL Ticket ID, continued
SSL Ticket ID, cont.
SSL Ticket ID Length
Network Analysis Policy Revision
Network Analysis Policy Revision, continued
Network Analysis Policy Revision, continued
Network Analysis Policy Revision, continued
Network Analysis Policy Revision, continued
Endpoint Profile ID
Endpoint Profile ID, continued
Security Group ID
Security Group ID, continued
Location IPv6
Location IPv6, continued
Location IPv6, continued
Location IPv6, continued
Location IPv6, continued
HTTP Response
DNS Query
HTTP Response, continued
String Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
DNS Query...
DNS Record Type
DNS Response Type
DNS TTL




The following table describes the fields of the Connection Statistics data block for 6.2-6.7.x.

***Table B-45 Connection Statistics Data Block 6.2-6.7.x Fields***

| Field | Data Type | Description |
|---|---|---|
| Connection Statistics Data Block Type | uint32 | Initiates a Connection Statistics data block for 6.2-6.7.x. The value is always `168`. |
| Connection Statistics Data Block Length | uint32 | Number of bytes in the Connection Statistics data block, including eight bytes for the connection statistics block type and length fields, plus the number of bytes in the connection data that follows. |
| Device ID | uint32 | The device that detected the connection event. |
| Ingress Zone | uint8[16] | Ingress security zone in the event that triggered the policy violation. |
| Egress Zone | uint8[16] | Egress security zone in the event that triggered the policy violation. |
| Ingress Interface | uint8[16] | Interface for the inbound traffic. |
| Egress Interface | uint8[16] | Interface for the outbound traffic. |
| Initiator IP Address | uint8[16] | IP address of the host that initiated the session described in the connection event, in IP address octets. |
| Responder IP Address | uint8[16] | IP address of the host that responded to the initiating host, in IP address octets. |
| Original Client IP Address | uint8[16] | IP address of the host behind the proxy that originated the request, in IP address octets. |
| Policy Revision | uint8[16] | Revision number of the rule associated with the triggered correlation event, if applicable. |
| Rule ID | uint32 | Internal identifier for the rule that triggered the event, if applicable. |

*Table B-45 Connection Statistics Data Block 6.2-6.7.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Tunnel Rule ID | uint32 | Internal identifier for the tunnel rule that triggered the event, if applicable. |
| Rule Action | uint16 | The action selected in the user interface for that rule (allow, block, and so forth). |
| Rule Reason | uint32 | The reason the rule triggered the event. |
| Initiator Port | uint16 | Port used by the initiating host. |
| Responder Port | uint16 | Port used by the responding host. |
| TCP Flags | uint16 | Indicates any TCP flags for the connection event. |
| Protocol | uint8 | The IANA-specified protocol number. |
| NetFlow Source | uint8[16] | IP address of the NetFlow-enabled device that exported the data for the connection. |
| Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| First Packet Timestamp | uint32 | UNIX timestamp of the date and time the first packet was exchanged in the session. |
| Last Packet Timestamp | uint32 | UNIX timestamp of the date and time the last packet was exchanged in the session. |
| Initiator Transmitted Packets | uint64 | Number of packets transmitted by the initiating host. |
| Responder Transmitted Packets | uint64 | Number of packets transmitted by the responding host. |
| Initiator Transmitted Bytes | uint64 | Number of bytes transmitted by the initiating host. |
| Responder Transmitted Bytes | uint64 | Number of bytes transmitted by the responding host. |
| Initiator Packets Dropped | uint64 | Number of packets dropped from the session initiator due to rate limiting. |
| Responder Packets Dropped | uint64 | Number of packets dropped from the session responder due to rate limiting. |
| Initiator Bytes Dropped | uint64 | Number of bytes dropped from the session initiator due to rate limiting. |
| Responder Bytes Dropped | uint64 | Number of bytes dropped from the session responders due to rate limiting. |
| QOS Applied Interface | uint8[16] | For rate-limited connections, the name of the interface on which rate limiting is applied. |
| QOS Rule ID | uint32 | Internal ID number of the Quality of Service rule applied to the connection, if applicable. |

*Table B-45 Connection Statistics Data Block 6.2-6.7.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| User ID | uint32 | Internal identification number for the user who last logged into the host that generated the traffic. |
| Application Protocol ID | uint32 | Application ID of the application protocol. |
| URL Category | uint32 | The internal identification number of the URL category. |
| URL Reputation | uint32 | The internal identification number for the URL reputation. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| String Block Type | uint32 | Initiates a String data block for the client application URL. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the client application URL String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the client application URL string. |
| Client Application URL | string | URL the client application accessed, if applicable (`/files/index.html`, for example). |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for the client application version. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block for the client application version, including eight bytes for the string block type and length, plus the number of bytes in the version. |
| Client Application Version | string | Client application version. |
| Monitor Rule 1 | uint32 | The ID of the first monitor rule associated with the connection event. |
| Monitor Rule 2 | uint32 | The ID of the second monitor rule associated with the connection event. |
| Monitor Rule 3 | uint32 | The ID of the third monitor rule associated with the connection event. |
| Monitor Rule 4 | uint32 | The ID of the fourth monitor rule associated with the connection event. |
| Monitor Rule 5 | uint32 | The ID of the fifth monitor rule associated with the connection event. |

*Table B-45 Connection Statistics Data Block 6.2-6.7.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Monitor Rule 6 | uint32 | The ID of the sixth monitor rule associated with the connection event. |
| Monitor Rule 7 | uint32 | The ID of the seventh monitor rule associated with the connection event. |
| Monitor Rule 8 | uint32 | The ID of the eighth monitor rule associated with the connection event. |
| Security Intelligence Source/ Destination | uint8 | Whether the source or destination IP address matched the IP block list. |
| Security Intelligence Layer | uint8 | The IP layer that matched the IP block list. |
| File Event Count | uint16 | Value used to distinguish between file events that happen during the same second. |
| Intrusion Event Count | uint16 | Value used to distinguish between intrusion events that happen during the same second. |
| Initiator Country | uint16 | Code for the country of the initiating host. |
| Responder Country | uint 16 | Code for the country of the responding host. |
| Original Client Country | uint 16 | Code for the country of the host behind the proxy which originated the request. |
| IOC Number | uint16 | ID Number of the compromise associated with this event. |
| Source Autonomous System | uint32 | Autonomous system number of the source, either origin or peer. |
| Destination Autonomous System | uint32 | Autonomous system number of the destination, either origin or peer. |
| SNMP Input | uint16 | SNMP index of the input interface. |
| SNMP Output | uint16 | SNMP index of the output interface. |
| Source TOS | uint8 | Type of Service byte setting for the incoming interface. |
| Destination TOS | uint8 | Type of Service byte setting for the outgoing interface. |
| Source Mask | uint8 | Source address prefix mask. |
| Destination Mask | uint8 | Destination address prefix mask. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |
| VLAN ID | uint16 | VLAN identification number that indicates which VLAN the host is a member of. |
| String Block Type | uint32 | Initiates a String data block containing the Referenced Host. This value is always `0`. |

*Table B-45 Connection Statistics Data Block 6.2-6.7.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the Referenced Host String data block, including eight bytes for the block type and header fields plus the number of bytes in the Referenced Host field. |
| Referenced Host | string | Host name information provided in HTTP or DNS. |
| String Block Type | uint32 | Initiates a String data block containing the User Agent. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the User Agent String data block, including eight bytes for the block type and header fields plus the number of bytes in the User Agent field. |
| User Agent | string | Information from the UserAgent header field in the session. |
| String Block Type | uint32 | Initiates a String data block containing the HTTP Referrer. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the HTTP Referrer String data block, including eight bytes for the block type and header fields plus the number of bytes in the HTTP Referrer field. |
| HTTP Referrer | string | The site from which a page originated. This is found int he Referred header information in HTTP traffic. |
| SSL Certificate Fingerprint | uint8[20] | SHA1 hash of the SSL Server certificate. |
| SSL Policy ID | uint8[16] | ID number of the SSL policy that handled the connection. |
| SSL Rule ID | uint32 | ID number of the SSL rule or default action that handled the connection. |
| SSL Cipher Suite | uint16 | Encryption suite used by the SSL connection. The value is stored in decimal format. See `www.iana.org/assignments/tls-parameters/tls-parameters.xhtml` for the cipher suite designated by the value. |
| SSL Version | uint8 | The SSL or TLS protocol version used to encrypt the connection. |
| SSL Server Certificate Status | uint32 | The status of the SSL certificate. Possible values include:<br>• `0` — Not checked — The server certificate status was not evaluated.<br>• `1` — Unknown — The server certificate status could not be determined.<br>• `2` — Valid — The server certificate is valid.<br>• `4` — Self-signed — The server certificate is self-signed.<br>• `16` — Invalid Issuer — The server certificate has an invalid issuer.<br>• `32` — Invalid Signature — The server certificate has an invalid signature.<br>• `64` — Expired — The server certificate is expired.<br>• `128` — Not valid yet — The server certificate is not yet valid.<br>• `256` — Revoked — The server certificate has been revoked. |

*Table B-45 Connection Statistics Data Block 6.2-6.7.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Actual Action | uint16 | The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |
| SSL Expected Action | uint16 | The action which should be performed on the connection based on the SSL Rule. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |

*Table B-45 Connection Statistics Data Block 6.2-6.7.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Status | uint16 | Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'No Match'<br>• 2 — 'Success'<br>• 3 — 'Uncached Session'<br>• 4 — 'Unknown Cipher Suite'<br>• 5 — 'Unsupported Cipher Suite'<br>• 6 — 'Unsupported SSL Version'<br>• 7 — 'SSL Compression Used'<br>• 8 — 'Session Undecryptable in Passive Mode'<br>• 9 — 'Handshake Error'<br>• 10 — 'Decryption Error'<br>• 11 — 'Pending Server Name Category Lookup'<br>• 12 — 'Pending Common Name Category Lookup'<br>• 13 — 'Internal Error'<br>• 14 — 'Network Parameters Unavailable'<br>• 15 — 'Invalid Server Certificate Handle'<br>• 16 — 'Server Certificate Fingerprint Unavailable'<br>• 17 — 'Cannot Cache Subject DN'<br>• 18 — 'Cannot Cache Issuer DN'<br>• 19 — 'Unknown SSL Version'<br>• 20 — 'External Certificate List Unavailable'<br>• 21 — 'External Certificate Fingerprint Unavailable'<br>• 22 — 'Internal Certificate List Invalid'<br>• 23 — 'Internal Certificate List Unavailable'<br>• 24 — 'Internal Certificate Unavailable'<br>• 25 — 'Internal Certificate Fingerprint Unavailable'<br>• 26 — 'Server Certificate Validation Unavailable'<br>• 27 — 'Server Certificate Validation Failure'<br>• 28 — 'Invalid Action' |
| SSL Flow Error | uint32 | Detailed SSL error code. These values may be needed for support purposes. |

*Table B-45 Connection Statistics Data Block 6.2-6.7.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Messages | uint32 | The messages exchanged between client and server during the SSL handshake. See http://tools.ietf.org/html/rfc5246 for more information.<br>• `0x00000001` — NSE_MT__HELLO_REQUEST<br>• `0x00000002` — NSE_MT__CLIENT_ALERT<br>• `0x00000004` — NSE_MT__SERVER_ALERT<br>• `0x00000008` — NSE_MT__CLIENT_HELLO<br>• `0x00000010` — NSE_MT__SERVER_HELLO<br>• `0x00000020` — NSE_MT__SERVER_CERTIFICATE<br>• `0x00000040` — NSE_MT__SERVER_KEY_EXCHANGE<br>• `0x00000080` — NSE_MT__CERTIFICATE_REQUEST<br>• `0x00000100` — NSE_MT__SERVER_HELLO_DONE<br>• `0x00000200` — NSE_MT__CLIENT_CERTIFICATE<br>• `0x00000400` — NSE_MT__CLIENT_KEY_EXCHANGE<br>• `0x00000800` — NSE_MT__CERTIFICATE_VERIFY<br>• `0x00001000` — NSE_MT__CLIENT_CHANGE_CIPHER_SPEC<br>• `0x00002000` — NSE_MT__CLIENT_FINISHED<br>• `0x00004000` — NSE_MT__SERVER_CHANGE_CIPHER_SPEC<br>• `0x00008000` — NSE_MT__SERVER_FINISHED<br>• `0x00010000` — NSE_MT__NEW_SESSION_TICKET<br>• `0x00020000` — NSE_MT__HANDSHAKE_OTHER<br>• `0x00040000` — NSE_MT__APP_DATA_FROM_CLIENT<br>• `0x00080000` — NSE_MT__APP_DATA_FROM_SERVER |
| SSL Flow Flags | uint64 | The debugging level flags for an encrypted connection. Possible values include:<br>• `0x00000001` — NSE_FLOW__VALID - must be set for other fields to be valid<br>• `0x00000002` — NSE_FLOW__INITIALIZED - internal structures ready for processing<br>• `0x00000004` — NSE_FLOW__INTERCEPT - SSL session has been intercepted |
| String Block Type | uint32 | Initiates a String data block containing the SSL Server Name. This value is always `0`. |

*Table B-45 Connection Statistics Data Block 6.2-6.7.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the SSL Server Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the SSL Server Name field. |
| SSL Server Name | string | Name provided in the server name indication in the SSL Client Hello. |
| SSL URL Category | uint32 | Category of the flow as identified from the server name and certificate common name. |
| SSL Session ID | uint8[32] | Value of the session ID used during the SSL handshake when the client and server agree to do session reuse |
| SSL Session ID Length | uint8 | Length of the SSL Session ID. While the session ID cannot exceed 32 bytes, it may be less than 32 bytes. |
| SSL Ticket ID | uint8[20] | Hash of the session ticket used when the client and server agree to use a session ticket. |
| SSL Ticket ID Length | uint8 | Length of the SSL Ticket ID. While the ticket ID cannot exceed 20 bytes, it may be less than 20 bytes. |
| Network Analysis Policy revision | uint8[16] | Revision of the Network Analysis Policy associated with the connection event. |
| Endpoint Profile ID | uint32 | ID number of the type of device used by the connection endpoint as identified by ISE. This is unique for each DC and resolved in metadata. |
| Security Group ID | uint32 | ID number assigned to the user by ISE based on policy. |
| Location IPv6 | uint8[16] | IP address of the interface communicating with ISE. Can be IPv4 or IPv6. |
| HTTP Response | uint32 | Response code of the HTTP Request. |
| String Block Type | uint32 | Initiates a String data block for the DNS query. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the DNS query string. |
| DNS Query | string | The content of the query sent to the DNS server. |
| DNS Record Type | uint16 | The numerical value for the type of DNS record. |
| DNS Response Type | uint16 | The numerical value for the type of DNS response. |
| DNS TTL | uint32 | The time to live for the DNS response, in seconds. |
| Sinkhole UUID | uin8[16] | Revision UUID associated with this sinkhole object. |
| Security Intelligence List 1 | uint32 | Security Intelligence List associated with the event. This maps to a Security Intelligence list in associated metadata. There may be three Security Intelligence lists associated with the connection. |

*Table B-45 Connection Statistics Data Block 6.2-6.7.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Security Intelligence List 2 | uint32 | Security Intelligence List associated with the event. This maps to a Security Intelligence list in associated metadata. There may be three Security Intelligence lists associated with the connection. |
| Security Intelligence List 3 | uint32 | Security Intelligence List associated with the event. This maps to a Security Intelligence list in associated metadata. There may be three Security Intelligence lists associated with the connection. |

# Connection Statistics Data Block 7.0

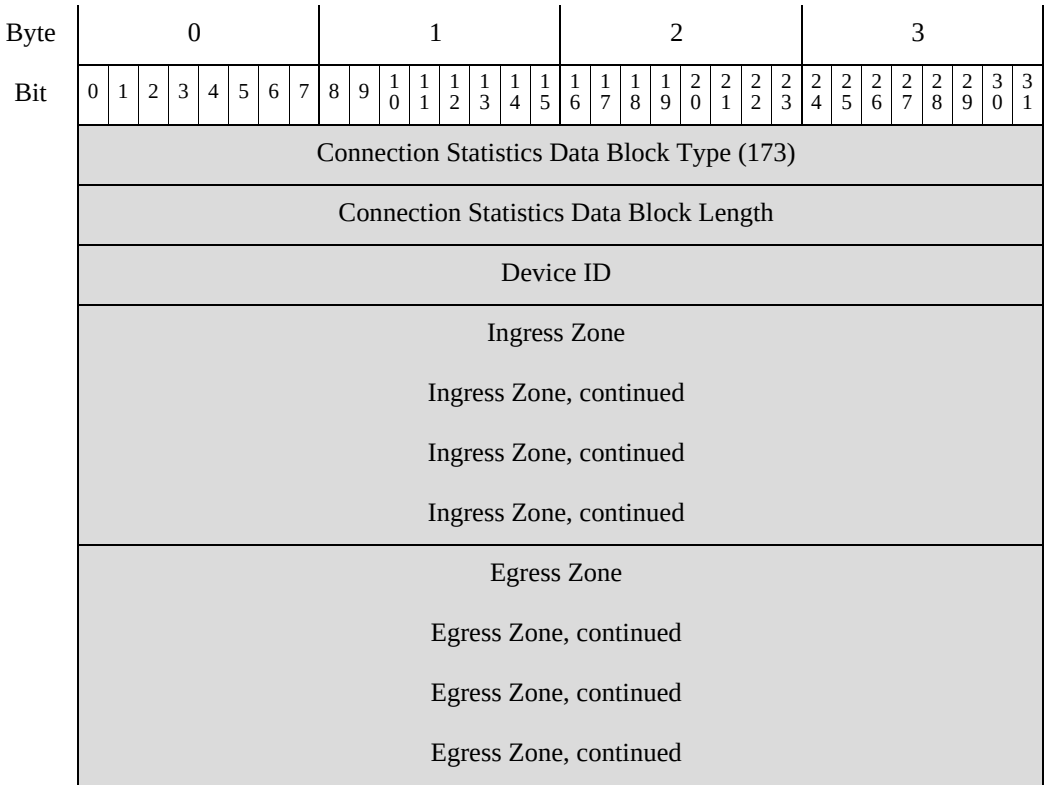
The connection statistics data block is used in connection data messages. Security Group Tag, virtual routing and forwarding, and dynamic attribute fields have been added to Connection Statistics Data Block for 7.0+. The connection statistics data block for version 7.0+ has a block type of 173 in the series 1 group of blocks. It supersedes block type 168, Connection Statistics Data Block 6.2-6.7.x, page B-256. It is superseded by block type 174

You request connection event records by setting the extended event flag—bit 30 in the Request Flags field—in the request message with an event version of 16 and an event code of 71. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.

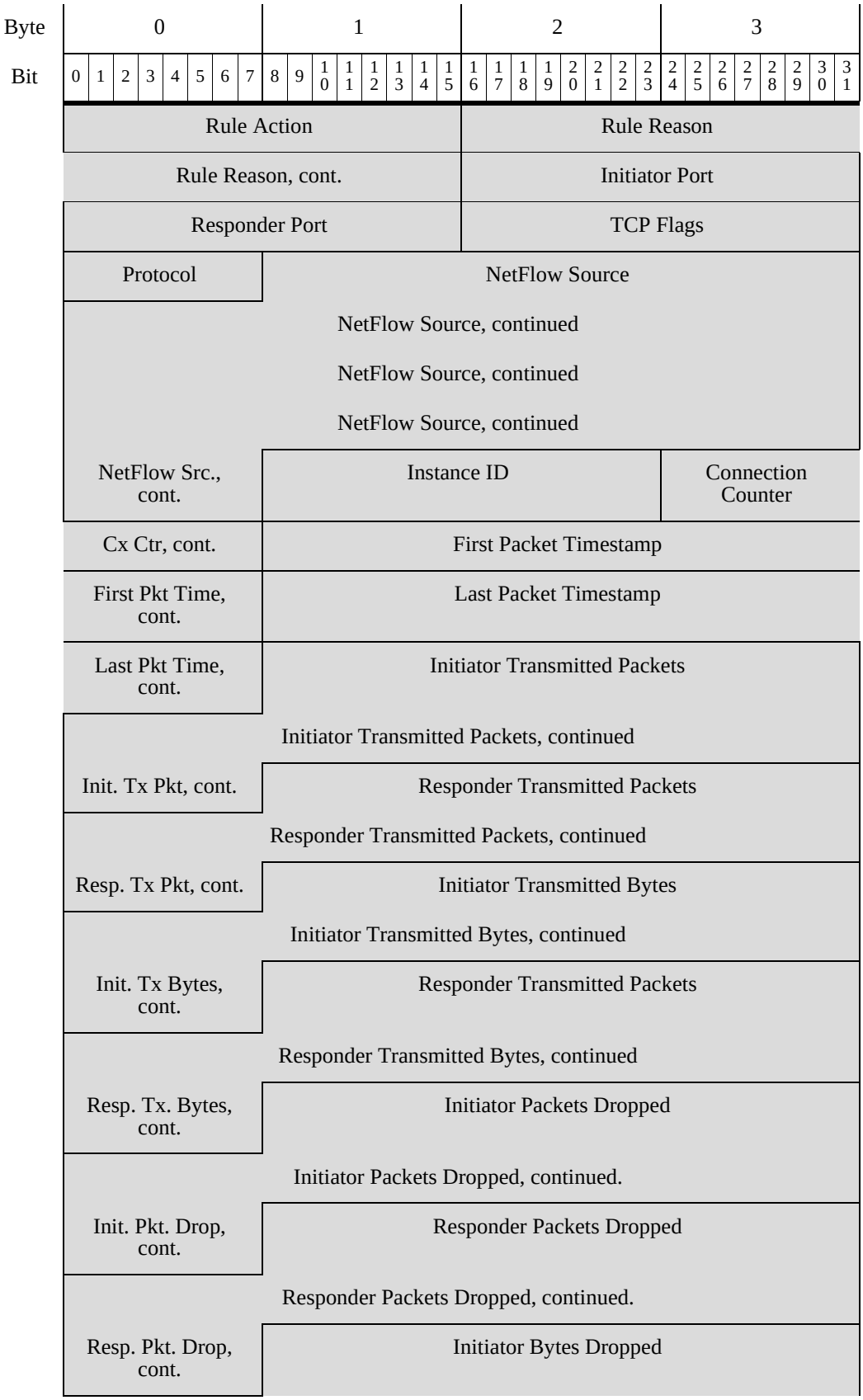
For more information on the Connection Statistics Data message, see Connection Statistics Data Message, page 4-53.

The following diagram shows the format of a Connection Statistics data block for 7.0:

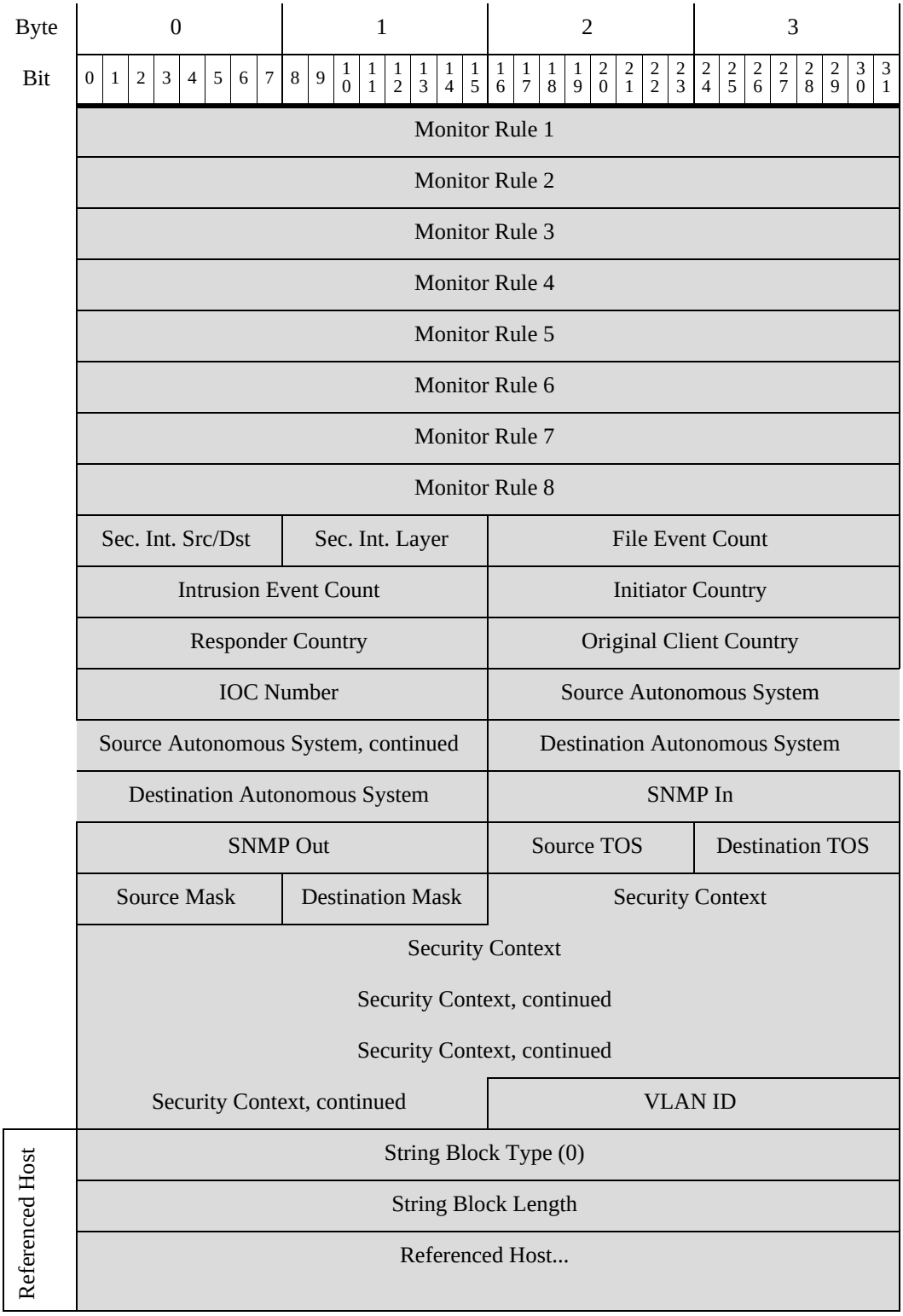
7

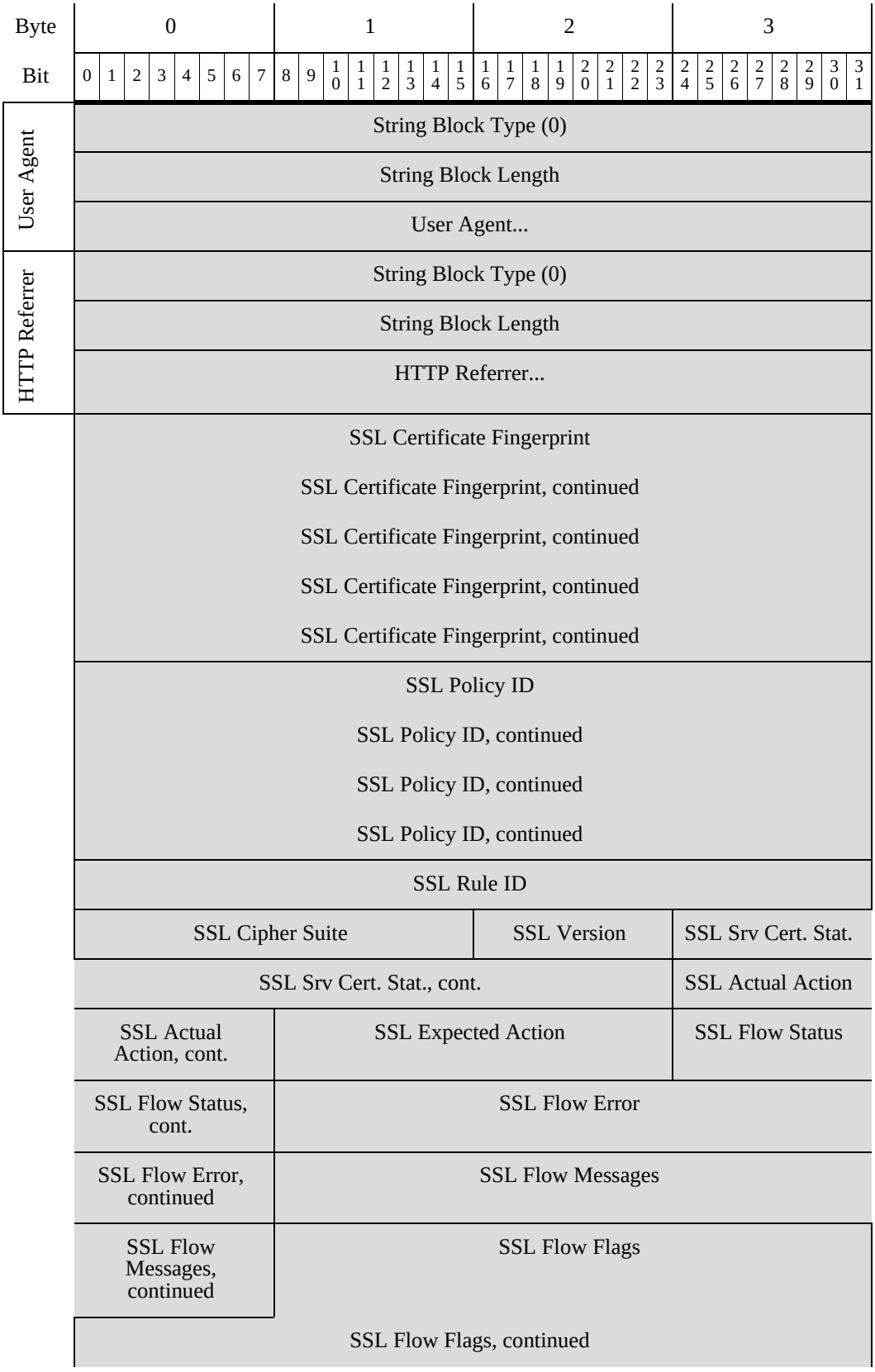


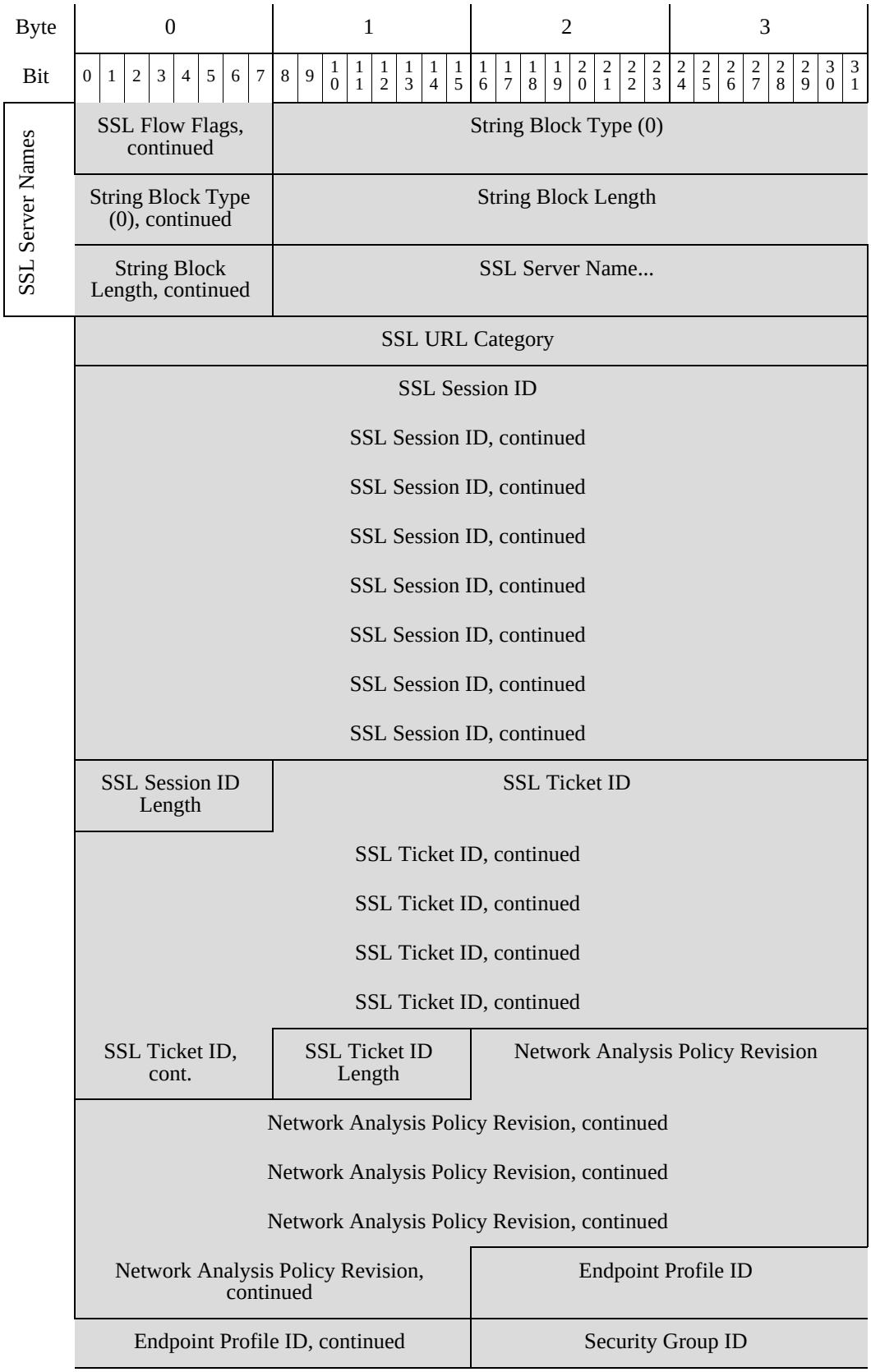
| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Ingress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ingress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Egress Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initiator IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Responder IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Original Client IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Original Client IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Original Client IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Original Client IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy Revision, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Tunnel Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

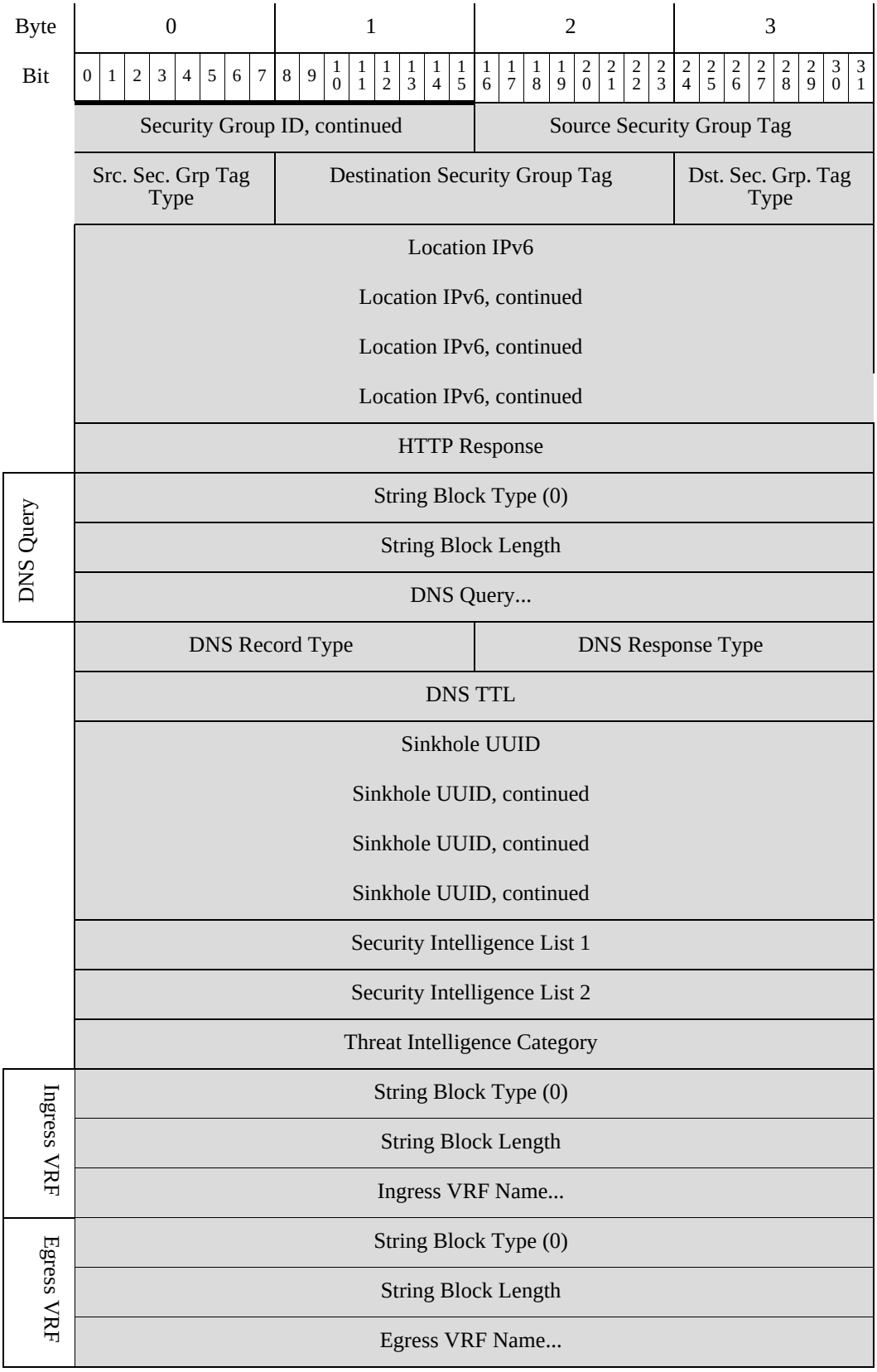

Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Rule Action
Rule Reason
Rule Reason, cont.
Initiator Port
Responder Port
TCP Flags
Protocol
NetFlow Source
NetFlow Source, continued
NetFlow Source, continued
NetFlow Source, continued
NetFlow Src., cont.
Instance ID
Connection Counter
Cx Ctr, cont.
First Packet Timestamp
First Pkt Time, cont.
Last Packet Timestamp
Last Pkt Time, cont.
Initiator Transmitted Packets
Initiator Transmitted Packets, continued
Init. Tx Pkt, cont.
Responder Transmitted Packets
Responder Transmitted Packets, continued
Resp. Tx Pkt, cont.
Initiator Transmitted Bytes
Initiator Transmitted Bytes, continued
Init. Tx Bytes, cont.
Responder Transmitted Packets
Responder Transmitted Bytes, continued
Resp. Tx. Bytes, cont.
Initiator Packets Dropped
Initiator Packets Dropped, continued.
Init. Pkt. Drop, cont.
Responder Packets Dropped
Responder Packets Dropped, continued.
Resp. Pkt. Drop, cont.
Initiator Bytes Dropped

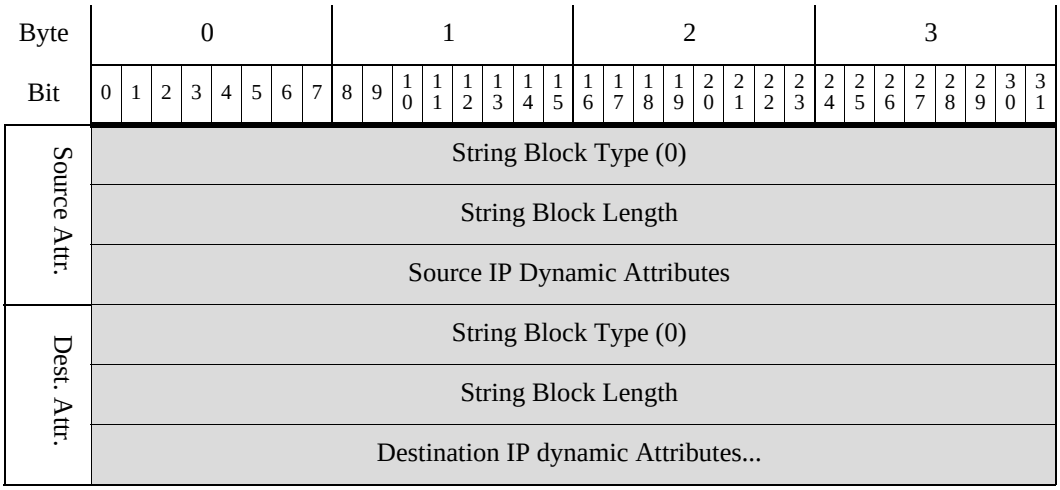
| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Initiator Bytes Dropped, continued. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Init. Byte Drop, cont. | | | | | | | | Responder Bytes Dropped | | | | | | | | | | | | | | | | | | | | | | | |
| | Responder Bytes Dropped, continued. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Rsp. Byte Drop, cont. | | | | | | | | QOS Applied Interface | | | | | | | | | | | | | | | | | | | | | | | |
| | QOS Applied Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | QOS Applied Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | QOS Applied Interface, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | QOS Intf., cont. | | | | | | | | QOS Rule ID | | | | | | | | | | | | | | | | | | | | | | | |
| | QOS Rule ID, cont. | | | | | | | | User ID | | | | | | | | | | | | | | | | | | | | | | | |
| | User ID, cont. | | | | | | | | Application Protocol ID | | | | | | | | | | | | | | | | | | | | | | | |
| | App Prot. ID, cont. | | | | | | | | URL Category | | | | | | | | | | | | | | | | | | | | | | | |
| | URL Category, cont. | | | | | | | | URL Reputation | | | | | | | | | | | | | | | | | | | | | | | |
| | URL Rep., cont. | | | | | | | | Client Application ID | | | | | | | | | | | | | | | | | | | | | | | |
| | Client App ID, cont. | | | | | | | | Web Application ID | | | | | | | | | | | | | | | | | | | | | | | |
| Client URL | Web App. ID, cont. | | | | | | | | Str. Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | |
| | Str. Block Type, cont. | | | | | | | | String Block Length | | | | | | | | | | | | | | | | | | | | | | | |
| | Str. Block Len., cont. | | | | | | | | Client App. URL... | | | | | | | | | | | | | | | | | | | | | | | |
| NetBIOS Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | NetBIOS Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client App Version | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Client Application Version... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Monitor Rule 1
Monitor Rule 2
Monitor Rule 3
Monitor Rule 4
Monitor Rule 5
Monitor Rule 6
Monitor Rule 7
Monitor Rule 8
Sec. Int. Src/Dst
Sec. Int. Layer
File Event Count
Intrusion Event Count
Initiator Country
Responder Country
Original Client Country
IOC Number
Source Autonomous System
Source Autonomous System, continued
Destination Autonomous System
Destination Autonomous System
SNMP In
SNMP Out
Source TOS
Destination TOS
Source Mask
Destination Mask
Security Context
Security Context
Security Context, continued
Security Context, continued
Security Context, continued
VLAN ID
Referenced Host
String Block Type (0)
String Block Length
Referenced Host...


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
User Agent
String Block Type (0)
String Block Length
User Agent...
HTTP Referrer
String Block Type (0)
String Block Length
HTTP Referrer...
SSL Certificate Fingerprint
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Policy ID
SSL Policy ID, continued
SSL Policy ID, continued
SSL Policy ID, continued
SSL Rule ID
SSL Cipher Suite
SSL Version
SSL Srv Cert. Stat.
SSL Srv Cert. Stat., cont.
SSL Actual Action
SSL Actual Action, cont.
SSL Expected Action
SSL Flow Status
SSL Flow Status, cont.
SSL Flow Error
SSL Flow Error, continued
SSL Flow Messages
SSL Flow Messages, continued
SSL Flow Flags
SSL Flow Flags, continued


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
SSL Server Names
SSL Flow Flags, continued
String Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
SSL Server Name...
SSL URL Category
SSL Session ID
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID, continued
SSL Session ID Length
SSL Ticket ID
SSL Ticket ID, continued
SSL Ticket ID, continued
SSL Ticket ID, continued
SSL Ticket ID, continued
SSL Ticket ID, cont.
SSL Ticket ID Length
Network Analysis Policy Revision
Network Analysis Policy Revision, continued
Network Analysis Policy Revision, continued
Network Analysis Policy Revision, continued
Network Analysis Policy Revision, continued
Endpoint Profile ID
Endpoint Profile ID, continued
Security Group ID


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Security Group ID, continued
Source Security Group Tag
Src. Sec. Grp Tag Type
Destination Security Group Tag
Dst. Sec. Grp. Tag Type
Location IPv6
Location IPv6, continued
Location IPv6, continued
Location IPv6, continued
HTTP Response
DNS Query
String Block Type (0)
String Block Length
DNS Query...
DNS Record Type
DNS Response Type
DNS TTL
Sinkhole UUID
Sinkhole UUID, continued
Sinkhole UUID, continued
Sinkhole UUID, continued
Security Intelligence List 1
Security Intelligence List 2
Threat Intelligence Category
Ingress VRF
String Block Type (0)
String Block Length
Ingress VRF Name...
Egress VRF
String Block Type (0)
String Block Length
Egress VRF Name...




The following table describes the fields of the Connection Statistics data block for 7.0.

***Table B-46 Connection Statistics Data Block 7.0 Fields***

| Field | Data Type | Description |
|---|---|---|
| Connection Statistics Data Block Type | uint32 | Initiates a Connection Statistics data block for 7.0+. The value is always `173`. |
| Connection Statistics Data Block Length | uint32 | Number of bytes in the Connection Statistics data block, including eight bytes for the connection statistics block type and length fields, plus the number of bytes in the connection data that follows. |
| Device ID | uint32 | The device that detected the connection event. |
| Ingress Zone | uint8[16] | Ingress security zone in the event that triggered the policy violation. |
| Egress Zone | uint8[16] | Egress security zone in the event that triggered the policy violation. |
| Ingress Interface | uint8[16] | Interface for the inbound traffic. |
| Egress Interface | uint8[16] | Interface for the outbound traffic. |
| Initiator IP Address | uint8[16] | IP address of the host that initiated the session described in the connection event, in IP address octets. |
| Responder IP Address | uint8[16] | IP address of the host that responded to the initiating host, in IP address octets. |
| Original Client IP Address | uint8[16] | IP address of the host behind the proxy that originated the request, in IP address octets. |
| Policy Revision | uint8[16] | Revision number of the rule associated with the triggered correlation event, if applicable. |
| Rule ID | uint32 | Internal identifier for the rule that triggered the event, if applicable. |

*Table B-46 Connection Statistics Data Block 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Tunnel Rule ID | uint32 | Internal identifier for the tunnel rule that triggered the event, if applicable. |
| Rule Action | uint16 | The action selected in the user interface for that rule (allow, block, and so forth). |
| Rule Reason | uint32 | The reason the rule triggered the event. |
| Initiator Port | uint16 | Port used by the initiating host. |
| Responder Port | uint16 | Port used by the responding host. |
| TCP Flags | uint16 | Indicates any TCP flags for the connection event. |
| Protocol | uint8 | The IANA-specified protocol number. |
| NetFlow Source | uint8[16] | IP address of the NetFlow-enabled device that exported the data for the connection. |
| Instance ID | uint16 | Numerical ID of the Snort instance on the managed device that generated the event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| First Packet Timestamp | uint32 | UNIX timestamp of the date and time the first packet was exchanged in the session. |
| Last Packet Timestamp | uint32 | UNIX timestamp of the date and time the last packet was exchanged in the session. |
| Initiator Transmitted Packets | uint64 | Number of packets transmitted by the initiating host. |
| Responder Transmitted Packets | uint64 | Number of packets transmitted by the responding host. |
| Initiator Transmitted Bytes | uint64 | Number of bytes transmitted by the initiating host. |
| Responder Transmitted Bytes | uint64 | Number of bytes transmitted by the responding host. |
| Initiator Packets Dropped | uint64 | Number of packets dropped from the session initiator due to rate limiting. |
| Responder Packets Dropped | uint64 | Number of packets dropped from the session responder due to rate limiting. |
| Initiator Bytes Dropped | uint64 | Number of bytes dropped from the session initiator due to rate limiting. |
| Responder Bytes Dropped | uint64 | Number of bytes dropped from the session responders due to rate limiting. |
| QOS Applied Interface | uint8[16] | For rate-limited connections, the name of the interface on which rate limiting is applied. |
| QOS Rule ID | uint32 | Internal ID number of the Quality of Service rule applied to the connection, if applicable. |

*Table B-46 Connection Statistics Data Block 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| User ID | uint32 | Internal identification number for the user who last logged into the host that generated the traffic. |
| Application Protocol ID | uint32 | Application ID of the application protocol. |
| URL Category | uint32 | The internal identification number of the URL category. |
| URL Reputation | uint32 | The internal identification number for the URL reputation. |
| Client Application ID | uint32 | The internal identification number of the detected client application, if applicable. |
| Web Application ID | uint32 | The internal identification number of the detected web application, if applicable. |
| String Block Type | uint32 | Initiates a String data block for the client application URL. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the client application URL String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the client application URL string. |
| Client Application URL | string | URL the client application accessed, if applicable (`/files/index.html`, for example). |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for the client application version. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block for the client application version, including eight bytes for the string block type and length, plus the number of bytes in the version. |
| Client Application Version | string | Client application version. |
| Monitor Rule 1 | uint32 | The ID of the first monitor rule associated with the connection event. |
| Monitor Rule 2 | uint32 | The ID of the second monitor rule associated with the connection event. |
| Monitor Rule 3 | uint32 | The ID of the third monitor rule associated with the connection event. |
| Monitor Rule 4 | uint32 | The ID of the fourth monitor rule associated with the connection event. |
| Monitor Rule 5 | uint32 | The ID of the fifth monitor rule associated with the connection event. |

*Table B-46 Connection Statistics Data Block 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Monitor Rule 6 | uint32 | The ID of the sixth monitor rule associated with the connection event. |
| Monitor Rule 7 | uint32 | The ID of the seventh monitor rule associated with the connection event. |
| Monitor Rule 8 | uint32 | The ID of the eighth monitor rule associated with the connection event. |
| Security Intelligence Source/ Destination | uint8 | Whether the source or destination IP address matched the IP block list. |
| Security Intelligence Layer | uint8 | The IP layer that matched the IP block list. |
| File Event Count | uint16 | Value used to distinguish between file events that happen during the same second. |
| Intrusion Event Count | uint16 | Value used to distinguish between intrusion events that happen during the same second. |
| Initiator Country | uint16 | Code for the country of the initiating host. |
| Responder Country | uint 16 | Code for the country of the responding host. |
| Original Client Country | uint 16 | Code for the country of the host behind the proxy which originated the request. |
| IOC Number | uint16 | ID Number of the compromise associated with this event. |
| Source Autonomous System | uint32 | Autonomous system number of the source, either origin or peer. |
| Destination Autonomous System | uint32 | Autonomous system number of the destination, either origin or peer. |
| SNMP Input | uint16 | SNMP index of the input interface. |
| SNMP Output | uint16 | SNMP index of the output interface. |
| Source TOS | uint8 | Type of Service byte setting for the incoming interface. |
| Destination TOS | uint8 | Type of Service byte setting for the outgoing interface. |
| Source Mask | uint8 | Source address prefix mask. |
| Destination Mask | uint8 | Destination address prefix mask. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |
| VLAN ID | uint16 | VLAN identification number that indicates which VLAN the host is a member of. |
| String Block Type | uint32 | Initiates a String data block containing the Referenced Host. This value is always `0`. |

*Table B-46 Connection Statistics Data Block 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the Referenced Host String data block, including eight bytes for the block type and header fields plus the number of bytes in the Referenced Host field. |
| Referenced Host | string | Host name information provided in HTTP or DNS. |
| String Block Type | uint32 | Initiates a String data block containing the User Agent. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the User Agent String data block, including eight bytes for the block type and header fields plus the number of bytes in the User Agent field. |
| User Agent | string | Information from the UserAgent header field in the session. |
| String Block Type | uint32 | Initiates a String data block containing the HTTP Referrer. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the HTTP Referrer String data block, including eight bytes for the block type and header fields plus the number of bytes in the HTTP Referrer field. |
| HTTP Referrer | string | The site from which a page originated. This is found int he Referred header information in HTTP traffic. |
| SSL Certificate Fingerprint | uint8[20] | SHA1 hash of the SSL Server certificate. |
| SSL Policy ID | uint8[16] | ID number of the SSL policy that handled the connection. |
| SSL Rule ID | uint32 | ID number of the SSL rule or default action that handled the connection. |
| SSL Cipher Suite | uint16 | Encryption suite used by the SSL connection. The value is stored in decimal format. See `www.iana.org/assignments/tls-parameters/tls-parameters.xhtml` for the cipher suite designated by the value. |
| SSL Version | uint8 | The SSL or TLS protocol version used to encrypt the connection. |
| SSL Server Certificate Status | uint32 | The status of the SSL certificate. Possible values include:<br>• `0` — Not checked — The server certificate status was not evaluated.<br>• `1` — Unknown — The server certificate status could not be determined.<br>• `2` — Valid — The server certificate is valid.<br>• `4` — Self-signed — The server certificate is self-signed.<br>• `16` — Invalid Issuer — The server certificate has an invalid issuer.<br>• `32` — Invalid Signature — The server certificate has an invalid signature.<br>• `64` — Expired — The server certificate is expired.<br>• `128` — Not valid yet — The server certificate is not yet valid.<br>• `256` — Revoked — The server certificate has been revoked. |

*Table B-46 Connection Statistics Data Block 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Actual Action | uint16 | The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |
| SSL Expected Action | uint16 | The action which should be performed on the connection based on the SSL Rule. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |

*Table B-46 Connection Statistics Data Block 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Status | uint16 | Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'No Match'<br>• 2 — 'Success'<br>• 3 — 'Uncached Session'<br>• 4 — 'Unknown Cipher Suite'<br>• 5 — 'Unsupported Cipher Suite'<br>• 6 — 'Unsupported SSL Version'<br>• 7 — 'SSL Compression Used'<br>• 8 — 'Session Undecryptable in Passive Mode'<br>• 9 — 'Handshake Error'<br>• 10 — 'Decryption Error'<br>• 11 — 'Pending Server Name Category Lookup'<br>• 12 — 'Pending Common Name Category Lookup'<br>• 13 — 'Internal Error'<br>• 14 — 'Network Parameters Unavailable'<br>• 15 — 'Invalid Server Certificate Handle'<br>• 16 — 'Server Certificate Fingerprint Unavailable'<br>• 17 — 'Cannot Cache Subject DN'<br>• 18 — 'Cannot Cache Issuer DN'<br>• 19 — 'Unknown SSL Version'<br>• 20 — 'External Certificate List Unavailable'<br>• 21 — 'External Certificate Fingerprint Unavailable'<br>• 22 — 'Internal Certificate List Invalid'<br>• 23 — 'Internal Certificate List Unavailable'<br>• 24 — 'Internal Certificate Unavailable'<br>• 25 — 'Internal Certificate Fingerprint Unavailable'<br>• 26 — 'Server Certificate Validation Unavailable'<br>• 27 — 'Server Certificate Validation Failure'<br>• 28 — 'Invalid Action' |
| SSL Flow Error | uint32 | Detailed SSL error code. These values may be needed for support purposes. |

*Table B-46 Connection Statistics Data Block 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Messages | uint32 | The messages exchanged between client and server during the SSL handshake. See http://tools.ietf.org/html/rfc5246 for more information.<br>• `0x00000001` — NSE_MT__HELLO_REQUEST<br>• `0x00000002` — NSE_MT__CLIENT_ALERT<br>• `0x00000004` — NSE_MT__SERVER_ALERT<br>• `0x00000008` — NSE_MT__CLIENT_HELLO<br>• `0x00000010` — NSE_MT__SERVER_HELLO<br>• `0x00000020` — NSE_MT__SERVER_CERTIFICATE<br>• `0x00000040` — NSE_MT__SERVER_KEY_EXCHANGE<br>• `0x00000080` — NSE_MT__CERTIFICATE_REQUEST<br>• `0x00000100` — NSE_MT__SERVER_HELLO_DONE<br>• `0x00000200` — NSE_MT__CLIENT_CERTIFICATE<br>• `0x00000400` — NSE_MT__CLIENT_KEY_EXCHANGE<br>• `0x00000800` — NSE_MT__CERTIFICATE_VERIFY<br>• `0x00001000` — NSE_MT__CLIENT_CHANGE_CIPHER_SPEC<br>• `0x00002000` — NSE_MT__CLIENT_FINISHED<br>• `0x00004000` — NSE_MT__SERVER_CHANGE_CIPHER_SPEC<br>• `0x00008000` — NSE_MT__SERVER_FINISHED<br>• `0x00010000` — NSE_MT__NEW_SESSION_TICKET<br>• `0x00020000` — NSE_MT__HANDSHAKE_OTHER<br>• `0x00040000` — NSE_MT__APP_DATA_FROM_CLIENT<br>• `0x00080000` — NSE_MT__APP_DATA_FROM_SERVER |
| SSL Flow Flags | uint64 | The debugging level flags for an encrypted connection. Possible values include:<br>• `0x00000001` — NSE_FLOW__VALID - must be set for other fields to be valid<br>• `0x00000002` — NSE_FLOW__INITIALIZED - internal structures ready for processing<br>• `0x00000004` — NSE_FLOW__INTERCEPT - SSL session has been intercepted |
| String Block Type | uint32 | Initiates a String data block containing the SSL Server Name. This value is always `0`. |

*Table B-46 Connection Statistics Data Block 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the SSL Server Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the SSL Server Name field. |
| SSL Server Name | string | Name provided in the server name indication in the SSL Client Hello. |
| SSL URL Category | uint32 | Category of the flow as identified from the server name and certificate common name. |
| SSL Session ID | uint8[32] | Value of the session ID used during the SSL handshake when the client and server agree to do session reuse |
| SSL Session ID Length | uint8 | Length of the SSL Session ID. While the session ID cannot exceed 32 bytes, it may be less than 32 bytes. |
| SSL Ticket ID | uint8[20] | Hash of the session ticket used when the client and server agree to use a session ticket. |
| SSL Ticket ID Length | uint8 | Length of the SSL Ticket ID. While the ticket ID cannot exceed 20 bytes, it may be less than 20 bytes. |
| Network Analysis Policy revision | uint8[16] | Revision of the Network Analysis Policy associated with the connection event. |
| Endpoint Profile ID | uint32 | ID number of the type of device used by the connection endpoint as identified by ISE. This is unique for each DC and resolved in metadata. |
| Security Group ID | uint32 | ID number assigned to the user by ISE based on policy. |
| Source Security Group Tag | uint16 | The Security Group Tag of the source of the connection. |
| Source Security Group Tag Type | uint8 | How the Source Security Group Tag was assigned:<br>• 0 — Unknown<br>• 1 — Inline<br>• 2 — Session Directory<br>• 3 — Security Group Tag Exchange Protocol (SXP) |
| Destination Security Group Tag | uint16 | The Security Group Tag of the destination of the connection. |
| Destination Security Group Tag Type | uint8 | How the Destination Security Group Tag was assigned:<br>• 0 — Unknown<br>• 1 — Inline<br>• 2 — Session Directory<br>• 3 — Security Group Tag Exchange Protocol (SXP) |
| Location IPv6 | uint8[16] | IP address of the interface communicating with ISE. Can be IPv4 or IPv6. |
| HTTP Response | uint32 | Response code of the HTTP Request. |

*Table B-46 Connection Statistics Data Block 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Type | uint32 | Initiates a String data block for the DNS query. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the DNS query string. |
| DNS Query | string | The content of the query sent to the DNS server. |
| DNS Record Type | uint16 | The numerical value for the type of DNS record. |
| DNS Response Type | uint16 | The numerical value for the type of DNS response. |
| DNS TTL | uint32 | The time to live for the DNS response, in seconds. |
| Sinkhole UUID | uin8[16] | Revision UUID associated with this sinkhole object. |
| Security Intelligence List 1 | uint32 | Security Intelligence List associated with the event. This maps to a Security Intelligence list in associated metadata. There may be three Security Intelligence lists associated with the connection. |
| Security Intelligence List 2 | uint32 | Security Intelligence List associated with the event. This maps to a Security Intelligence list in associated metadata. There may be three Security Intelligence lists associated with the connection. |
| Threat Intelligence Category | uint32 | Threat Intelligence Category associated with the event. This maps to a Threat Intelligence list in associated metadata. |
| String Block Type | uint32 | Initiates a String data block containing the name of the ingress VRF. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Ingress VRF name field. |
| Ingress VRF Name | string | The virtual router through which traffic entered the network. |
| String Block Type | uint32 | Initiates a String data block containing the name of the egress VRF. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Egress VRF name field. |
| Egress VRF Name | string | The name of the virtual router through which traffic exited the network. |
| String Block Type | uint32 | Initiates a String data block containing the name of the Source IP Dynamic Attribute. This value is always 0. |
| String Block Length | uint32 | The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Source IP Dynamic Attribute field. |

*Table B-46 Connection Statistics Data Block 7.0 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Source IP Dynamic Attribute | string | Dynamic Attributes associated with the source IP address. |
| String Block Type | uint32 | Initiates a String data block containing the name of the Destination IP Dynamic Attribute. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Destination IP Dynamic Attribute field. |
| Destination IP Dynamic Attribute | string | Dynamic Attributes associated with the destination IP address. |

# Legacy File Event Data Structures

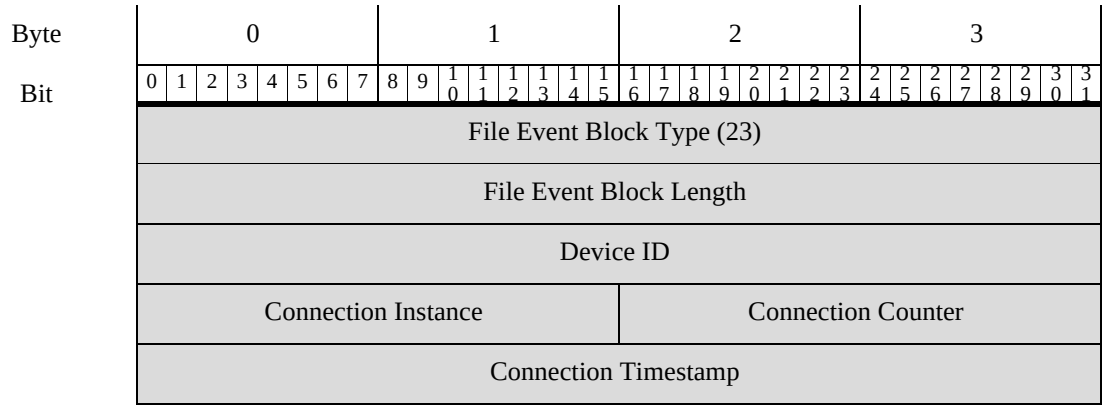
The following topics describe other legacy file event data structures:

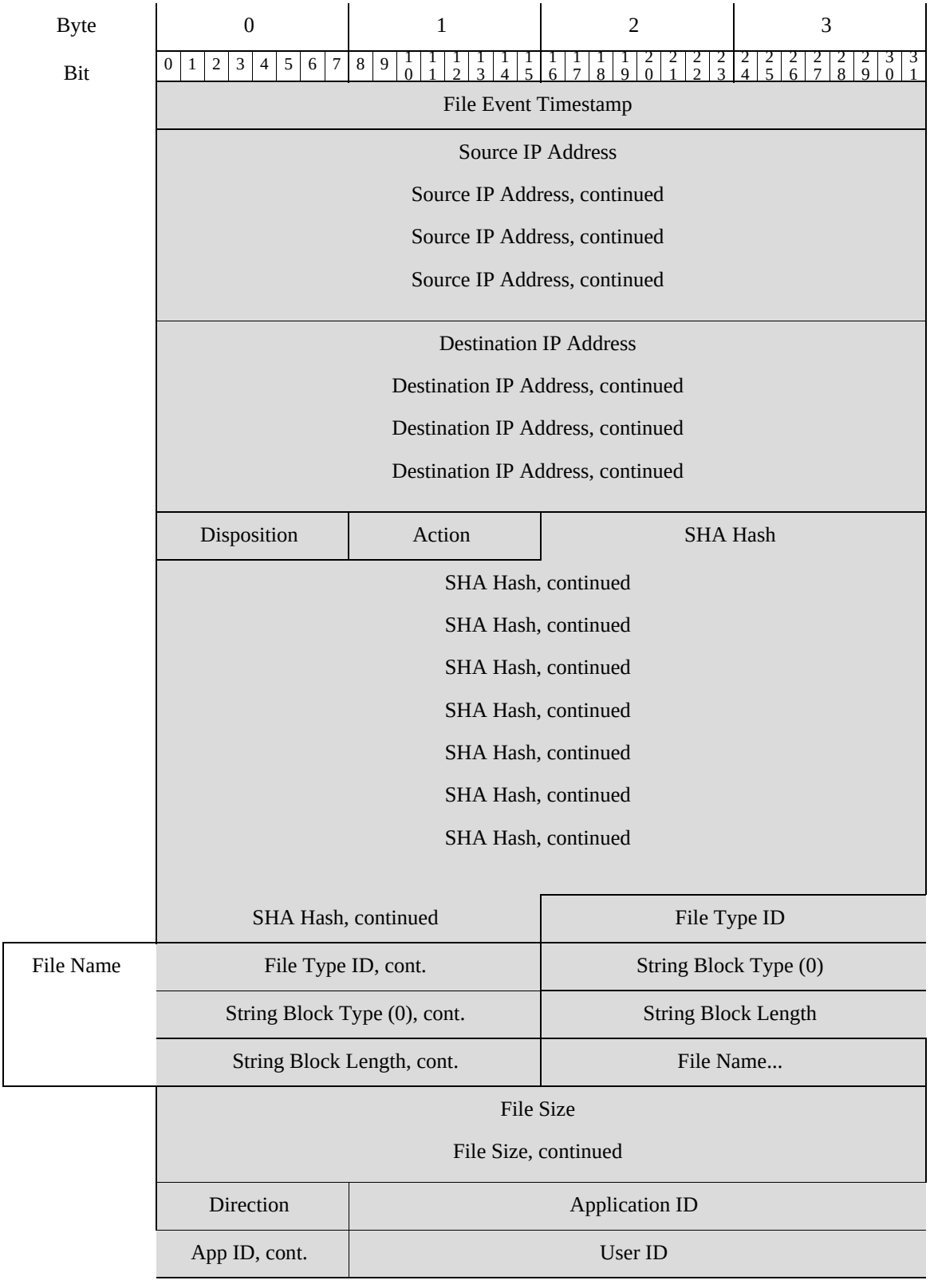

- File Event for 5.1.1.x, page B-290
- File Event for 5.2.x, page B-294
- File Event for 5.3, page B-298
- File Event for 5.3.1, page B-304
- File Event for 5.4.x, page B-310
- File Event SHA Hash for 5.1.1-5.2.x, page B-330

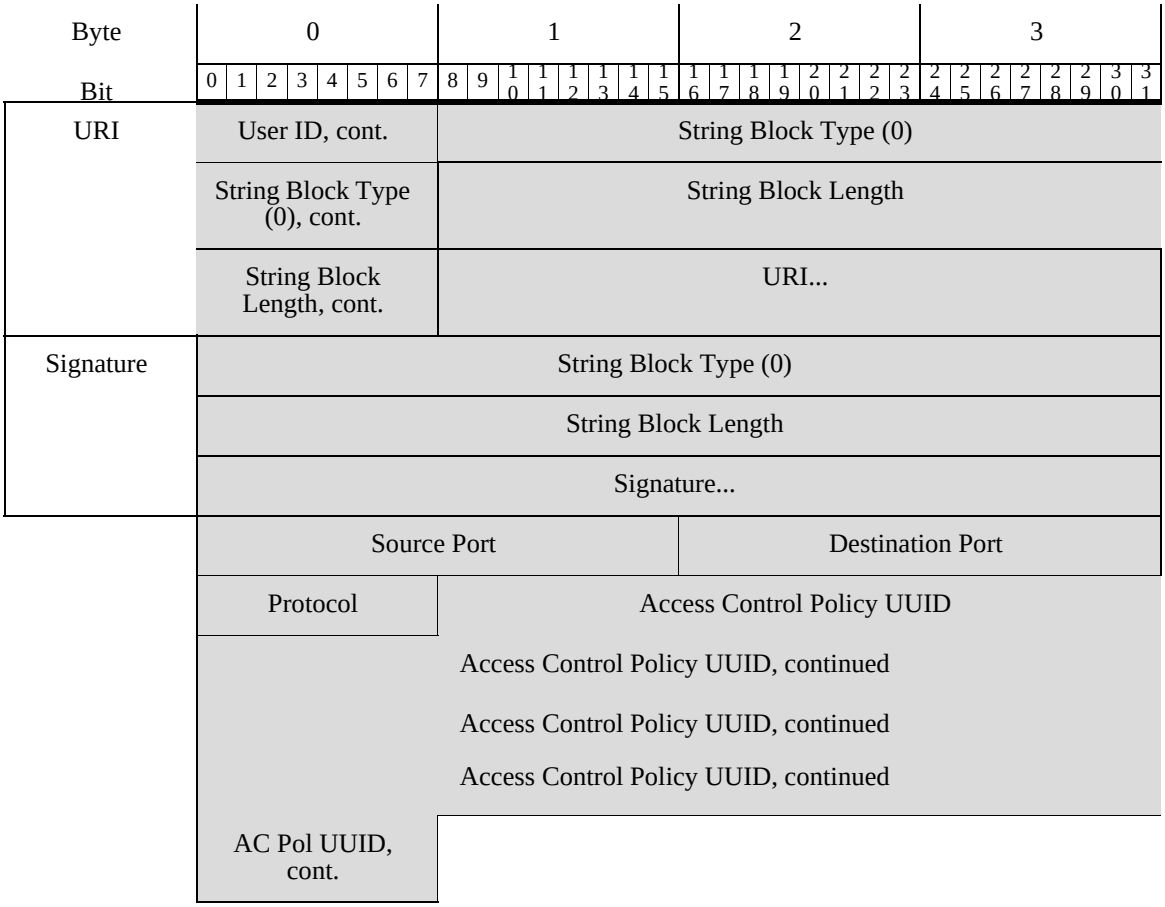

## File Event for 5.1.1.x

The file event contains information on files that are sent over the network. This includes the connection information, whether the file is malware, and specific information to identify the file. The file event has a block type of 23 in the series 2 group of blocks.

The following graphic shows the structure of the File Event data block:




Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
File Event Timestamp
Source IP Address
Source IP Address, continued
Source IP Address, continued
Source IP Address, continued
Destination IP Address
Destination IP Address, continued
Destination IP Address, continued
Destination IP Address, continued
Disposition
Action
SHA Hash
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
File Type ID
File Name
File Type ID, cont.
String Block Type (0)
String Block Type (0), cont.
String Block Length
String Block Length, cont.
File Name...
File Size
File Size, continued
Direction
Application ID
App ID, cont.
User ID




The following table describes the fields in the file event data block:

***Table B-47 File Event Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| File Event Block Type | uint32 | Initiates whether file event data block. This value is always `23`. |
| File Event Block Length | uint32 | Total number of bytes in the file event block, including eight bytes for the file event block type and length fields, plus the number of bytes of data that follows. |
| Device ID | uint32 | ID for the device that generated the event. |
| Connection Instance | uint16 | Snort instance on the device that generated the event. Used to link the event with a connection or intrusion event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Connection Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the associated connection event. |
| File Event Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of when the file type is identified and the file event generated. |

*Table B-47 File Event Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Source IP Address | uint8[16] | IPv4 or IPv6 address for the source of the connection. |
| Destination IP Address | uint8[16] | IPv4 or IPv6 address for the destination of the connection. |
| Disposition | uint8 | The malware status of the file. Possible values include:<br>• 1 — CLEAN — The file is clean and does not contain malware.<br>• 2 — UNKNOWN — It is unknown whether the file contains malware.<br>• 3 — MALWARE — The file contains malware.<br>• 4 — CACHE_MISS — The software was unable to send a request to the Cisco cloud for a disposition.<br>• 5 — NO_CLOUD_RESP — The Cisco cloud services did not respond to the request. |
| Action | uint8 | The action taken on the file based on the file type. Can have the following values:<br>• 1 — Detect<br>• 2 — Block<br>• 3 — Malware Cloud Lookup<br>• 4 — Malware Block<br>• 5 — Malware Allow List |
| SHA Hash | uint8[32] | SHA-256 hash of the file, in binary format. |
| File Type ID | uint32 | ID number that maps to the file type. |
| File Name | string | Name of the file. |
| File Size | uint64 | Size of the file in bytes. |
| Direction | uint8 | Value that indicates whether the file was uploaded or downloaded. Can have the following values:<br>• 1 — Download<br>• 2 — Upload<br>Currently the value depends on the protocol (for example, if the connection is HTTP it is a download). |
| Application ID | uint32 | ID number that maps to the application using the file transfer. |
| User ID | uint32 | ID number for the user logged into the destination host, as identified by the system. |
| URI | string | Uniform Resource Identifier (URI) of the connection. |
| Signature | string | SHA-256 hash of the file, in string format. |
| Source Port | uint16 | Port number for the source of the connection. |
| Destination Port | uint16 | Port number for the destination of the connection. |

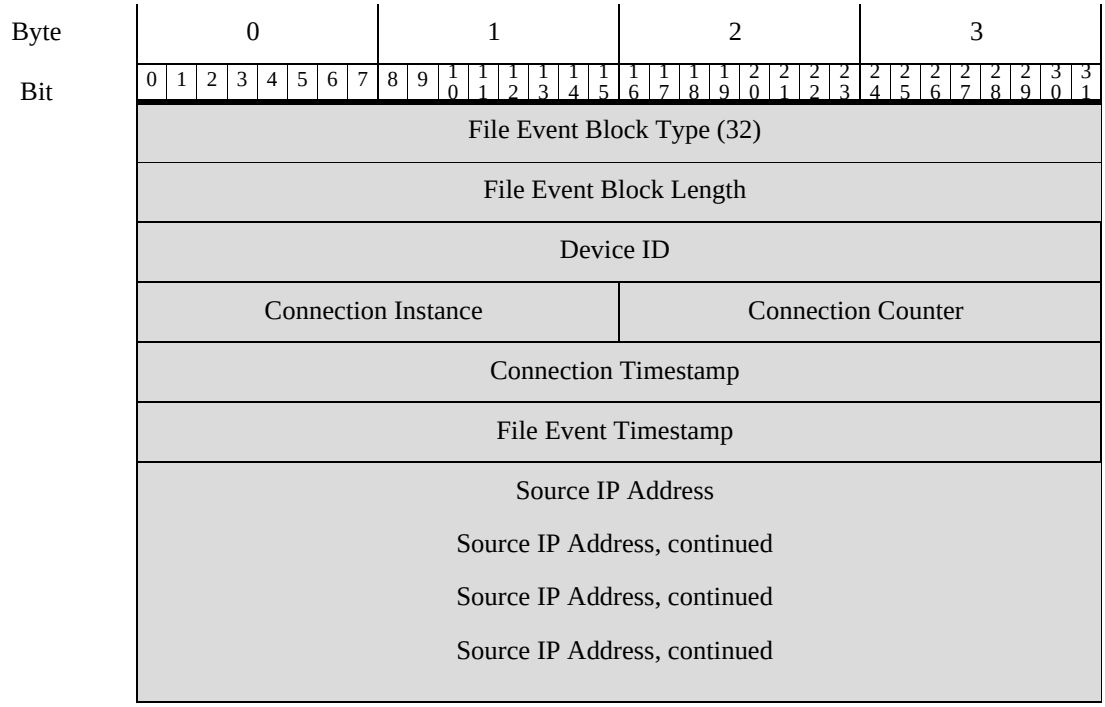
*Table B-47 File Event Data Block Fields (continued)*

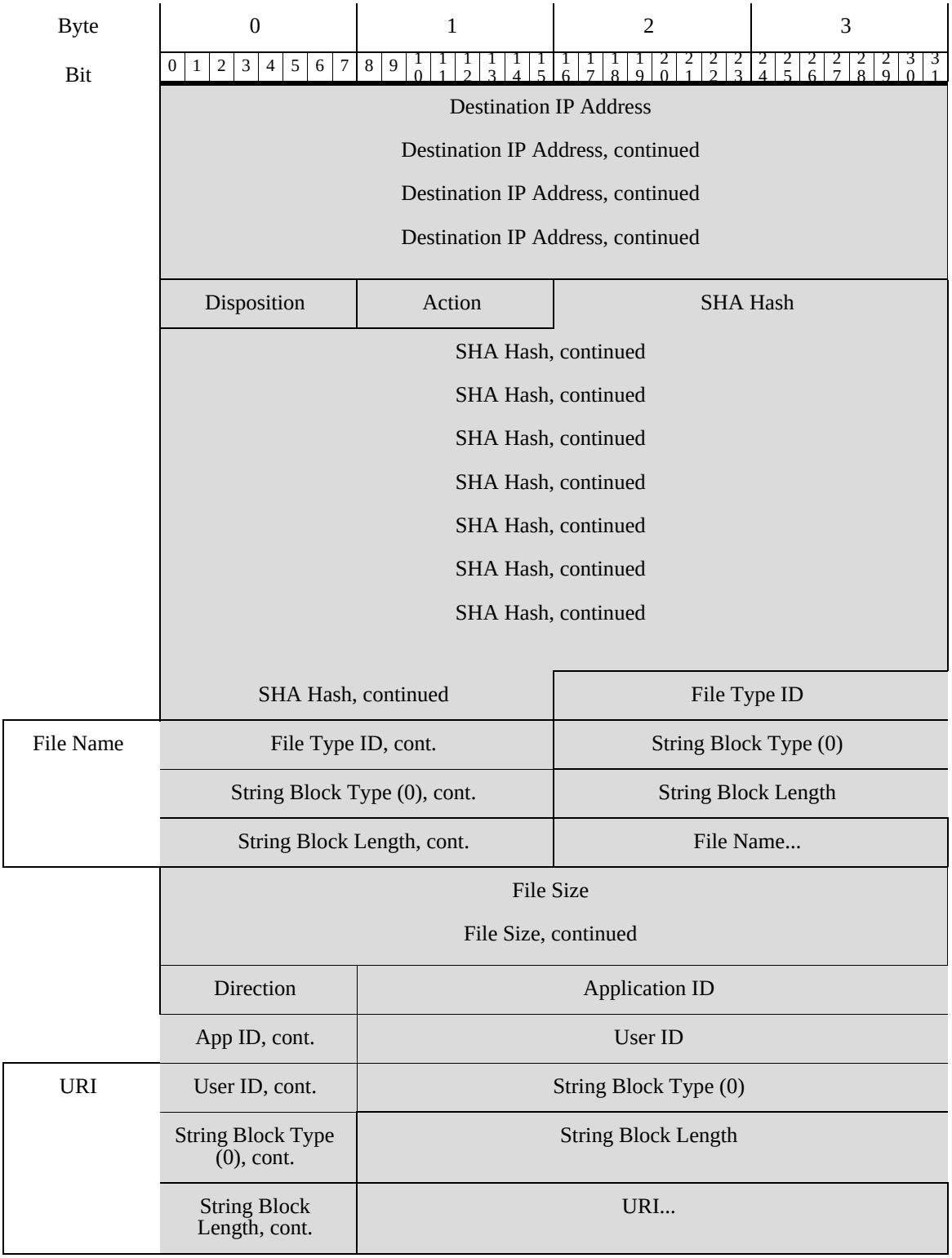
| Field | Data Type | Description |
|---|---|---|
| Protocol | uint8 | IANA protocol number specified by the user. For example:<br>• 1 — ICMP<br>• 4 — IP<br>• 6 — TCP<br>• 17 — UDP<br>This is currently only TCP. |
| Access Control Policy UUID | uint8[16] | Unique identifier for the access control policy that triggered the event. |

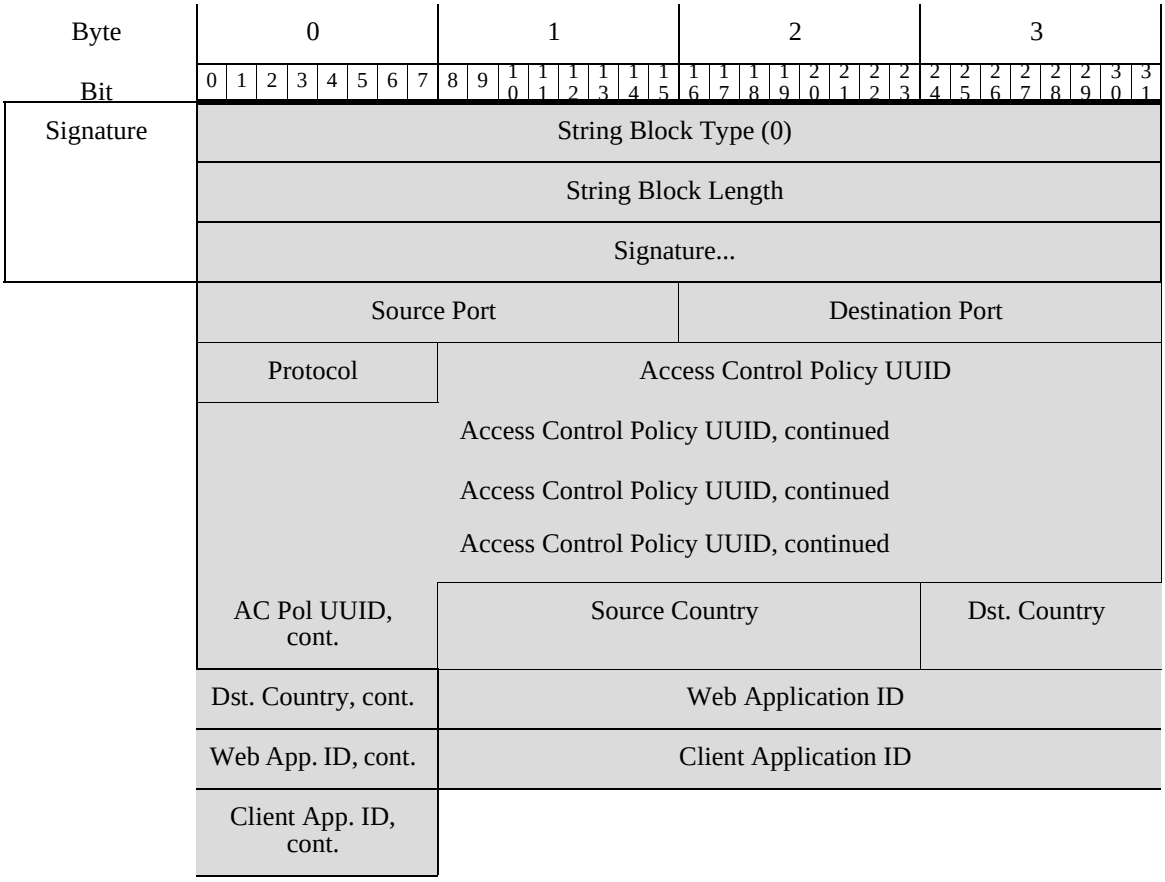
# File Event for 5.2.x

The file event contains information on files that are sent over the network. This includes the connection information, whether the file is malware, and specific information to identify the file. The file event has a block type of 32 in the series 2 group of blocks. It supersedes block type 23. New fields have been added to track source and destination country, as well as the client and web application instances.

The following graphic shows the structure of the File Event data block:




Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Destination IP Address
Destination IP Address, continued
Destination IP Address, continued
Destination IP Address, continued
Disposition
Action
SHA Hash
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
File Type ID
File Name
File Type ID, cont.
String Block Type (0)
String Block Type (0), cont.
String Block Length
String Block Length, cont.
File Name...
File Size
File Size, continued
Direction
Application ID
App ID, cont.
User ID
URI
User ID, cont.
String Block Type (0)
String Block Type (0), cont.
String Block Length
String Block Length, cont.
URI...




The following table describes the fields in the file event data block:

***Table B-48 File Event Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| File Event Block Type | uint32 | Initiates whether file event data block. This value is always `23`. |
| File Event Block Length | uint32 | Total number of bytes in the file event block, including eight bytes for the file event block type and length fields, plus the number of bytes of data that follows. |
| Device ID | uint32 | ID for the device that generated the event. |
| Connection Instance | uint16 | Snort instance on the device that generated the event. Used to link the event with a connection or intrusion event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Connection Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the associated connection event. |
| File Event Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of when the file type is identified and the file event generated. |
| Source IP Address | uint8[16] | IPv4 or IPv6 address for the source of the connection. |

*Table B-48 File Event Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Destination IP Address | uint8[16] | IPv4 or IPv6 address for the destination of the connection. |
| Disposition | uint8 | The malware status of the file. Possible values include:<br>• 1 — CLEAN — The file is clean and does not contain malware.<br>• 2 — NEUTRAL — It is unknown whether the file contains malware.<br>• 3 — MALWARE — The file contains malware.<br>• 4 — CACHE_MISS — The software was unable to send a request to the Cisco cloud for a disposition, or the Cisco cloud services did not respond to the request. |
| Action | uint8 | The action taken on the file based on the file type. Can have the following values:<br>• 1 — Detect<br>• 2 — Block<br>• 3 — Malware Cloud Lookup<br>• 4 — Malware Block<br>• 5 — Malware Allow List |
| SHA Hash | uint8[32] | SHA-256 hash of the file, in binary format. |
| File Type ID | uint32 | ID number that maps to the file type. |
| File Name | string | Name of the file. |
| File Size | uint64 | Size of the file in bytes. |
| Direction | uint8 | Value that indicates whether the file was uploaded or downloaded. Can have the following values:<br>• 1 — Download<br>• 2 — Upload<br>Currently the value depends on the protocol (for example, if the connection is HTTP it is a download). |
| Application ID | uint32 | ID number that maps to the application using the file transfer. |
| User ID | uint32 | ID number for the user logged into the destination host, as identified by the system. |
| URI | string | Uniform Resource Identifier (URI) of the connection. |
| Signature | string | SHA-256 hash of the file, in string format. |
| Source Port | uint16 | Port number for the source of the connection. |
| Destination Port | uint16 | Port number for the destination of the connection. |

*Table B-48 File Event Data Block Fields (continued)*

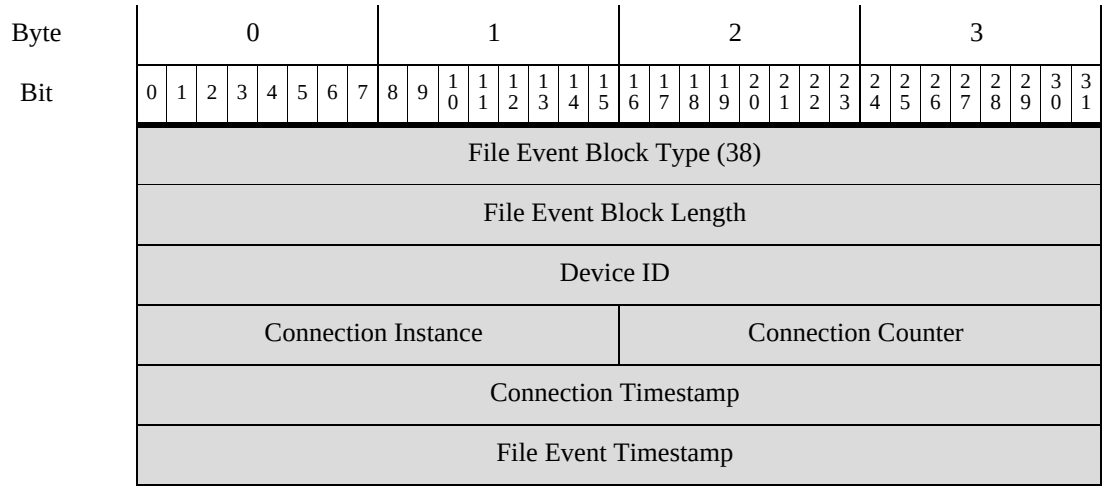
| Field | Data Type | Description |
|---|---|---|
| Protocol | uint8 | IANA protocol number specified by the user. For example:<br>• 1 — ICMP<br>• 4 — IP<br>• 6 — TCP<br>• 17 — UDP<br>This is currently only TCP. |
| Access Control Policy UUID | uint8[16] | Unique identifier for the access control policy that triggered the event. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint16 | Code for the country of the destination host. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |

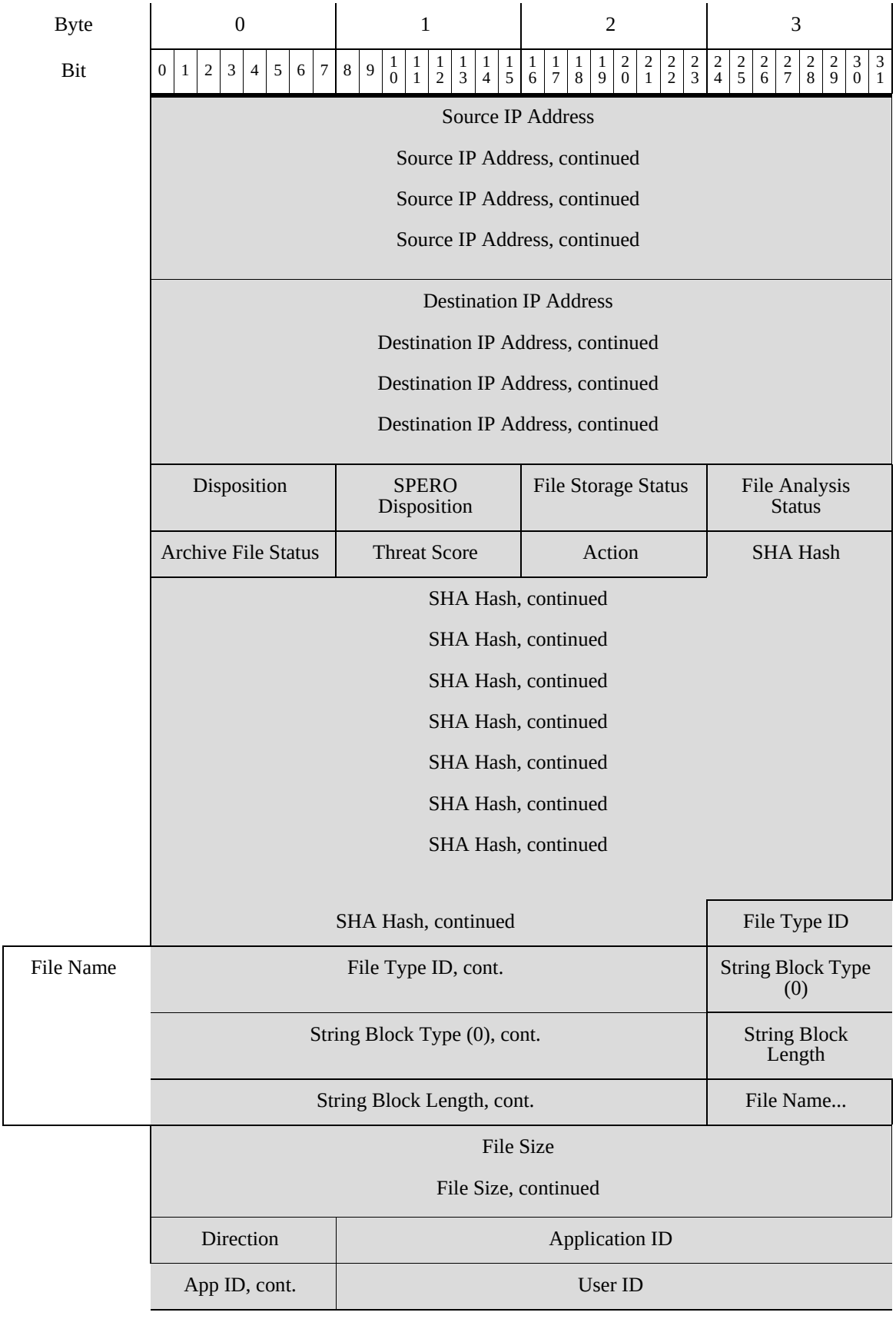
# File Event for 5.3

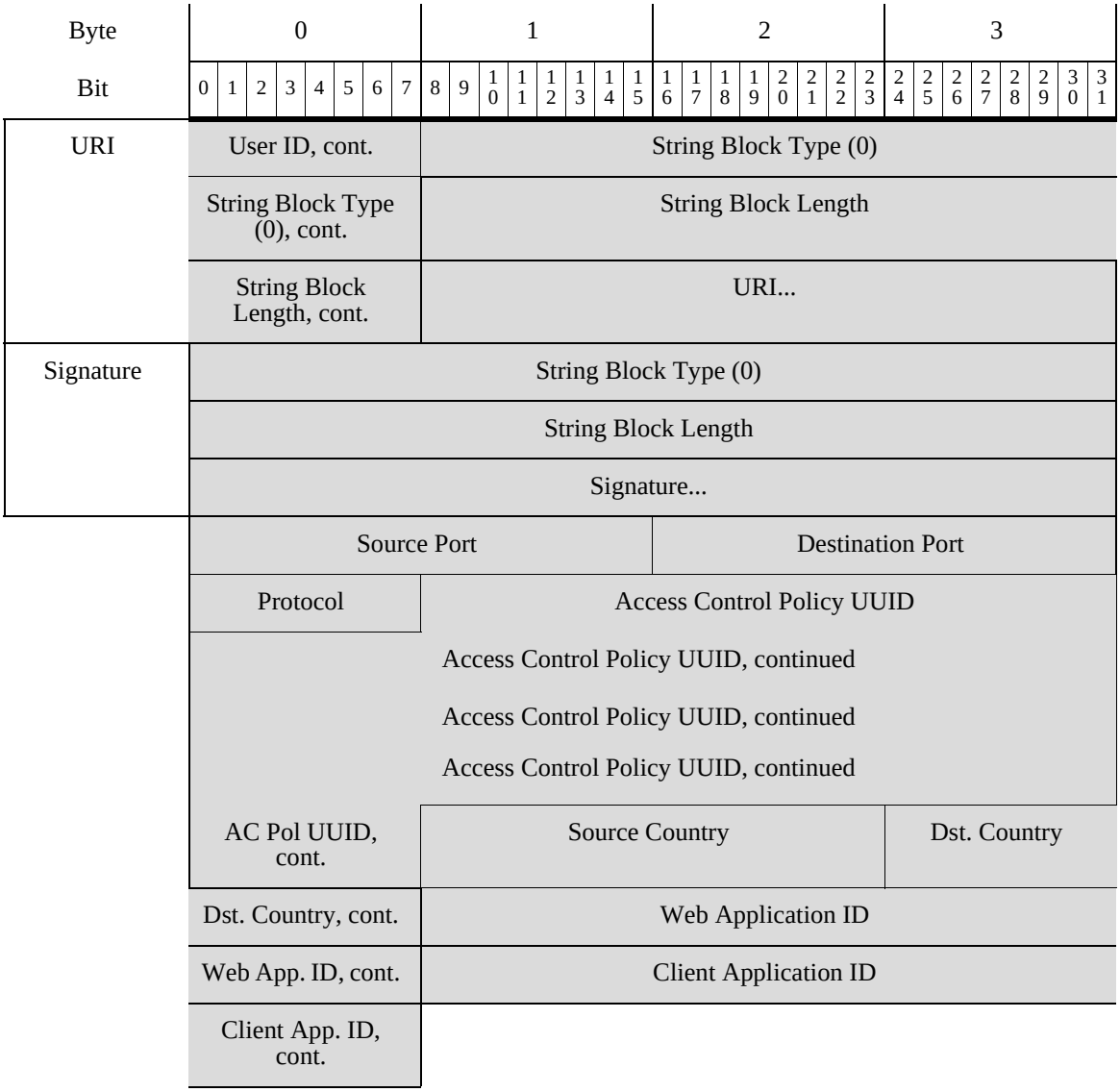
The file event contains information on files that are sent over the network. This includes the connection information, whether the file is malware, and specific information to identify the file. The file event has a block type of 38 in the series 2 group of blocks. It supersedes block type 32. New fields have been added to track dynamic file analysis and file storage.

You request file event records by setting the file event flag—bit 30 in the Request Flags field—in the request message with an event version of 3 and an event code of 111. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.

The following graphic shows the structure of the File Event data block.




Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Source IP Address
Source IP Address, continued
Source IP Address, continued
Source IP Address, continued
Destination IP Address
Destination IP Address, continued
Destination IP Address, continued
Destination IP Address, continued
Disposition
SPERO Disposition
File Storage Status
File Analysis Status
Archive File Status
Threat Score
Action
SHA Hash
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
File Type ID
File Name
File Type ID, cont.
String Block Type (0)
String Block Type (0), cont.
String Block Length
String Block Length, cont.
File Name...
File Size
File Size, continued
Direction
Application ID
App ID, cont.
User ID




The following table describes the fields in the file event data block.

***Table B-49 File Event Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| File Event Block Type | uint32 | Initiates whether file event data block. This value is always 23. |
| File Event Block Length | uint32 | Total number of bytes in the file event block, including eight bytes for the file event block type and length fields, plus the number of bytes of data that follows. |
| Device ID | uint32 | ID for the device that generated the event. |
| Connection Instance | uint16 | Snort instance on the device that generated the event. Used to link the event with a connection or intrusion event. |

*Table B-49 File Event Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Connection Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the associated connection event. |
| File Event Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of when the file type is identified and the file event generated. |
| Source IP Address | uint8[16] | IPv4 or IPv6 address for the source of the connection. |
| Destination IP Address | uint8[16] | IPv4 or IPv6 address for the destination of the connection. |
| Disposition | uint8 | The malware status of the file. Possible values include:<br>• 1 — CLEAN The file is clean and does not contain malware.<br>• 2 — UNKNOWN It is unknown whether the file contains malware.<br>• 3 — MALWARE The file contains malware.<br>• 4 — UNAVAILABLE The software was unable to send a request to the Cisco cloud for a disposition, or the Cisco cloud services did not respond to the request.<br>• 5 — CUSTOM SIGNATURE The file matches a user-defined hash, and is treated in a fashion designated by the user. |
| SPERO Disposition | uint8 | Indicates whether the SPERO signature was used in file analysis. If the value is 1, 2, or 3, SPERO analysis was used. If there is any other value SPERO analysis was not used. |
| File Storage Status | uint8 | The storage status of the file. Possible values are:<br>• 1 — File Stored<br>• 2 — File Stored<br>• 3 — Unable to Store File<br>• 4 — Unable to Store File<br>• 5 — Unable to Store File<br>• 6 — Unable to Store File<br>• 7 — Unable to Store File<br>• 8 — File Size is Too Large<br>• 9 — File Size is Too Small<br>• 10 — Unable to Store File<br>• 11 — File Not Stored, Disposition Unavailable |

*Table B-49 File Event Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| File Analysis Status | uint8 | Indicates whether the file was sent for dynamic analysis. Possible values are:<br>• 0 — File Not Sent for Analysis<br>• 1 — Sent for Analysis<br>• 2 — Sent for Analysis<br>• 4 — Sent for Analysis<br>• 5 — Failed to Send<br>• 6 — Failed to Send<br>• 7 — Failed to Send<br>• 8 — Failed to Send<br>• 9 — File Size is Too Small<br>• 10 — File Size is Too Large<br>• 11 — Sent for Analysis<br>• 12 — Analysis Complete<br>• 13 — Failure (Network Issue)<br>• 14 — Failure (Rate Limit)<br>• 15 — Failure (File Too Large)<br>• 16 — Failure (File Read Error)<br>• 17 — Failure (Internal Library Error)<br>• 19 — File Not Sent, Disposition Unavailable<br>• 20 — Failure (Cannot Run File)<br>• 21 — Failure (Analysis Timeout)<br>• 22 — Sent for Analysis<br>• 23 — File Not Supported |
| Archive File Status | uint8 | This is always 0. |
| Threat Score | uint8 | A numeric value from 0 to 100 based on the potentially malicious behaviors observed during dynamic analysis. |
| Action | uint8 | The action taken on the file based on the file type. Can have the following values:<br>• 1 — Detect<br>• 2 — Block<br>• 3 — Malware Cloud Lookup<br>• 4 — Malware Block<br>• 5 — Malware Allow List |
| SHA Hash | uint8[32] | SHA-256 hash of the file, in binary format. |

*Table B-49 File Event Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| File Type ID | uint32 | ID number that maps to the file type. The meaning of this field is transmitted in the metadata with this event. See AMP for Endpoints File Type Metadata, page 3-38 for more information. |
| File Name | string | Name of the file. |
| File Size | uint64 | Size of the file in bytes. |
| Direction | uint8 | Value that indicates whether the file was uploaded or downloaded. Can have the following values:<br>• 1 — Download<br>• 2 — Upload<br>Currently the value depends on the protocol (for example, if the connection is HTTP it is a download). |
| Application ID | uint32 | ID number that maps to the application using the file transfer. |
| User ID | uint32 | ID number for the user logged into the destination host, as identified by the system. |
| URI | string | Uniform Resource Identifier (URI) of the connection. |
| Signature | string | SHA-256 hash of the file, in string format. |
| Source Port | uint16 | Port number for the source of the connection. |
| Destination Port | uint16 | Port number for the destination of the connection. |
| Protocol | uint8 | IANA protocol number specified by the user. For example:<br>• 1 — ICMP<br>• 4 — IP<br>• 6 — TCP<br>• 17 — UDP<br>This is currently only TCP. |
| Access Control Policy UUID | uint8[16] | Unique identifier for the access control policy that triggered the event. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint16 | Code for the country of the destination host. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |

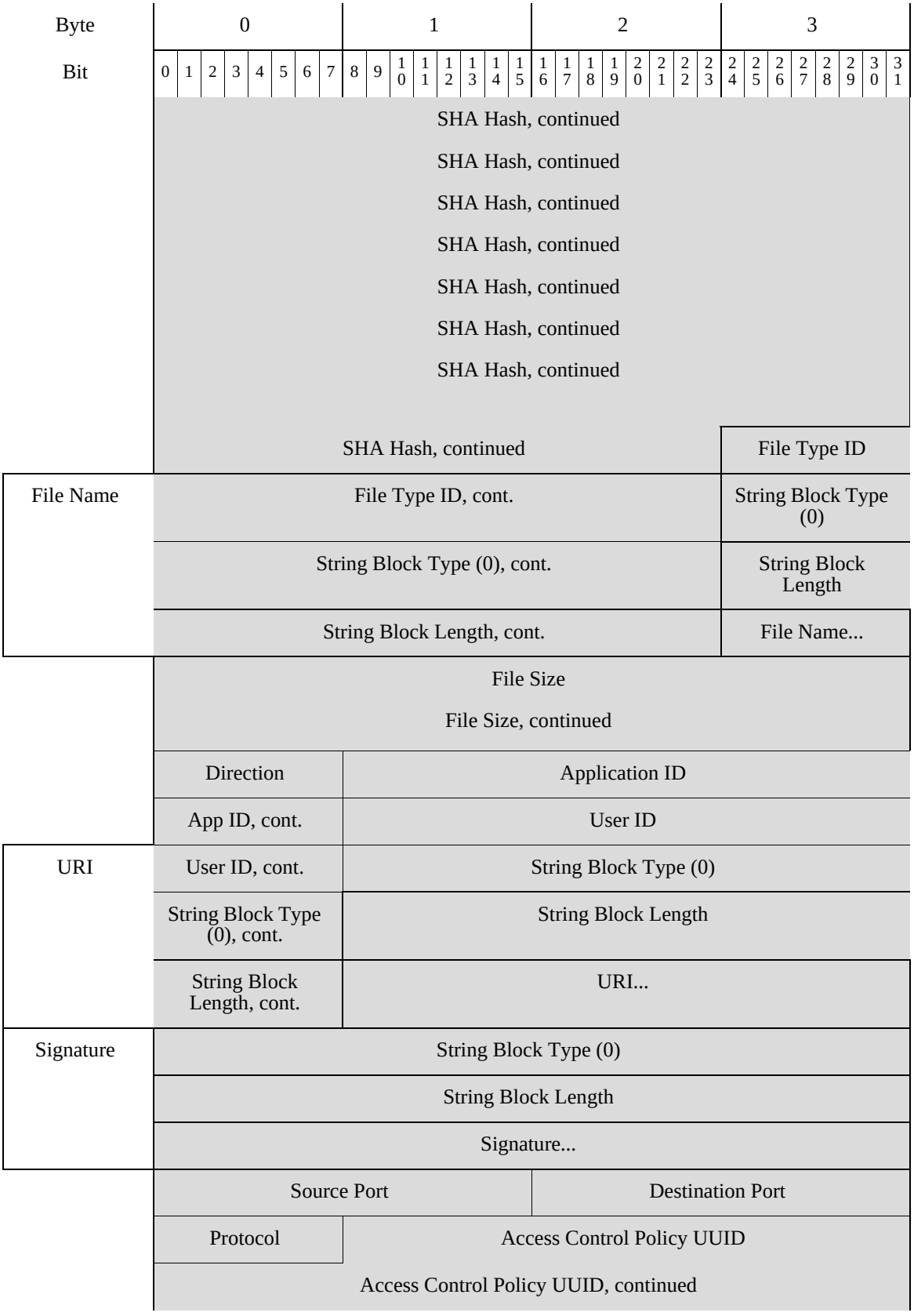
# File Event for 5.3.1

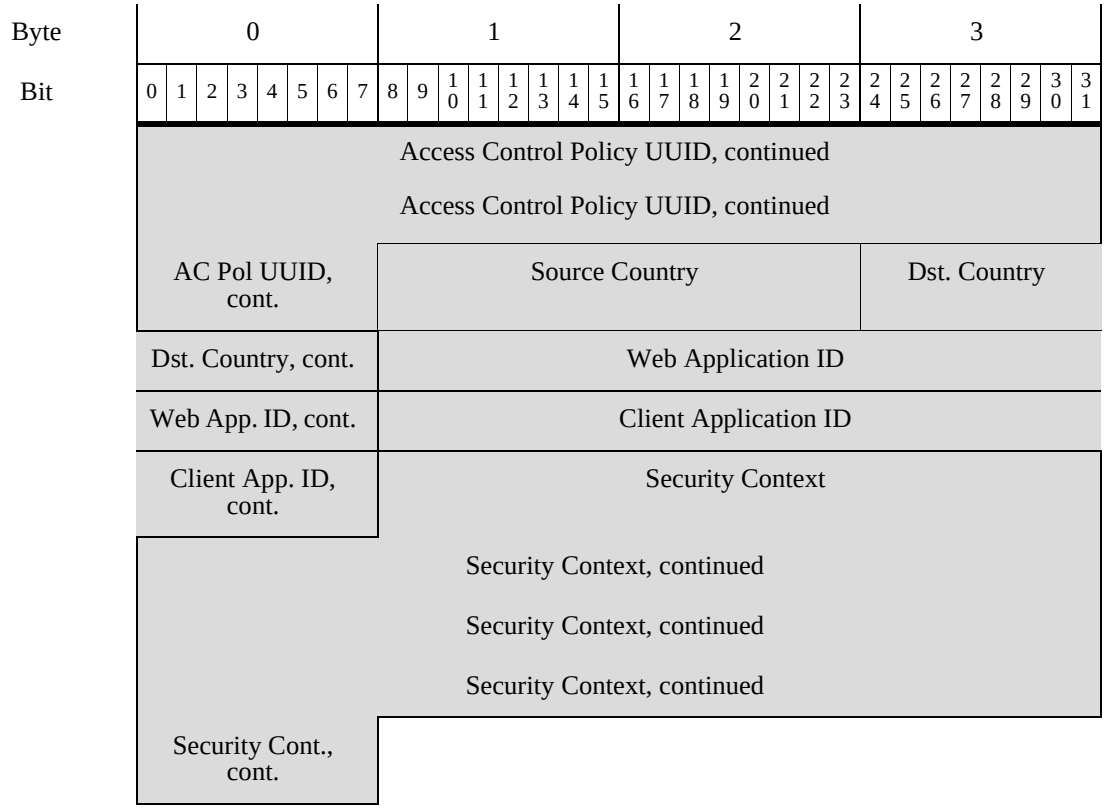
The file event contains information on files that are sent over the network. This includes the connection information, whether the file is malware, and specific information to identify the file. The file event has a block type of 43 in the series 2 group of blocks. It supersedes block type 38. A security context field has been added.

You request file event records by setting the file event flag—bit 30 in the Request Flags field—in the request message with an event version of 4 and an event code of 111. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.

The following graphic shows the structure of the File Event data block.

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| File Event Block Type (43) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Event Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Connection Instance | | | | | | | | | | | | | | | | | Connection Counter | | | | | | | | | | | | | | | |
| Connection Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Event Timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Source IP Address<br>Source IP Address, continued<br>Source IP Address, continued<br>Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Destination IP Address<br>Destination IP Address, continued<br>Destination IP Address, continued<br>Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Disposition | | | | | | | | | SPERO Disposition | | | | | | | | File Storage Status | | | | | | | | File Analysis Status | | | | | | | |
| Archive File Status | | | | | | | | | Threat Score | | | | | | | | Action | | | | | | | | SHA Hash | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
SHA Hash, continued
File Type ID
File Name
File Type ID, cont.
String Block Type (0)
String Block Type (0), cont.
String Block Length
String Block Length, cont.
File Name...
File Size
File Size, continued
Direction
Application ID
App ID, cont.
User ID
URI
User ID, cont.
String Block Type (0)
String Block Type (0), cont.
String Block Length
String Block Length, cont.
URI...
Signature
String Block Type (0)
String Block Length
Signature...
Source Port
Destination Port
Protocol
Access Control Policy UUID
Access Control Policy UUID, continued




The following table describes the fields in the file event data block.

***Table B-50 File Event Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| File Event Block Type | uint32 | Initiates whether file event data block. This value is always `43`. |
| File Event Block Length | uint32 | Total number of bytes in the file event block, including eight bytes for the file event block type and length fields, plus the number of bytes of data that follows. |
| Device ID | uint32 | ID for the device that generated the event. |
| Connection Instance | uint16 | Snort instance on the device that generated the event. Used to link the event with a connection or intrusion event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Connection Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the associated connection event. |
| File Event Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of when the file type is identified and the file event generated. |
| Source IP Address | uint8[16] | IPv4 or IPv6 address for the source of the connection. |
| Destination IP Address | uint8[16] | IPv4 or IPv6 address for the destination of the connection. |

*Table B-50 File Event Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Disposition | uint8 | The malware status of the file. Possible values include:<br>• 1 — CLEAN The file is clean and does not contain malware.<br>• 2 — UNKNOWN It is unknown whether the file contains malware.<br>• 3 — MALWARE The file contains malware.<br>• 4 — UNAVAILABLE The software was unable to send a request to the Cisco cloud for a disposition, or the Cisco cloud services did not respond to the request.<br>• 5 — CUSTOM SIGNATURE The file matches a user-defined hash, and is treated in a fashion designated by the user. |
| SPERO Disposition | uint8 | Indicates whether the SPERO signature was used in file analysis. If the value is 1, 2, or 3, SPERO analysis was used. If there is any other value SPERO analysis was not used. |
| File Storage Status | uint8 | The storage status of the file. Possible values are:<br>• 1 — File Stored<br>• 2 — File Stored<br>• 3 — Unable to Store File<br>• 4 — Unable to Store File<br>• 5 — Unable to Store File<br>• 6 — Unable to Store File<br>• 7 — Unable to Store File<br>• 8 — File Size is Too Large<br>• 9 — File Size is Too Small<br>• 10 — Unable to Store File<br>• 11 — File Not Stored, Disposition Unavailable |

*Table B-50 File Event Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| File Analysis Status | uint8 | Indicates whether the file was sent for dynamic analysis. Possible values are:<br>• 0 — File Not Sent for Analysis<br>• 1 — Sent for Analysis<br>• 2 — Sent for Analysis<br>• 4 — Sent for Analysis<br>• 5 — Failed to Send<br>• 6 — Failed to Send<br>• 7 — Failed to Send<br>• 8 — Failed to Send<br>• 9 — File Size is Too Small<br>• 10 — File Size is Too Large<br>• 11 — Sent for Analysis<br>• 12 — Analysis Complete<br>• 13 — Failure (Network Issue)<br>• 14 — Failure (Rate Limit)<br>• 15 — Failure (File Too Large)<br>• 16 — Failure (File Read Error)<br>• 17 — Failure (Internal Library Error)<br>• 19 — File Not Sent, Disposition Unavailable<br>• 20 — Failure (Cannot Run File)<br>• 21 — Failure (Analysis Timeout)<br>• 22 — Sent for Analysis<br>• 23 — File Not Supported<br>• 23 —File Transmit File Capacity Handled — File capacity handled (stored on the sensor) because file could not be submitted to the sandbox for analysis<br>• 25 — File Transmit Server Limited Exceeded Capacity Handled — File capacity handled due to rate limiting on server<br>• 26 — Communication Failure — File capacity handled due to cloud connectivity failure<br>• 27 — Not Sent — File not sent due to configuration<br>• 28 — Preclass No Match —File not sent for dynamic analysis since pre-classification didn't find any embedded or suspicious object in the file<br>• 29 — Transmit Sent Sandbox Private Cloud — File sent to the private cloud for dynamic analysis<br>• 30 — Transmit Not Send Sendbox Private Cloud - File not send to the private cloud for analysis |

*Table B-50 File Event Data Block Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Archive File Status | uint8 | This is always `0`. |
| Threat Score | uint8 | A numeric value from 0 to 100 based on the potentially malicious behaviors observed during dynamic analysis. |
| Action | uint8 | The action taken on the file based on the file type. Can have the following values:<br>• `1` — Detect<br>• `2` — Block<br>• `3` — Malware Cloud Lookup<br>• `4` — Malware Block<br>• `5` — Malware Allow List |
| SHA Hash | uint8[32] | SHA-256 hash of the file, in binary format. |
| File Type ID | uint32 | ID number that maps to the file type. The meaning of this field is transmitted in the metadata with this event. See AMP for Endpoints File Type Metadata, page 3-38 for more information. |
| File Name | string | Name of the file. |
| File Size | uint64 | Size of the file in bytes. |
| Direction | uint8 | Value that indicates whether the file was uploaded or downloaded. Can have the following values:<br>• `1` — Download<br>• `2` — Upload<br>Currently the value depends on the protocol (for example, if the connection is HTTP it is a download). |
| Application ID | uint32 | ID number that maps to the application using the file transfer. |
| User ID | uint32 | ID number for the user logged into the destination host, as identified by the system. |
| URI | string | Uniform Resource Identifier (URI) of the connection. |
| Signature | string | SHA-256 hash of the file, in string format. |
| Source Port | uint16 | Port number for the source of the connection. |
| Destination Port | uint16 | Port number for the destination of the connection. |
| Protocol | uint8 | IANA protocol number specified by the user. For example:<br>• `1` — ICMP<br>• `4` — IP<br>• `6` — TCP<br>• `17` — UDP<br>This is currently only TCP. |

*Table B-50 File Event Data Block Fields (continued)*

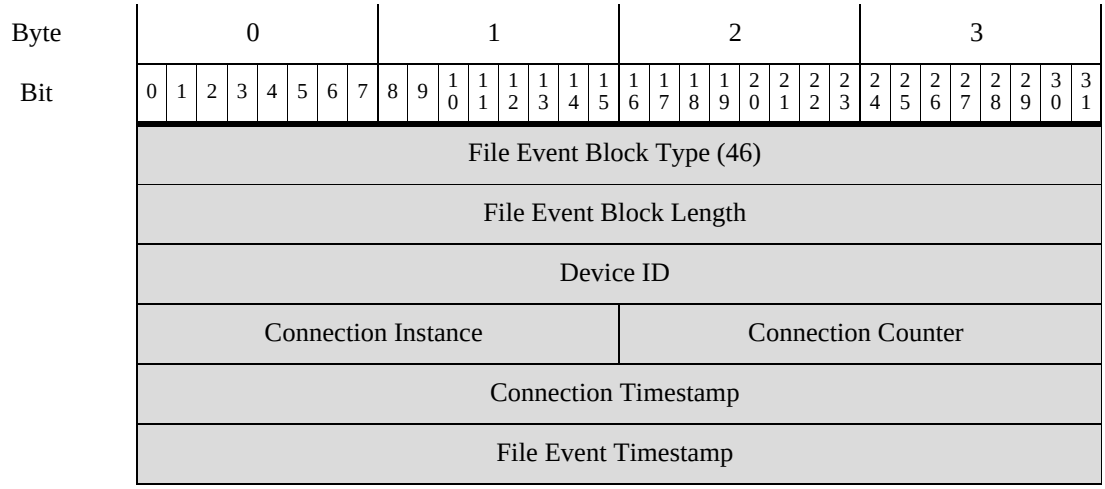
| Field | Data Type | Description |
|---|---|---|
| Access Control Policy UUID | uint8[16] | Unique identifier for the access control policy that triggered the event. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint16 | Code for the country of the destination host. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |

# File Event for 5.4.x

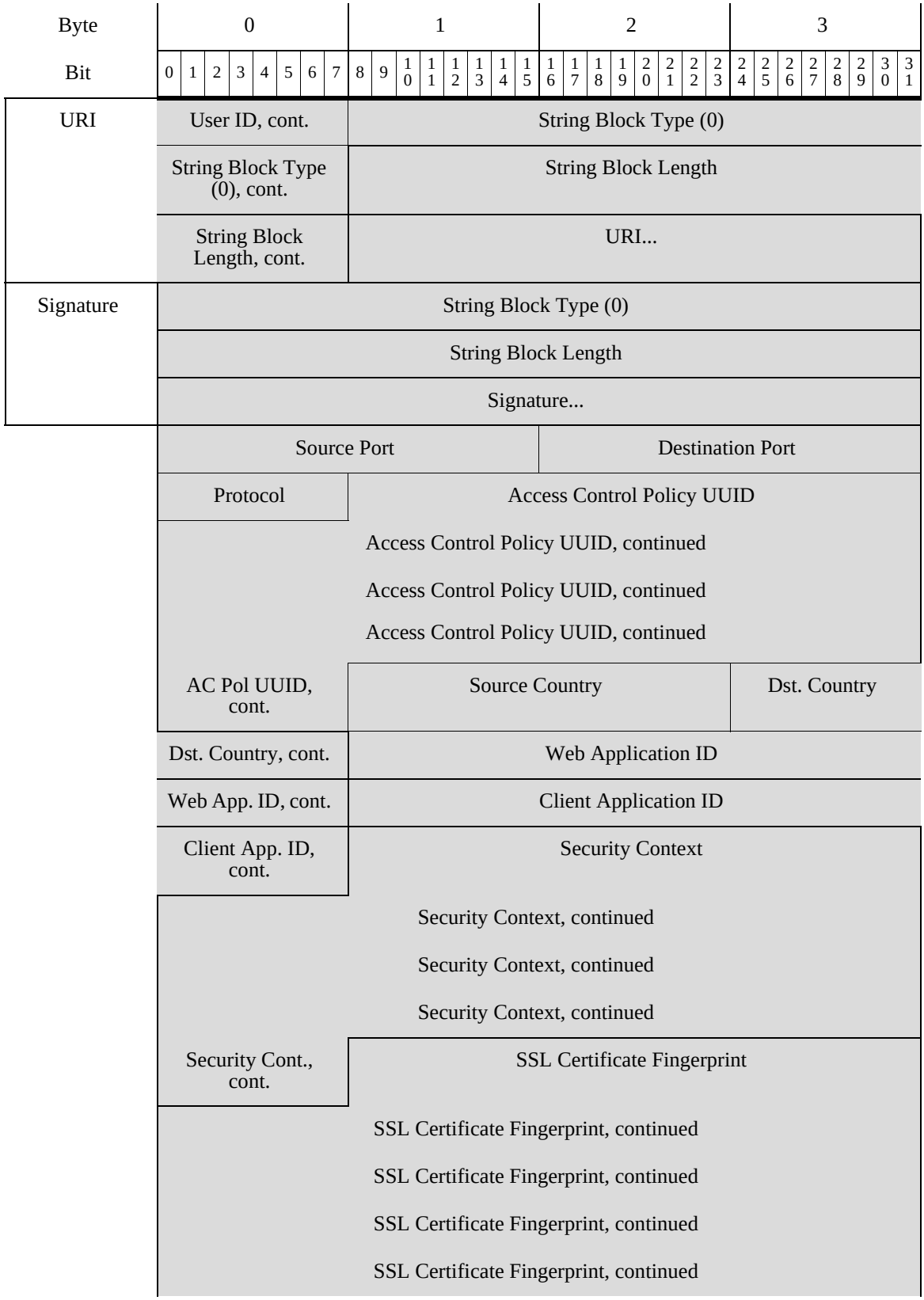
The file event contains information on files that are sent over the network. This includes the connection information, whether the file is malware, and specific information to identify the file. The file event has a block type of 46 in the series 2 group of blocks. It supersedes block type 43. Fields for SSL and file archive support have been added.

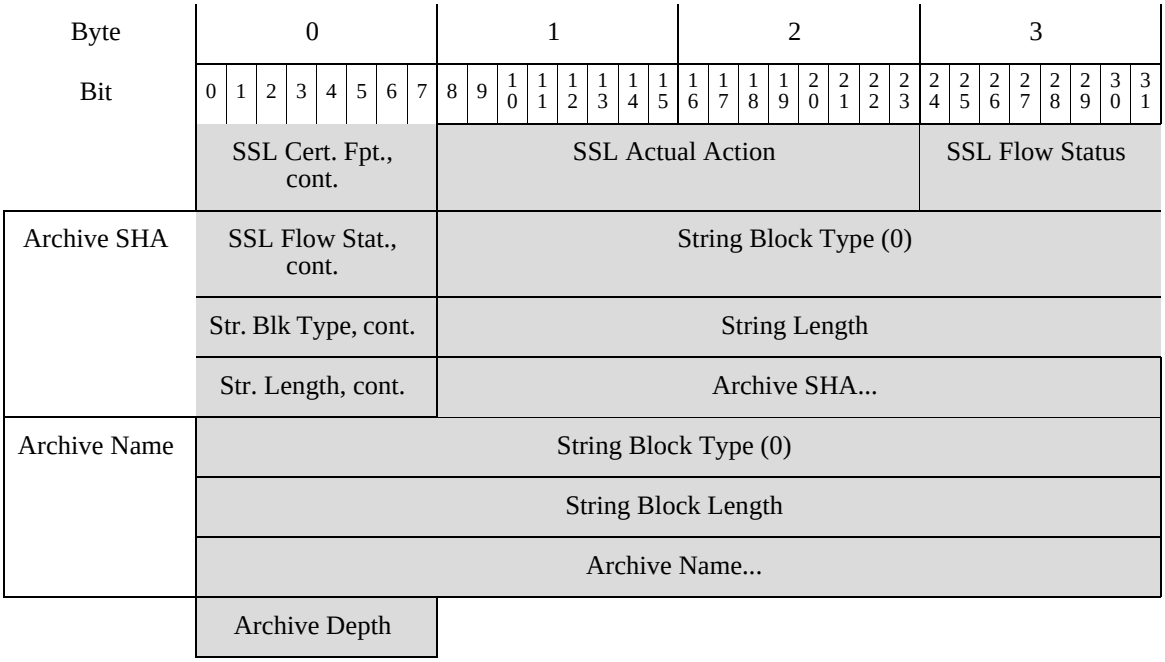
You request file event records by setting the file event flag—bit 30 in the Request Flags field—in the request message with an event version of 5 and an event code of 111. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.

The following graphic shows the structure of the File Event data block.



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Source IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Destination IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Disposition | | | | | | | | SPERO Disposition | | | | | | | | File Storage Status | | | | | | | | File Analysis Status | | | | | | | |
| | Archive File Status | | | | | | | | Threat Score | | | | | | | | Action | | | | | | | | SHA Hash | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | File Type ID | | | | | | | |
| File Name | File Type ID, cont. | | | | | | | | | | | | | | | | | | | | | | | | String Block Type (0) | | | | | | | |
| | String Block Type (0), cont. | | | | | | | | | | | | | | | | | | | | | | | | String Block Length | | | | | | | |
| | String Block Length, cont. | | | | | | | | | | | | | | | | | | | | | | | | File Name... | | | | | | | |
| | File Size | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Size, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Direction | | | | | | | | Application ID | | | | | | | | | | | | | | | | | | | | | | | |
| | App ID, cont. | | | | | | | | User ID | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
URI
User ID, cont.
String Block Type (0)
String Block Type (0), cont.
String Block Length
String Block Length, cont.
URI...
Signature
String Block Type (0)
String Block Length
Signature...
Source Port
Destination Port
Protocol
Access Control Policy UUID
Access Control Policy UUID, continued
Access Control Policy UUID, continued
Access Control Policy UUID, continued
AC Pol UUID, cont.
Source Country
Dst. Country
Dst. Country, cont.
Web Application ID
Web App. ID, cont.
Client Application ID
Client App. ID, cont.
Security Context
Security Context, continued
Security Context, continued
Security Context, continued
Security Cont., cont.
SSL Certificate Fingerprint
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued




The following table describes the fields in the file event data block.

***Table B-51 File Event Data Block for 5.4.x Fields***

| Field | Data Type | Description |
|---|---|---|
| File Event Block Type | uint32 | Initiates whether file event data block. This value is always 46. |
| File Event Block Length | uint32 | Total number of bytes in the file event block, including eight bytes for the file event block type and length fields, plus the number of bytes of data that follows. |
| Device ID | uint32 | ID for the device that generated the event. |
| Connection Instance | uint16 | Snort instance on the device that generated the event. Used to link the event with a connection or intrusion event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Connection Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the associated connection event. |
| File Event Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of when the file type is identified and the file event generated. |
| Source IP Address | uint8[16] | IPv4 or IPv6 address for the source of the connection. |
| Destination IP Address | uint8[16] | IPv4 or IPv6 address for the destination of the connection. |

*Table B-51 File Event Data Block for 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Disposition | uint8 | The malware status of the file. Possible values include:<br>• 1 — CLEAN The file is clean and does not contain malware.<br>• 2 — UNKNOWN It is unknown whether the file contains malware.<br>• 3 — MALWARE The file contains malware.<br>• 4 — UNAVAILABLE The software was unable to send a request to the Cisco cloud for a disposition, or the Cisco cloud services did not respond to the request.<br>• 5 — CUSTOM SIGNATURE The file matches a user-defined hash, and is treated in a fashion designated by the user. |
| SPERO Disposition | uint8 | Indicates whether the SPERO signature was used in file analysis. If the value is 1, 2, or 3, SPERO analysis was used. If there is any other value SPERO analysis was not used. |
| File Storage Status | uint8 | The storage status of the file. Possible values are:<br>• 1 — File Stored<br>• 2 — File Stored<br>• 3 — Unable to Store File<br>• 4 — Unable to Store File<br>• 5 — Unable to Store File<br>• 6 — Unable to Store File<br>• 7 — Unable to Store File<br>• 8 — File Size is Too Large<br>• 9 — File Size is Too Small<br>• 10 — Unable to Store File<br>• 11 — File Not Stored, Disposition Unavailable |

*Table B-51 File Event Data Block for 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| File Analysis Status | uint8 | Indicates whether the file was sent for dynamic analysis. Possible values are:<br>• 0 — File Not Sent for Analysis<br>• 1 — Sent for Analysis<br>• 2 — Sent for Analysis<br>• 4 — Sent for Analysis<br>• 5 — Failed to Send<br>• 6 — Failed to Send<br>• 7 — Failed to Send<br>• 8 — Failed to Send<br>• 9 — File Size is Too Small<br>• 10 — File Size is Too Large<br>• 11 — Sent for Analysis<br>• 12 — Analysis Complete<br>• 13 — Failure (Network Issue)<br>• 14 — Failure (Rate Limit)<br>• 15 — Failure (File Too Large)<br>• 16 — Failure (File Read Error)<br>• 17 — Failure (Internal Library Error)<br>• 19 — File Not Sent, Disposition Unavailable<br>• 20 — Failure (Cannot Run File)<br>• 21 — Failure (Analysis Timeout)<br>• 22 — Sent for Analysis<br>• 23 — File Not Supported |

*Table B-51 File Event Data Block for 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Archive File Status | uint8 | The status of an archive being inspected. Can have the following values:<br>• 0 — N/A — File is not being inspected as an archive<br>• 1 — Pending — Archive is being inspected<br>• 2 — Extracted — Successfully inspected without any problems<br>• 3 — Failed — Failed to inspect, insufficient system resources<br>• 4 — Depth Exceeded — Successful, but archive exceeded the nested inspection depth<br>• 5 — Encrypted — Partially Successful, Archive was or contains an archive that is encrypted<br>• 6 — Not Inspectable — Partially Successful, File is possibly Malformed or Corrupt |
| Threat Score | uint8 | A numeric value from 0 to 100 based on the potentially malicious behaviors observed during dynamic analysis. |
| Action | uint8 | The action taken on the file based on the file type. Can have the following values:<br>• 1 — Detect<br>• 2 — Block<br>• 3 — Malware Cloud Lookup<br>• 4 — Malware Block<br>• 5 — Malware Allow List<br>• 6 — Cloud Lookup Timeout<br>• 7 — Custom Detection<br>• 8 — Custom Detection Block<br>• 9 — Archive Block (Depth Exceeded)<br>• 10 — Archive Block (Encrypted)<br>• 11 — Archive Block (Failed to Inspect) |
| SHA Hash | uint8[32] | SHA-256 hash of the file, in binary format. |
| File Type ID | uint32 | ID number that maps to the file type. The meaning of this field is transmitted in the metadata with this event. See AMP for Endpoints File Type Metadata, page 3-38 for more information. |
| File Name | string | Name of the file. |
| File Size | uint64 | Size of the file in bytes. |

*Table B-51 File Event Data Block for 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Direction | uint8 | Value that indicates whether the file was uploaded or downloaded. Can have the following values:<br>• 1 — Download<br>• 2 — Upload<br>Currently the value depends on the protocol (for example, if the connection is HTTP it is a download). |
| Application ID | uint32 | ID number that maps to the application using the file transfer. |
| User ID | uint32 | ID number for the user logged into the destination host, as identified by the system. |
| URI | string | Uniform Resource Identifier (URI) of the connection. |
| Signature | string | SHA-256 hash of the file, in string format. |
| Source Port | uint16 | Port number for the source of the connection. |
| Destination Port | uint16 | Port number for the destination of the connection. |
| Protocol | uint8 | IANA protocol number specified by the user. For example:<br>• 1 — ICMP<br>• 4 — IP<br>• 6 — TCP<br>• 17 — UDP<br>This is currently only TCP. |
| Access Control Policy UUID | uint8[16] | Unique identifier for the access control policy that triggered the event. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint16 | Code for the country of the destination host. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |
| SSL Certificate Fingerprint | uint8[20] | SHA1 hash of the SSL Server certificate. |

*Table B-51 File Event Data Block for 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Actual Action | uint16 | The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |

*Table B-51 File Event Data Block for 5.4.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Status | uint16 | Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'No Match'<br>• 2 — 'Success'<br>• 3 — 'Uncached Session'<br>• 4 — 'Unknown Cipher Suite'<br>• 5 — 'Unsupported Cipher Suite'<br>• 6 — 'Unsupported SSL Version'<br>• 7 — 'SSL Compression Used'<br>• 8 — 'Session Undecryptable in Passive Mode'<br>• 9 — 'Handshake Error'<br>• 10 — 'Decryption Error'<br>• 11 — 'Pending Server Name Category Lookup'<br>• 12 — 'Pending Common Name Category Lookup'<br>• 13 — 'Internal Error'<br>• 14 — 'Network Parameters Unavailable'<br>• 15 — 'Invalid Server Certificate Handle'<br>• 16 — 'Server Certificate Fingerprint Unavailable'<br>• 17 — 'Cannot Cache Subject DN'<br>• 18 — 'Cannot Cache Issuer DN'<br>• 19 — 'Unknown SSL Version'<br>• 20 — 'External Certificate List Unavailable'<br>• 21 — 'External Certificate Fingerprint Unavailable'<br>• 22 — 'Internal Certificate List Invalid'<br>• 23 — 'Internal Certificate List Unavailable'<br>• 24 — 'Internal Certificate Unavailable'<br>• 25 — 'Internal Certificate Fingerprint Unavailable'<br>• 26 — 'Server Certificate Validation Unavailable'<br>• 27 — 'Server Certificate Validation Failure'<br>• 28 — 'Invalid Action' |
| String Block Type | uint32 | Initiates a String data block containing the Archive SHA. This value is always 0. |

*Table B-51 File Event Data Block for 5.4.x Fields (continued)*

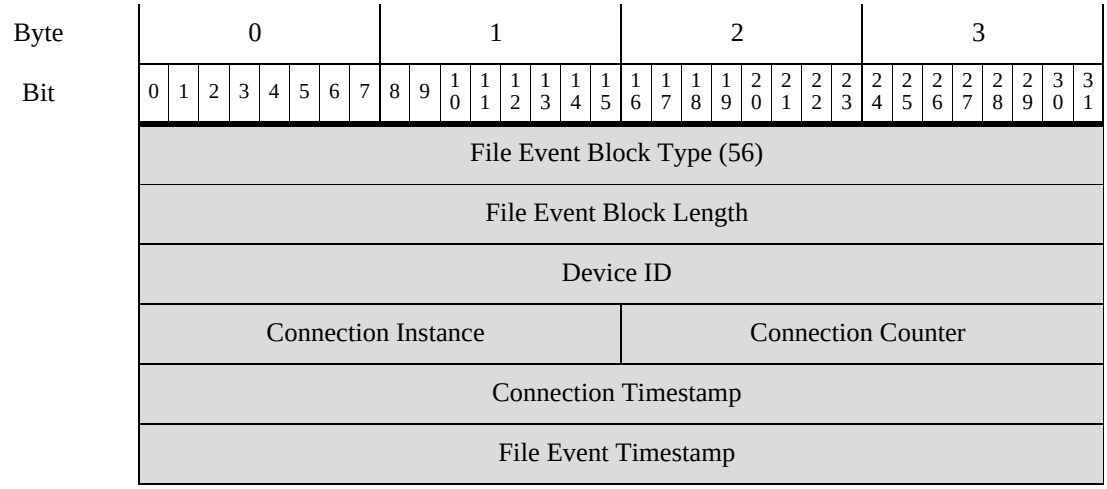
| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the Archive SHA String data block, including eight bytes for the block type and header fields plus the number of bytes in the intrusion policy name. |
| Archive SHA | string | SHA1 hash of the parent archive in which the file is contained. |
| String Block Type | uint32 | Initiates a String data block containing the Archive Name. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Archive Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the intrusion policy name. |
| Archive Name | string | Name of the parent archive. |
| Archive Depth | uint8 | Number of layers in which the file is nested. For example, if a text file is in a zip archive, this has a value of `1`. |

# File Event for 6.x

The File Event data block contains information on files that are sent over the network. This includes the connection information, whether the file is malware, and specific information to identify the file. The file event has a block type of 56 in the series 2 group of blocks. It supersedes block type 46 and is superseded by block type 79. Fields for ISE integration, file analysis, local malware analysis, and capacity handling statuses have been added.

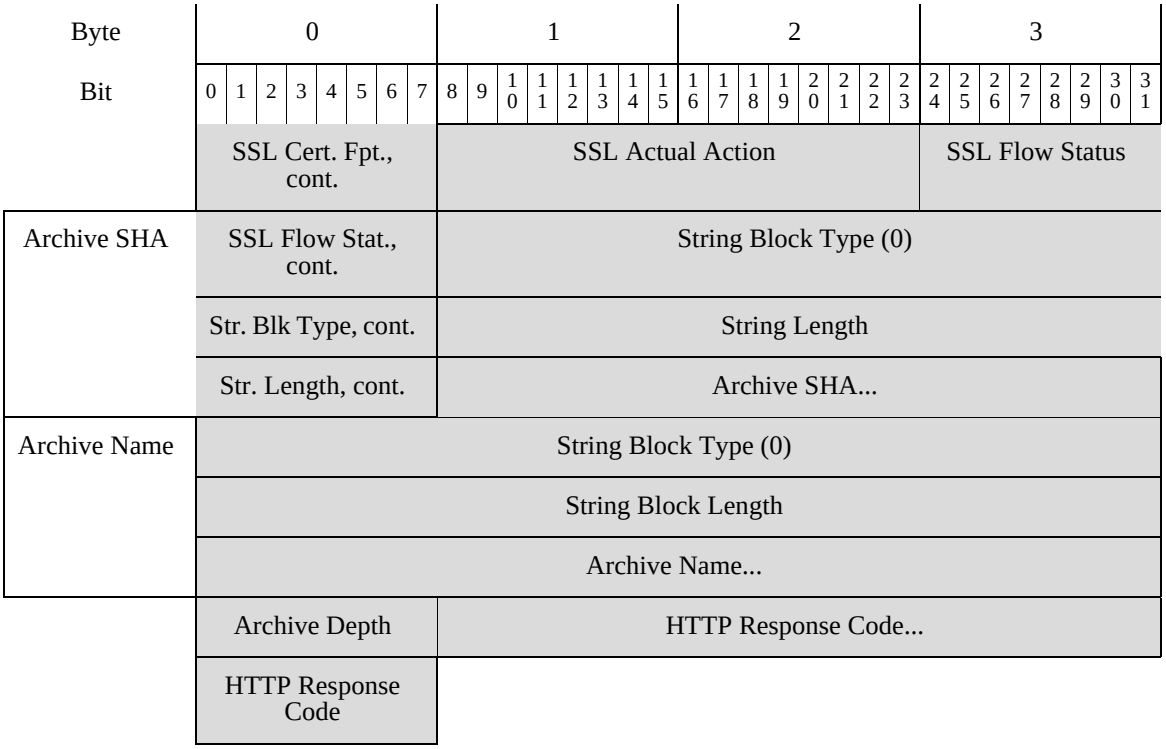
You request file event records by setting the file event flag—bit 30 in the Request Flags field—in the request message with an event version of 5 and an event code of 111. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.

The following graphic shows the structure of the File Event data block.



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Source IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Source IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Destination IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Destination IP Address, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Disposition | | | | | | | | SPERO Disposition | | | | | | | | File Storage Status | | | | | | | | File Analysis Status | | | | | | | |
| | Local Malware Analysis Stat. | | | | | | | | Archive File Status | | | | | | | | Threat Score | | | | | | | | Action | | | | | | | |
| | SHA Hash | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SHA Hash, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Type ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| File Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Name... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Size | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | File Size, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Direction | | | | | | | | Application ID | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | App ID, cont. | | | | | | | | User ID | | | | | | | | | | | | | | | | | | | | | | | |
| URI | User ID, cont. | | | | | | | | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Type (0), cont. | | | | | | | | String Block Length | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length, cont. | | | | | | | | URI... | | | | | | | | | | | | | | | | | | | | | | | |
| Signature | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Signature... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Source Port | | | | | | | | | | | | | | | | Destination Port | | | | | | | | | | | | | | | |
| | Protocol | | | | | | | | Access Control Policy UUID | | | | | | | | | | | | | | | | | | | | | | | |
| | Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Access Control Policy UUID, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | AC Pol UUID, cont. | | | | | | | | Source Country | | | | | | | | | | | | | | | | Dst. Country | | | | | | | |
| | Dst. Country, cont. | | | | | | | | Web Application ID | | | | | | | | | | | | | | | | | | | | | | | |
| | Web App. ID, cont. | | | | | | | | Client Application ID | | | | | | | | | | | | | | | | | | | | | | | |
| | Client App. ID, cont. | | | | | | | | Security Context | | | | | | | | | | | | | | | | | | | | | | | |
| | Security Context, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Security Context, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Security Context, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Security Cont., cont. | | | | | | | | SSL Certificate Fingerprint | | | | | | | | | | | | | | | | | | | | | | | |
| | SSL Certificate Fingerprint, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SSL Certificate Fingerprint, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SSL Certificate Fingerprint, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | SSL Certificate Fingerprint, continued | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |




The following table describes the fields in the file event data block.

***Table B-52 File Event Data Block for 6.x Fields***

| Field | Data Type | Description |
|---|---|---|
| File Event Block Type | uint32 | Initiates whether file event data block. This value is always 56. |
| File Event Block Length | uint32 | Total number of bytes in the file event block, including eight bytes for the file event block type and length fields, plus the number of bytes of data that follows. |
| Device ID | uint32 | ID for the device that generated the event. |
| Connection Instance | uint16 | Snort instance on the device that generated the event. Used to link the event with a connection or intrusion event. |
| Connection Counter | uint16 | Value used to distinguish between connection events that happen during the same second. |
| Connection Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of the associated connection event. |
| File Event Timestamp | uint32 | UNIX timestamp (seconds since 01/01/1970) of when the file type is identified and the file event generated. |
| Source IP Address | uint8[16] | IPv4 or IPv6 address for the source of the connection. |
| Destination IP Address | uint8[16] | IPv4 or IPv6 address for the destination of the connection. |

*Table B-52 File Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Disposition | uint8 | The malware status of the file. Possible values include:<br>• 1 — CLEAN The file is clean and does not contain malware.<br>• 2 — UNKNOWN It is unknown whether the file contains malware.<br>• 3 — MALWARE The file contains malware.<br>• 4 — UNAVAILABLE The software was unable to send a request to the AMP cloud for a disposition, or the AMP cloud services did not respond to the request.<br>• 5 — CUSTOM SIGNATURE The file matches a user-defined hash, and is treated in a fashion designated by the user. |
| SPERO Disposition | uint8 | Indicates whether the SPERO signature was used in file analysis. If the value is 1, 2, or 3, SPERO analysis was used. If there is any other value SPERO analysis was not used. |
| File Storage Status | uint8 | The storage status of the file. Possible values are:<br>• 1 — File Stored<br>• 2 — File Stored<br>• 3 — Unable to Store File<br>• 4 — Unable to Store File<br>• 5 — Unable to Store File<br>• 6 — Unable to Store File<br>• 7 — Unable to Store File<br>• 8 — File Size is Too Large<br>• 9 — File Size is Too Small<br>• 10 — Unable to Store File<br>• 11 — File Not Stored, Disposition Unavailable |

*Table B-52 File Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| File Analysis Status | uint8 | Indicates whether the file was sent for dynamic analysis. Possible values are:<br>• 0 — File Not Sent for Analysis<br>• 1 — Sent for Analysis<br>• 2 — Sent for Analysis<br>• 4 — Sent for Analysis<br>• 5 — Failed to Send<br>• 6 — Failed to Send<br>• 7 — Failed to Send<br>• 8 — Failed to Send<br>• 9 — File Size is Too Small<br>• 10 — File Size is Too Large<br>• 11 — Sent for Analysis<br>• 12 — Analysis Complete<br>• 13 — Failure (Network Issue)<br>• 14 — Failure (Rate Limit)<br>• 15 — Failure (File Too Large)<br>• 16 — Failure (File Read Error)<br>• 17 — Failure (Internal Library Error)<br>• 19 — File Not Sent, Disposition Unavailable<br>• 20 — Failure (Cannot Run File)<br>• 21 — Failure (Analysis Timeout)<br>• 22 — Sent for Analysis<br>• 23 —File Transmit File Capacity Handled — File capacity handled (stored on the sensor) because file could not be submitted to the sandbox for analysis<br>• 25 — File Transmit Server Limited Exceeded Capacity Handled — File capacity handled due to rate limiting on server<br>• 26 — Communication Failure — File capacity handled due to cloud connectivity failure<br>• 27 — Not Sent — File not sent due to configuration<br>• 28 — Preclass No Match —File not sent for dynamic analysis since pre-classification didn't find any embedded or suspicious object in the file<br>• 29 — Transmit Sent Sandbox Private Cloud — File sent to the private cloud for dynamic analysis<br>• 30 — Transmit Not Send Sendbox Private Cloud - File not send to the private cloud for analysis |

*Table B-52 File Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Local Malware Analysis Status | uint8 | The malware analysis status of the file. Possible values are:<br>• 0 — File not Analyzed<br>• 1 — Analysis Done<br>• 2 — Analysis Failed<br>• 3 — Manual Analysis Request |
| Archive File Status | uint8 | The status of an archive being inspected. Can have the following values:<br>• 0 — N/A — File is not being inspected as an archive<br>• 1 — Pending — Archive is being inspected<br>• 2 — Extracted — Successfully inspected without any problems<br>• 3 — Failed — Failed to inspect, insufficient system resources<br>• 4 — Depth Exceeded — Successful, but archive exceeded the nested inspection depth<br>• 5 — Encrypted — Partially Successful, Archive was or contains an archive that is encrypted<br>• 6 — Not Inspectable — Partially Successful, File is possibly Malformed or Corrupt |
| Threat Score | uint8 | A numeric value from 0 to 100 based on the potentially malicious behaviors observed during dynamic analysis. |
| Action | uint8 | The action taken on the file based on the file type. Can have the following values:<br>• 1 — Detect<br>• 2 — Block<br>• 3 — Malware Cloud Lookup<br>• 4 — Malware Block<br>• 5 — Malware Allow List<br>• 6 — Cloud Lookup Timeout<br>• 7 — Custom Detection<br>• 8 — Custom Detection Block<br>• 9 — Archive Block (Depth Exceeded)<br>• 10 — Archive Block (Encrypted)<br>• 11 — Archive Block (Failed to Inspect) |
| SHA Hash | uint8[32] | SHA-256 hash of the file, in binary format. |

*Table B-52 File Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| File Type ID | uint32 | ID number that maps to the file type. The meaning of this field is transmitted in the metadata with this event. See AMP for Endpoints File Type Metadata, page 3-38 for more information. |
| File Name | string | Name of the file. |
| File Size | uint64 | Size of the file in bytes. |
| Direction | uint8 | Value that indicates whether the file was uploaded or downloaded. Can have the following values:<br>• 1 — Download<br>• 2 — Upload<br>Currently the value depends on the protocol (for example, if the connection is HTTP it is a download). |
| Application ID | uint32 | ID number that maps to the application using the file transfer. |
| User ID | uint32 | ID number for the user logged into the destination host, as identified by the system. |
| URI | string | Uniform Resource Identifier (URI) of the connection. |
| Signature | string | SHA-256 hash of the file, in string format. |
| Source Port | uint16 | Port number for the source of the connection. |
| Destination Port | uint16 | Port number for the destination of the connection. |
| Protocol | uint8 | IANA protocol number specified by the user. For example:<br>• 1 — ICMP<br>• 4 — IP<br>• 6 — TCP<br>• 17 — UDP<br>This is currently only TCP. |
| Access Control Policy UUID | uint8[16] | Unique identifier for the access control policy that triggered the event. |
| Source Country | uint16 | Code for the country of the source host. |
| Destination Country | uint16 | Code for the country of the destination host. |
| Web Application ID | uint32 | The internal identification number for the web application, if applicable. |
| Client Application ID | uint32 | The internal identification number for the client application, if applicable. |
| Security Context | uint8(16) | ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode. |

*Table B-52 File Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Certificate Fingerprint | uint8[20] | SHA1 hash of the SSL Server certificate. |
| SSL Actual Action | uint16 | The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'Do Not Decrypt'<br>• 2 — 'Block'<br>• 3 — 'Block With Reset'<br>• 4 — 'Decrypt (Known Key)'<br>• 5 — 'Decrypt (Replace Key)'<br>• 6 — 'Decrypt (Resign)' |

*Table B-52 File Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| SSL Flow Status | uint16 | Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include:<br>• 0 — 'Unknown'<br>• 1 — 'No Match'<br>• 2 — 'Success'<br>• 3 — 'Uncached Session'<br>• 4 — 'Unknown Cipher Suite'<br>• 5 — 'Unsupported Cipher Suite'<br>• 6 — 'Unsupported SSL Version'<br>• 7 — 'SSL Compression Used'<br>• 8 — 'Session Undecryptable in Passive Mode'<br>• 9 — 'Handshake Error'<br>• 10 — 'Decryption Error'<br>• 11 — 'Pending Server Name Category Lookup'<br>• 12 — 'Pending Common Name Category Lookup'<br>• 13 — 'Internal Error'<br>• 14 — 'Network Parameters Unavailable'<br>• 15 — 'Invalid Server Certificate Handle'<br>• 16 — 'Server Certificate Fingerprint Unavailable'<br>• 17 — 'Cannot Cache Subject DN'<br>• 18 — 'Cannot Cache Issuer DN'<br>• 19 — 'Unknown SSL Version'<br>• 20 — 'External Certificate List Unavailable'<br>• 21 — 'External Certificate Fingerprint Unavailable'<br>• 22 — 'Internal Certificate List Invalid'<br>• 23 — 'Internal Certificate List Unavailable'<br>• 24 — 'Internal Certificate Unavailable'<br>• 25 — 'Internal Certificate Fingerprint Unavailable'<br>• 26 — 'Server Certificate Validation Unavailable'<br>• 27 — 'Server Certificate Validation Failure'<br>• 28 — 'Invalid Action' |
| String Block Type | uint32 | Initiates a String data block containing the Archive SHA. This value is always 0. |

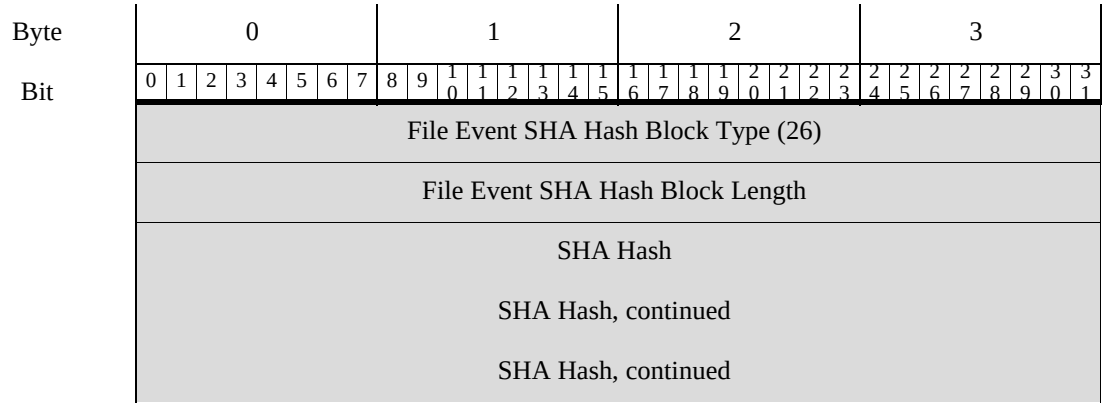
*Table B-52 File Event Data Block for 6.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Length | uint32 | The number of bytes included in the Archive SHA String data block, including eight bytes for the block type and header fields plus the number of bytes in the intrusion policy name. |
| Archive SHA | string | SHA1 hash of the parent archive in which the file is contained. |
| String Block Type | uint32 | Initiates a String data block containing the Archive Name. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the Archive Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the intrusion policy name. |
| Archive Name | string | Name of the parent archive. |
| Archive Depth | uint8 | Number of layers in which the file is nested. For example, if a text file is in a zip archive, this has a value of `1`. |
| HTTP Response Code | uint32 | HTTP Response Code |

# File Event SHA Hash for 5.1.1-5.2.x

The eStreamer service uses the File Event SHA Hash data block to contain metadata of the mapping of the SHA hash of a file to its filename. The block type is 26 in the series 2 list of data blocks. It can be requested if file log events have been requested in the extended requests—event code 111—and either bit 20 is set or metadata is requested with an event version of 4 and an event code of 21.

The following diagram shows the structure of a file event hash data block:



| | |
|---|---|
| | SHA Hash, continued |
| | SHA Hash, continued |
| | SHA Hash, continued |
| | SHA Hash, continued |
| | SHA Hash, continued |
| File Name | String Block Type (0) |
| | String Block Length |
| | File Name or Disposition... |

The following table describes the fields in the file event SHA hash data block.

***Table B-53 File Event SHA Hash 5.1.1-5.2.x Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| File Event SHA Hash Block Type | uint32 | Initiates a File Event SHA Hash block. This value is always `26`. |
| File Event SHA Hash Block Length | uint32 | Total number of bytes in the File Event SHA Hash block, including eight bytes for the File Event SHA Hash block type and length fields, plus the number of bytes of data that follows. |
| SHA Hash | uint8[32] | The SHA-256 hash of the file in binary format. |
| String Block Type | uint32 | Initiates a String data block containing the descriptive name associated with the file. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Name field. |
| File Name or Disposition | string | The descriptive name or disposition of the file. If the file is clean, this value is `Clean`. If the file's disposition is unknown, the value is `Neutral`. If the file contains malware, the file name is given. |

# Legacy Correlation Event Data Structures

The following topics describe other legacy correlation (compliance) data structures:

- Correlation Event for 5.0 - 5.0.2, page B-332
- Correlation Event for 5.1-5.3.x, page B-339

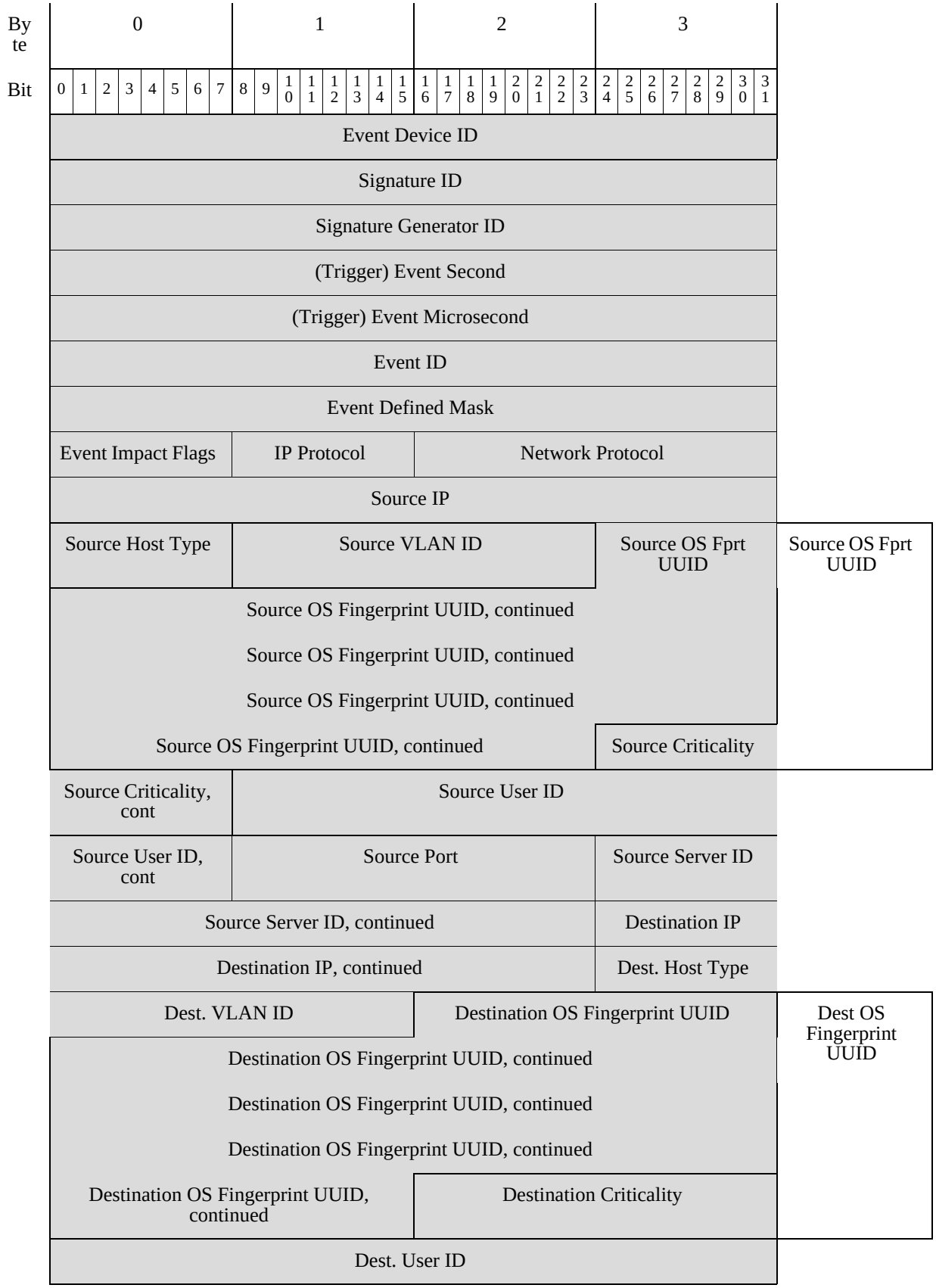
# Correlation Event for 5.0 - 5.0.2

Correlation events (called compliance events in pre-5.0 versions) contain information about correlation policy violations. This message uses the standard eStreamer message header and specifies a record type of 112, followed by a correlation data block of type 116. Data block type 116 differs from its predecessor (block type 107) in including additional information about the associated security zone and interface.

You can request 5.0 correlation events from eStreamer only by extended request, for which you request event type code 31 and version code 7 in the Stream Request message (see Submitting Extended Requests, page 2-4 for information about submitting extended requests). You can optionally enable bit 23 in the flags field of the initial event stream request message, to include the extended event header. You can also enable bit 20 in the flags field to include user metadata.

Note that the record structure includes a String block type, which is a block in series 1. For information about series 1 blocks, see Understanding Discovery (Series 1) Blocks, page 4-62.

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Header Version (1) | | | | | | | | | | | | | | | | | Message Type (4) | | | | | | | | | | | | | | | |
| Message Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Netmap ID | | | | | | | | | | | | | | | | | Record Type (112) | | | | | | | | | | | | | | | |
| Record Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| eStreamer Server Timestamp (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reserved for Future Use (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Correlation Block Type (116) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Correlation Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (Correlation) Event Second | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Priority | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Description... | | | | | | | | | | | | | | | | | | | | | | | | | Event Type | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Event Device ID
Signature ID
Signature Generator ID
(Trigger) Event Second
(Trigger) Event Microsecond
Event ID
Event Defined Mask
Event Impact Flags
IP Protocol
Network Protocol
Source IP
Source Host Type
Source VLAN ID
Source OS Fprt UUID
Source OS Fprt UUID
Source OS Fingerprint UUID, continued
Source OS Fingerprint UUID, continued
Source OS Fingerprint UUID, continued
Source OS Fingerprint UUID, continued
Source Criticality
Source Criticality, cont
Source User ID
Source User ID, cont
Source Port
Source Server ID
Source Server ID, continued
Destination IP
Destination IP, continued
Dest. Host Type
Dest. VLAN ID
Destination OS Fingerprint UUID
Dest OS Fingerprint UUID
Destination OS Fingerprint UUID, continued
Destination OS Fingerprint UUID, continued
Destination OS Fingerprint UUID, continued
Destination OS Fingerprint UUID, continued
Destination Criticality
Dest. User ID

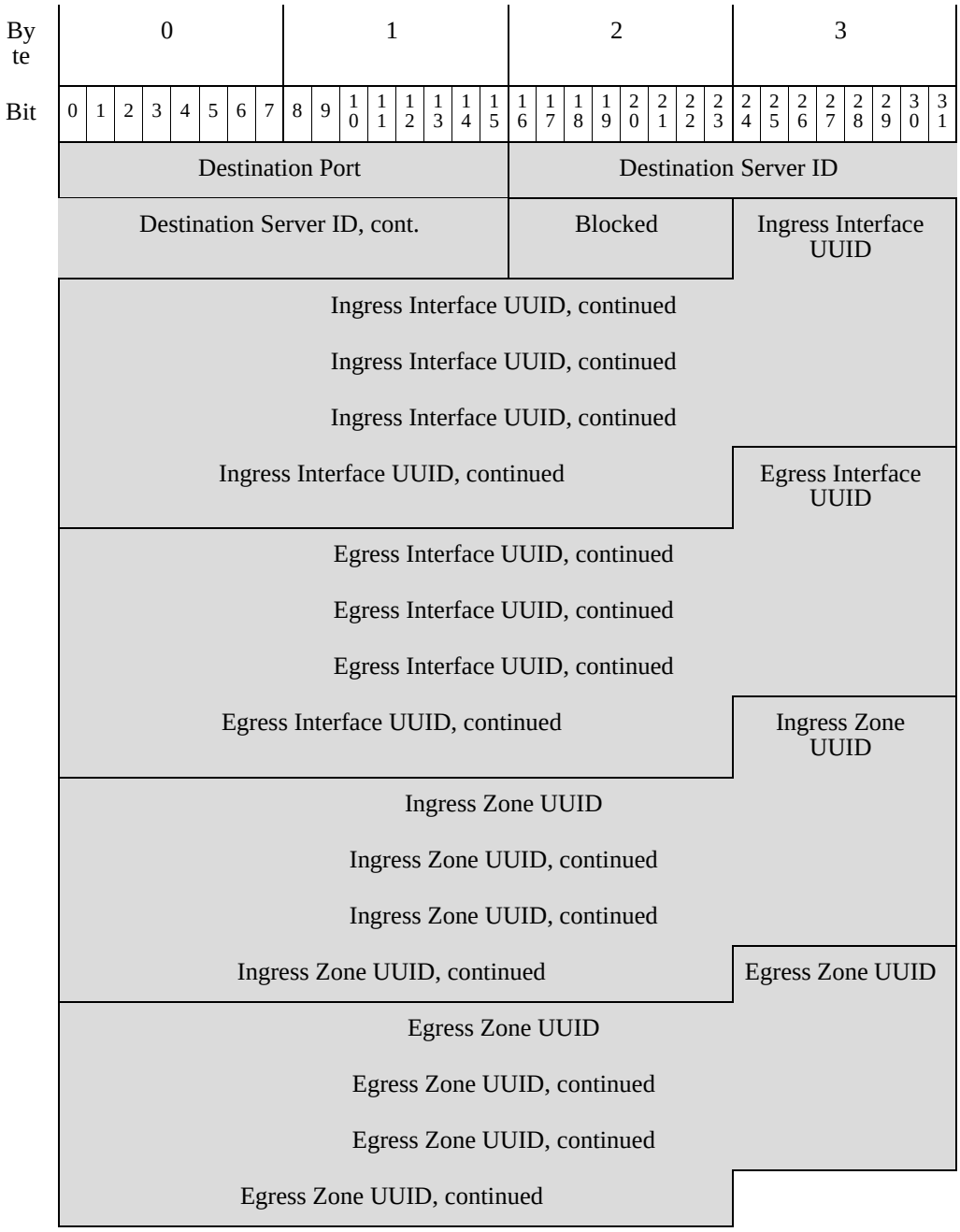



***Table B-54 Correlation Event 5.0 - 5.0.2 Data Fields***

| Field | Data Type | Description |
|---|---|---|
| Correlation Block Type | uint32 | Indicates a correlation event data block follows. This field always has a value of `107`. See Understanding Discovery (Series 1) Blocks, page 4-62. |
| Correlation Block Length | uint32 | Length of the correlation data block, which includes 8 bytes for the correlation block type and length plus the correlation data that follows. |

*Table B-54 Correlation Event 5.0 - 5.0.2 Data Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Device ID | uint32 | Internal identification number of the managed device or Defense Center that generated the correlation event. A value of zero indicates the Defense Center. You can obtain managed device names by requesting Version 3 metadata. See Managed Device Record Metadata, page 3-33 for more information. |
| (Correlation) Event Second | uint32 | UNIX timestamp indicating the time that the correlation event was generated (in seconds from 01/01/1970). |
| Event ID | uint32 | Correlation event identification number. |
| Policy ID | uint32 | Identification number of the correlation policy that was violated. See Service Record, page 4-15 for information about how to obtain policy identification numbers from the database. |
| Rule ID | uint32 | Identification number of the correlation rule that triggered to violate the policy. See Service Record, page 4-15 for information about how to obtain policy identification numbers from the database. |
| Priority | uint32 | Priority assigned to the event. This is an integer value from `0` to `5`. |
| String Block Type | uint32 | Initiates a string data block that contains the correlation violation event description. This value is always set to `0`. For more information about string blocks, see String Data Block, page 4-71. |
| String Block Length | uint32 | Number of bytes in the event description string block, which includes four bytes for the string block type and four bytes for the string block length, plus the number of bytes in the description. |
| Description | string | Description of the correlation event. |
| Event Type | uint8 | Indicates whether the correlation event was triggered by an intrusion, host discovery, or user event:<br>• `1` — Intrusion<br>• `2` — Host discovery<br>• `3` — User |
| Event Device ID | uint32 | Identification number of the device that generated the event that triggered the correlation event. You can obtain device name by requesting Version 3 metadata. See Managed Device Record Metadata, page 3-33 for more information. |
| Signature ID | uint32 | If the event was an intrusion event, indicates the rule identification number that corresponds with the event. Otherwise, the value is `0`. |
| Signature Generator ID | uint32 | If the event was an intrusion event, indicates the ID number of the Secure Firewall System preprocessor or rules engine that generated the event. |
| (Trigger) Event Second | uint32 | UNIX timestamp indicating the time of the event that triggered the correlation policy rule (in seconds from 01/01/1970). |
| (Trigger) Event Microsecond | uint32 | Microsecond (one millionth of a second) increment that the event was detected. |
| Event ID | uint32 | Identification number of the event generated by the device. |

*Table B-54 Correlation Event 5.0 - 5.0.2 Data Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Event Defined Mask | bits[32] | Set bits in this field indicate which of the fields that follow in the message are valid. See Table B-55 on page B-338 for a list of each bit value. |
| Event Impact Flags | bits[8] | Impact flag value of the event. The low-order eight bits indicate the impact level. Values are:<br>• `0x01` (bit 0) — Source or destination host is in a network monitored by the system.<br>• `0x02` (bit 1) — Source or destination host exists in the network map.<br>• `0x04` (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol.<br>• `0x08` (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event.<br>• `0x10` (bit 4) — There is a vulnerability mapped to the server detected in the event.<br>• `0x20` (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface.<br>• `0x40` (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red (bit 6). The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software.<br>• `0x80` (bit 7) — There is a vulnerability mapped to the client detected in the event.<br>The following impact level values map to specific priorities on the Defense Center. An `x` indicates the value can be 0 or 1:<br>• (0, unknown): `00X00000`<br>• red (1, vulnerable): `XXXX1XXX, XXX1XXXX, X1XXXXXX, 1XXXXXXX`<br>• orange (2, potentially vulnerable): `00X00111`<br>• yellow (3, currently not vulnerable): `00X00011`<br>• blue (4, unknown target): `00X00001` |
| IP Protocol | uint8 | Identifier of the IP protocol associated with the event, if applicable. |
| Network Protocol | uint16 | Network protocol associated with the event, if applicable. |
| Source IP | uint8[4] | IP address of the source host in the event, in IP address octets. |
| Source Host Type | uint8 | Source host's type:<br>• `0` — Host<br>• `1` — Router<br>• `2` — Bridge |

*Table B-54 Correlation Event 5.0 - 5.0.2 Data Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Source VLAN ID | uint16 | Source host's VLAN identification number, if applicable. |
| Source OS Fingerprint UUID | uint8[16] | A fingerprint ID number that acts a unique identifier for the source host's operating system.<br>See Service Record, page 4-15 for information about obtaining the values that map to the fingerprint IDs. |
| Source Criticality | uint16 | User-defined criticality value for the source host:<br>• 0 — None<br>• 1 — Low<br>• 2 — Medium<br>• 3 — High |
| Source User ID | uint32 | Identification number for the user logged into the source host, as identified by the system. |
| Source Port | uint16 | Source port in the event. |
| Source Server ID | uint32 | Identification number for the server running on the source host. |
| Destination IP Address | uint8[4] | IP address of the destination host associated with the policy violation (if applicable). This value will be 0 if there is no destination IP address. |
| Destination Host Type | uint8 | Destination host's type:<br>• 0 — Host<br>• 1 — Router<br>• 2 — Bridge |
| Destination VLAN ID | uint16 | Destination host's VLAN identification number, if applicable. |
| Destination OS Fingerprint UUID | uint8[16] | A fingerprint ID number that acts as a unique identifier for the destination host's operating system.<br>See Service Record, page 4-15 for information about obtaining the values that map to the fingerprint IDs. |
| Destination Criticality | uint16 | User-defined criticality value for the destination host:<br>• 0 — None<br>• 1 — Low<br>• 2 — Medium<br>• 3 — High |
| Destination User ID | uint32 | Identification number for the user logged into the destination host, as identified by the system. |
| Destination Port | uint16 | Destination port in the event. |
| Destination Service ID | uint32 | Identification number for the server running on the source host. |

*Table B-54 Correlation Event 5.0 - 5.0.2 Data Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Blocked | uint8 | Value indicating what happened to the packet that triggered the intrusion event.<br>• 0 — Intrusion event not dropped<br>• 1 — Intrusion event was dropped (drop when deployment is inline, switched, or routed)<br>• 2 — The packet that triggered the event would have been dropped, if the intrusion policy had been applied to a device in inline, switched, or routed deployment. |
| Ingress Interface UUID | uint8[16] | An interface ID that acts as the unique identifier for the ingress interface associated with correlation event. |
| Egress Interface UUID | uint8[16] | An interface ID that acts as the unique identifier for the egress interface associated with correlation event. |
| Ingress Zone UUID | uint8[16] | A zone ID that acts as the unique identifier for the ingress security zone associated with correlation event. |
| Egress Zone UUID | uint8[16] | A zone ID that acts as the unique identifier for the egress security zone associated with correlation event. |

The following table describes each Event Defined Mask value.

*Table B-55 Event Defined Values*

| Description | Mask Value |
|---|---|
| Event Impact Flags | 0x00000001 |
| IP Protocol | 0x00000002 |
| Network Protocol | 0x00000004 |
| Source IP | 0x00000008 |
| Source Host Type | 0x00000010 |
| Source VLAN ID | 0x00000020 |
| Source Fingerprint ID | 0x00000040 |
| Source Criticality | 0x00000080 |
| Source Port | 0x00000100 |
| Source Server | 0x00000200 |
| Destination IP | 0x00000400 |
| Destination Host Type | 0x00000800 |
| Destination VLAN ID | 0x00001000 |
| Destination Fingerprint ID | 0x00002000 |
| Destination Criticality | 0x00004000 |
| Destination Port | 0x00008000 |
| Destination Server | 0x00010000 |

*Table B-55 Event Defined Values (continued)*

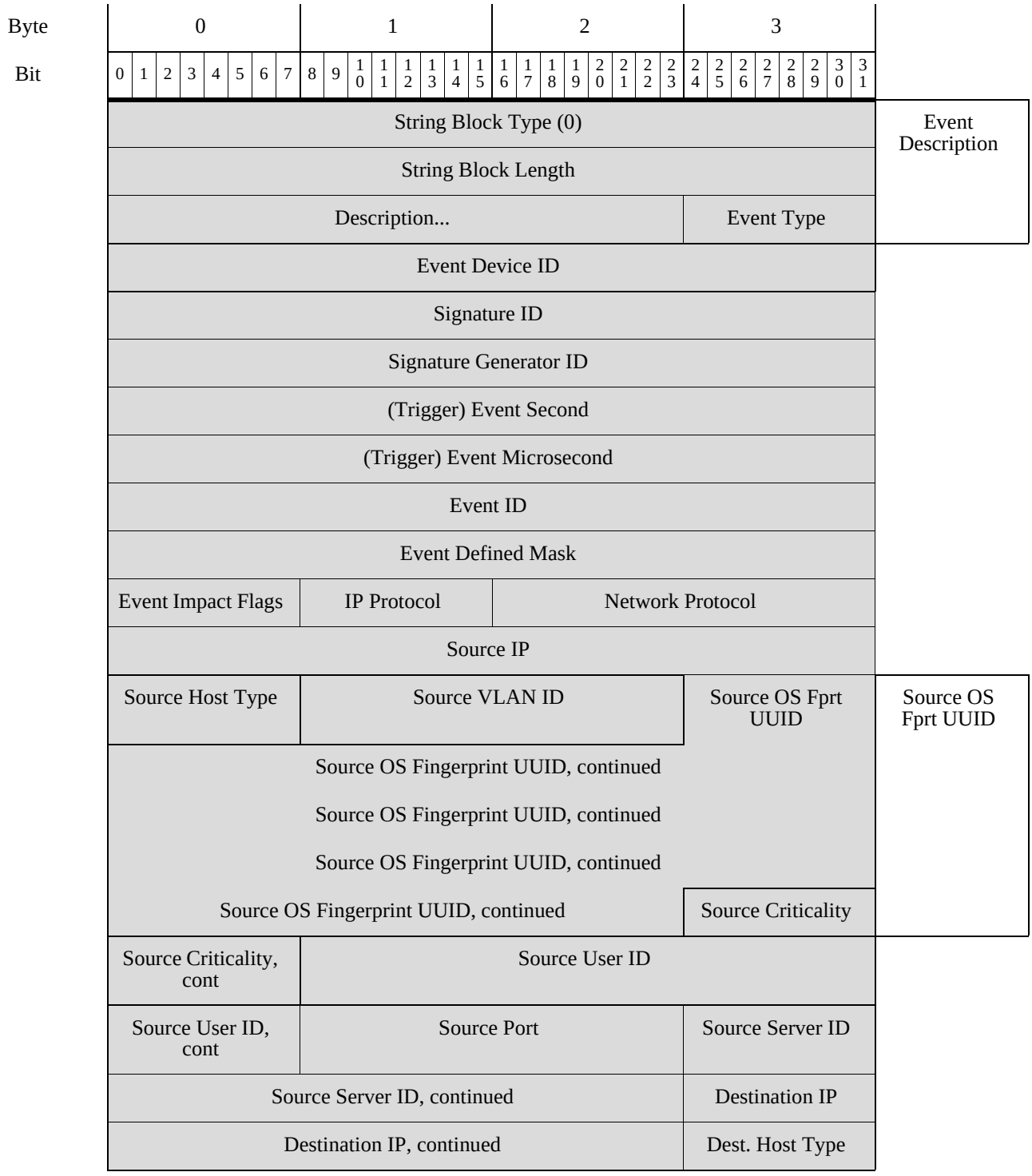
| Description | Mask Value |
|---|---|
| Source User | 0x00020000 |
| Destination User | 0x00040000 |

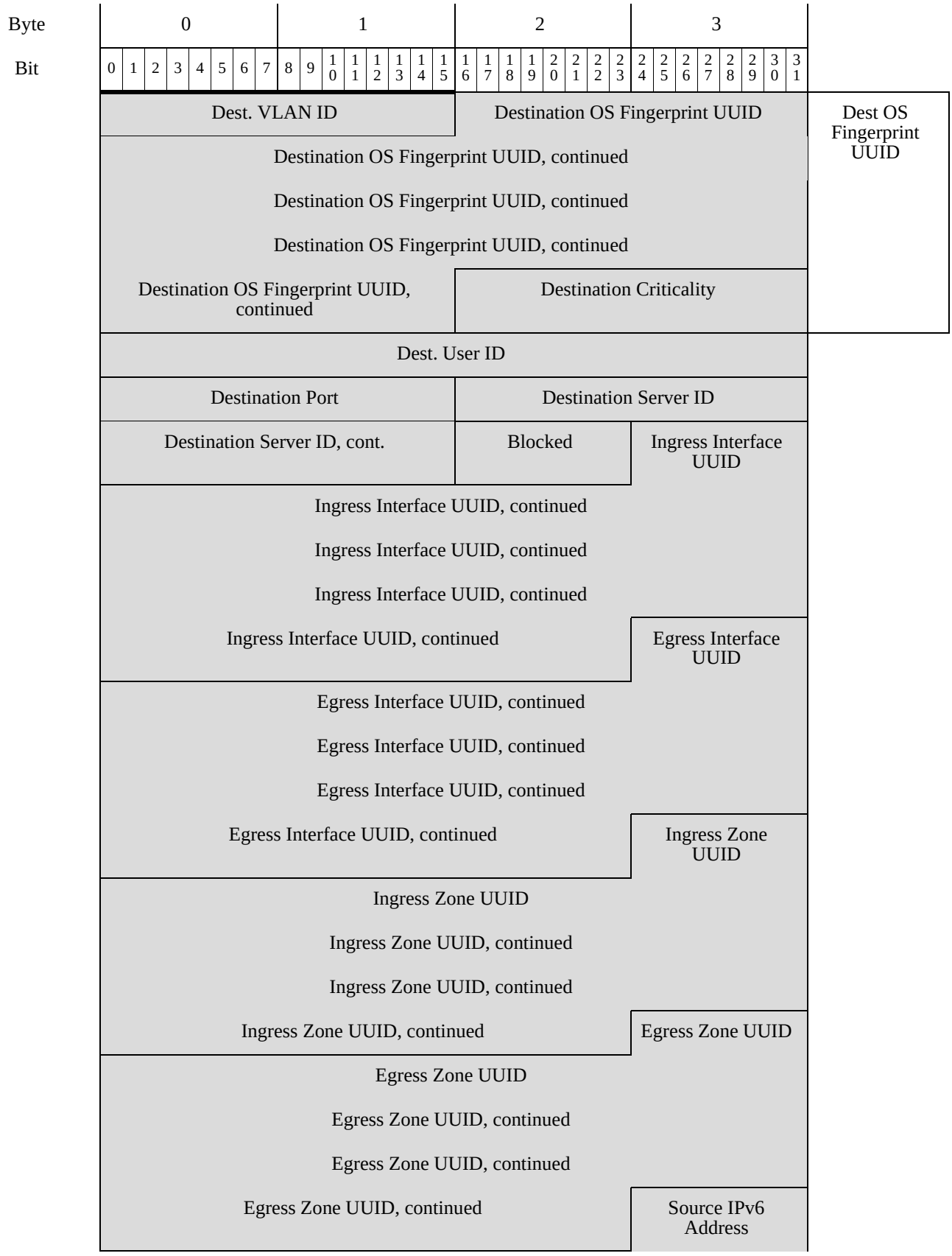
# Correlation Event for 5.1-5.3.x

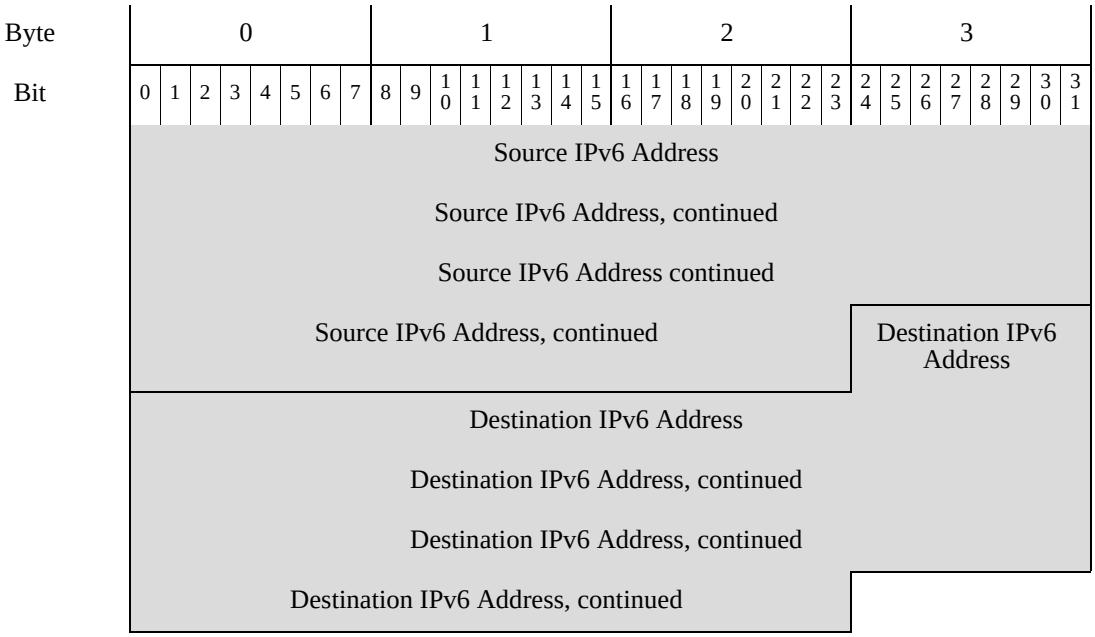
Correlation events (called compliance events in pre-5.0 versions) contain information about correlation policy violations. This message uses the standard eStreamer message header and specifies a record type of 112, followed by a correlation data block of type 128 in the series 1 set of data blocks. Data block type 128 differs from its predecessor (block type 116) in including IPv6 support.

You can request 5.1-5.3.x correlation events from eStreamer only by extended request, for which you request event type code 31 and version code 8 in the Stream Request message (see Submitting Extended Requests, page 2-4 for information about submitting extended requests). You can optionally enable bit 23 in the flags field of the initial event stream request message, to include the extended event header. You can also enable bit 20 in the flags field to include user metadata.

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Header Version (1) | | | | | | | | | | | | | | | | | Message Type (4) | | | | | | | | | | | | | | | |
| Message Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Netmap ID | | | | | | | | | | | | | | | | | Record Type (112) | | | | | | | | | | | | | | | |
| Record Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| eStreamer Server Timestamp (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reserved for Future Use (in events, only if bit 23 is set) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Correlation Block Type (128) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Correlation Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Device ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (Correlation) Event Second | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Event ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policy ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Rule ID | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Priority | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
String Block Type (0)
String Block Length
Description...
Event Type
Event Description
Event Device ID
Signature ID
Signature Generator ID
(Trigger) Event Second
(Trigger) Event Microsecond
Event ID
Event Defined Mask
Event Impact Flags
IP Protocol
Network Protocol
Source IP
Source Host Type
Source VLAN ID
Source OS Fprt UUID
Source OS Fprt UUID
Source OS Fingerprint UUID, continued
Source OS Fingerprint UUID, continued
Source OS Fingerprint UUID, continued
Source OS Fingerprint UUID, continued
Source Criticality
Source Criticality, cont
Source User ID
Source User ID, cont
Source Port
Source Server ID
Source Server ID, continued
Destination IP
Destination IP, continued
Dest. Host Type


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Dest. VLAN ID
Destination OS Fingerprint UUID
Dest OS Fingerprint UUID
Destination OS Fingerprint UUID, continued
Destination OS Fingerprint UUID, continued
Destination OS Fingerprint UUID, continued
Destination OS Fingerprint UUID, continued
Destination Criticality
Dest. User ID
Destination Port
Destination Server ID
Destination Server ID, cont.
Blocked
Ingress Interface UUID
Ingress Interface UUID, continued
Ingress Interface UUID, continued
Ingress Interface UUID, continued
Ingress Interface UUID, continued
Egress Interface UUID
Egress Interface UUID, continued
Egress Interface UUID, continued
Egress Interface UUID, continued
Egress Interface UUID, continued
Ingress Zone UUID
Ingress Zone UUID
Ingress Zone UUID, continued
Ingress Zone UUID, continued
Ingress Zone UUID, continued
Egress Zone UUID
Egress Zone UUID
Egress Zone UUID, continued
Egress Zone UUID, continued
Egress Zone UUID, continued
Source IPv6 Address




Note that the record structure includes a String block type, which is a block in series 1. For information about series 1 blocks, see Understanding Discovery (Series 1) Blocks, page 4-62.

***Table B-56 Correlation Event 5.1-5.3.x Data Fields***

| Field | Data Type | Description |
|---|---|---|
| Correlation Block Type | uint32 | Indicates a correlation event data block follows. This field always has a value of 128. See Understanding Discovery (Series 1) Blocks, page 4-62. |
| Correlation Block Length | uint32 | Length of the correlation data block, which includes 8 bytes for the correlation block type and length plus the correlation data that follows. |
| Device ID | uint32 | Internal identification number of the managed device or Defense Center that generated the correlation event. A value of zero indicates the Defense Center. You can obtain managed device names by requesting Version 3 metadata. See Managed Device Record Metadata, page 3-33 for more information. |
| (Correlation) Event Second | uint32 | UNIX timestamp indicating the time that the correlation event was generated (in seconds from 01/01/1970). |
| Event ID | uint32 | Correlation event identification number. |
| Policy ID | uint32 | Identification number of the correlation policy that was violated. See Service Record, page 4-15 for information about how to obtain policy identification numbers from the database. |
| Rule ID | uint32 | Identification number of the correlation rule that triggered to violate the policy. See Service Record, page 4-15 for information about how to obtain policy identification numbers from the database. |
| Priority | uint32 | Priority assigned to the event. This is an integer value from 0 to 5. |

*Table B-56 Correlation Event 5.1-5.3.x Data Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| String Block Type | uint32 | Initiates a string data block that contains the correlation violation event description. This value is always set to 0. For more information about string blocks, see String Data Block, page 4-71. |
| String Block Length | uint32 | Number of bytes in the event description string block, which includes four bytes for the string block type and four bytes for the string block length, plus the number of bytes in the description. |
| Description | string | Description of the correlation event. |
| Event Type | uint8 | Indicates whether the correlation event was triggered by an intrusion, host discovery, or user event:<br>• 1 — Intrusion<br>• 2 — Host discovery<br>• 3 — User |
| Event Device ID | uint32 | Identification number of the device that generated the event that triggered the correlation event. You can obtain device name by requesting Version 3 metadata. See Managed Device Record Metadata, page 3-33 for more information. |
| Signature ID | uint32 | If the event was an intrusion event, indicates the rule identification number that corresponds with the event. Otherwise, the value is 0. |
| Signature Generator ID | uint32 | If the event was an intrusion event, indicates the ID number of the Secure Firewall System preprocessor or rules engine that generated the event. |
| (Trigger) Event Second | uint32 | UNIX timestamp indicating the time of the event that triggered the correlation policy rule (in seconds from 01/01/1970). |
| (Trigger) Event Microsecond | uint32 | Microsecond (one millionth of a second) increment that the event was detected. |
| Event ID | uint32 | Identification number of the event generated by the Cisco device. |
| Event Defined Mask | bits[32] | Set bits in this field indicate which of the fields that follow in the message are valid. See Table B-55 on page B-338 for a list of each bit value. |

*Table B-56 Correlation Event 5.1-5.3.x Data Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Event Impact Flags | bits[8] | Impact flag value of the event. The low-order eight bits indicate the impact level. Values are:<br>• 0x01 (bit 0) — Source or destination host is in a network monitored by the system.<br>• 0x02 (bit 1) — Source or destination host exists in the network map.<br>• 0x04 (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol.<br>• 0x08 (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event.<br>• 0x10 (bit 4) — There is a vulnerability mapped to the server detected in the event.<br>• 0x20 (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface.<br>• 0x40 (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software.<br>• 0x80 (bit 7) — There is a vulnerability mapped to the client detected in the event. (version 5.0+ only)<br>The following impact level values map to specific priorities on the Defense Center. An `x` indicates the value can be 0 or 1:<br>• (0, unknown): `00X00000`<br>• red (1, vulnerable): `XXXX1XXX, XXX1XXXX, X1XXXXXX, 1XXXXXXX` (version 5.0+ only)<br>• orange (2, potentially vulnerable): `00X0011X`<br>• yellow (3, currently not vulnerable): `00X0001X`<br>• blue (4, unknown target): `00X00001` |
| IP Protocol | uint8 | Identifier of the IP protocol associated with the event, if applicable. |
| Network Protocol | uint16 | Network protocol associated with the event, if applicable. |
| Source IP Address | uint8[4] | This field is reserved but no longer populated. The Source IPv4 address is stored in the Source IPv6 Address field. See IP Addresses, page 1-4 for more information. |
| Source Host Type | uint8 | Source host's type:<br>• 0 — Host<br>• 1 — Router<br>• 2 — Bridge |

*Table B-56 Correlation Event 5.1-5.3.x Data Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Source VLAN ID | uint16 | Source host's VLAN identification number, if applicable. |
| Source OS Fingerprint UUID | uint8[16] | A fingerprint ID number that acts a unique identifier for the source host's operating system.<br>See Service Record, page 4-15 for information about obtaining the values that map to the fingerprint IDs. |
| Source Criticality | uint16 | User-defined criticality value for the source host:<br>• 0 — None<br>• 1 — Low<br>• 2 — Medium<br>• 3 — High |
| Source User ID | uint32 | Identification number for the user logged into the source host, as identified by the system. |
| Source Port | uint16 | Source port in the event. |
| Source Server ID | uint32 | Identification number for the server running on the source host. |
| Destination IP Address | uint8[4] | This field is reserved but no longer populated. The Destination IPv4 address is stored in the Destination IPv6 Address field. See IP Addresses, page 1-4 for more information. |
| Destination Host Type | uint8 | Destination host's type:<br>• 0 — Host<br>• 1 — Router<br>• 2 — Bridge |
| Destination VLAN ID | uint16 | Destination host's VLAN identification number, if applicable. |
| Destination OS Fingerprint UUID | uint8[16] | A fingerprint ID number that acts as a unique identifier for the destination host's operating system.<br>See Service Record, page 4-15 for information about obtaining the values that map to the fingerprint IDs. |
| Destination Criticality | uint16 | User-defined criticality value for the destination host:<br>• 0 — None<br>• 1 — Low<br>• 2 — Medium<br>• 3 — High |
| Destination User ID | uint32 | Identification number for the user logged into the destination host, as identified by the system. |
| Destination Port | uint16 | Destination port in the event. |
| Destination Service ID | uint32 | Identification number for the server running on the source host. |

*Table B-56 Correlation Event 5.1-5.3.x Data Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Blocked | uint8 | Value indicating what happened to the packet that triggered the intrusion event.<br>• 0 — Intrusion event not dropped<br>• 1 — Intrusion event was dropped (drop when deployment is inline, switched, or routed)<br>• 2 — The packet that triggered the event would have been dropped, if the intrusion policy had been applied to a device in inline, switched, or routed deployment. |
| Ingress Interface UUID | uint8[16] | An interface ID that acts as the unique identifier for the ingress interface associated with correlation event. |
| Egress Interface UUID | uint8[16] | An interface ID that acts as the unique identifier for the egress interface associated with correlation event. |
| Ingress Zone UUID | uint8[16] | A zone ID that acts as the unique identifier for the ingress security zone associated with correlation event. |
| Egress Zone UUID | uint8[16] | A zone ID that acts as the unique identifier for the egress security zone associated with correlation event. |
| Source IPv6 Address | uint8[16] | IP address of the source host in the event, in IPv6 address octets. |
| Destination IPv6 Address | uint8[16] | IP address of the destination host in the event, in IPv6 address octets. |

# Legacy Host Data Structures

To request these structures, you must use a Host Request Message. To request a legacy structure, the Host Request Message must use an older format. See Host Request Message Format, page 2-27 for more information.

The following topics describe legacy host data structures, including both host profile and full host profile structures:


- Full Host Profile Data Block 5.0 - 5.0.2, page B-347
- Full Host Profile Data Block 5.1.1, page B-356
- Full Host Profile Data Block 5.2.x, page B-364
- Host Profile Data Block for 5.1.x, page B-376
- IP Range Specification Data Block for 5.0 - 5.1.1.x, page B-382
- Access Control Policy Rule Reason Data Block, page B-382

# Full Host Profile Data Block 5.0 - 5.0.2

The Full Host Profile data block for version 5.0 - 5.0.2 contains a full set of data describing one host. It has the format shown in the graphic below and explained in the following table. Note that, except for List data blocks, the graphic does not show the fields of the encapsulated data blocks. These encapsulated data blocks are described separately in Understanding Discovery & Connection Data Structures, page 4-1. The Full Host Profile data block a block type value of 111.

**Note** An asterisk(*) next to a block name in the following diagram indicates that multiple instances of the data block may occur.

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Full Host Profile Data Block (111) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Data Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | IP Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Hops | | | | | | | | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type, continued | | | | | | | | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | |
| OS Derived Fingerprints | Generic List Block Length, continued | | | | | | | | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | |
| | OS Fingerprint Block Type (130)*, con't | | | | | | | | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | |
| | OS Fingerprint Block Length, con't | | | | | | | | Operating System Derived Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Server Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Server Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Client Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Client Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VDB Native Fingerprints 1 | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System VDB Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VDB Native Fingerprints 2 | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System VDB Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System User Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Scan Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Scan Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Application Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Application Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Conflict Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Conflict Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (TCP) Full Server Data | List Block Type (11)... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (TCP) Full Server Data Blocks (104)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (UDP) Full Server Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (UDP) Full Server Data Blocks (104)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Network Protocol Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (Network) Protocol Data Blocks (4)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transport Protocol Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (Transport) Protocol Data Blocks (4)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| MAC Address Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Host MAC Address Data Blocks (95)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Last Seen | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Host Type | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Business Criticality | | | | | | | | | | | | | | | | VLAN ID | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | VLAN Type | | | | | | | | VLAN Priority | | | | | | | | Generic List Block Type (31) | | | | | | | | | | | | | | | |
| Host Client Data | Generic List Block Type, continued | | | | | | | | | | | | | | | | Generic List Block Length | | | | | | | | | | | | | | | |
| | Generic List Block Length, continued | | | | | | | | | | | | | | | | Full Host Client Application Data Blocks (112)* | | | | | | | | | | | | | | | |
| NetBIOS Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | NetBIOS Name String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Notes Data | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Notes String.... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (VDB) Host Vulns | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (VDB) Host Vulnerability Data Blocks (85)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3rd Pty/VDB) Host Vulns | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (Third Party/VDB) Host Vulnerability Data Blocks (85)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3rd Pty Scan Host Vulns | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (Third Party Scan) Host Vulnerability Data Blocks with Original Vuln IDs (85)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Attribute Value Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Attribute Value Data Blocks * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

The following table describes the components of the Full Host Profile for 5.0 - 5.0.2 record.

***Table B-57 Full Host Profile Record 5.0 - 5.0.2 Fields***

| Field | Data Type | Description |
|---|---|---|
| IP Address | uint8[4] | IP address of the host, in IP address octets. |
| Hops | uint8 | Number of network hops from the host to the device. |

*Table B-57 Full Host Profile Record 5.0 - 5.0.2 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data derived from the existing fingerprints for the host. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Derived Fingerprint Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host derived from the existing fingerprints for the host. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a server fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Server Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a server fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a client fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Client Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a client fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a Cisco VDB fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |

*Table B-57 Full Host Profile Record 5.0 - 5.0.2 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Operating System Fingerprint (VDB) Native Fingerprint 1) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using the fingerprints in the Cisco vulnerability database (VDB). See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a Cisco VDB fingerprint. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (VDB) Native Fingerprint 2) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using the fingerprints in the Cisco vulnerability database (VDB). See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data added by a user. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (User Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host added by a user. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data added by a vulnerability scanner. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Scan Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host added by a vulnerability scanner. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data added by an application. This value is always `31`. |

*Table B-57 Full Host Profile Record 5.0 - 5.0.2 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Application Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host added by an application. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data selected through fingerprint conflict resolution. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Conflict Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host selected through fingerprint conflict resolution. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Full Server data blocks conveying TCP service data. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus the length of all encapsulated Full Server data blocks. |
| (TCP) Full Server Data Blocks * | variable | List of Full Server data blocks conveying data about the TCP services on the host. See Full Host Server Data Block 4.10.0+, page 4-141 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Full Server data blocks conveying UDP service data. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus the length of all encapsulated Full Server data blocks. |
| (UDP) Full Server Data Blocks * | variable | List of Full Server data blocks conveying data about the UDP sub-servers on the host. See Full Host Server Data Block 4.10.0+, page 4-141 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Protocol data blocks conveying network protocol data. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus the length of all encapsulated Protocol data blocks. |
| (Network) Protocol Data Blocks * | variable | List of Protocol data blocks conveying data about the network protocols on the host. See Protocol Data Block, page 4-75 for a description of this data block. |

*Table B-57 Full Host Profile Record 5.0 - 5.0.2 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| List Block Type | uint32 | Initiates a List data block comprising Protocol data blocks conveying transport protocol data. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus the length of all encapsulated Protocol data blocks. |
| (Transport) Protocol Data Blocks * | variable | List of Protocol data blocks conveying data about the transport protocols on the host. See Protocol Data Block, page 4-75 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block containing Host MAC Address data blocks. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the list, including the list header and all encapsulated Host MAC Address data blocks. |
| Host MAC Address Data Blocks * | variable | List of Host MAC Address data blocks. See Host MAC Address 4.9+, page 4-115 for a description of this data block. |
| Last Seen | uint32 | UNIX timestamp that represents the last time the system detected host activity. |
| Host Type | uint32 | Indicates host type. Values include:<br>• `0` — Host<br>• `1` — Router<br>• `2` — Bridge<br>• `3` — NAT (network address translation device)<br>• `4` — LB (load balancer) |
| Business Criticality | uint16 | Indicates criticality of host to business. |
| VLAN ID | uint16 | VLAN identification number that indicates which VLAN the host is a member of. |
| VLAN Type | uint8 | Type of packet encapsulated in the VLAN tag. |
| VLAN Priority | uint8 | Priority value included in the VLAN tag. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Host Vulnerability data blocks conveying Client Application data. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Client Application data blocks. |
| Full Host Client Application Data Blocks * | variable | List of Client Application data blocks. See Full Host Client Application Data Block 5.0+, page 4-155 for a description of this data block. |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |

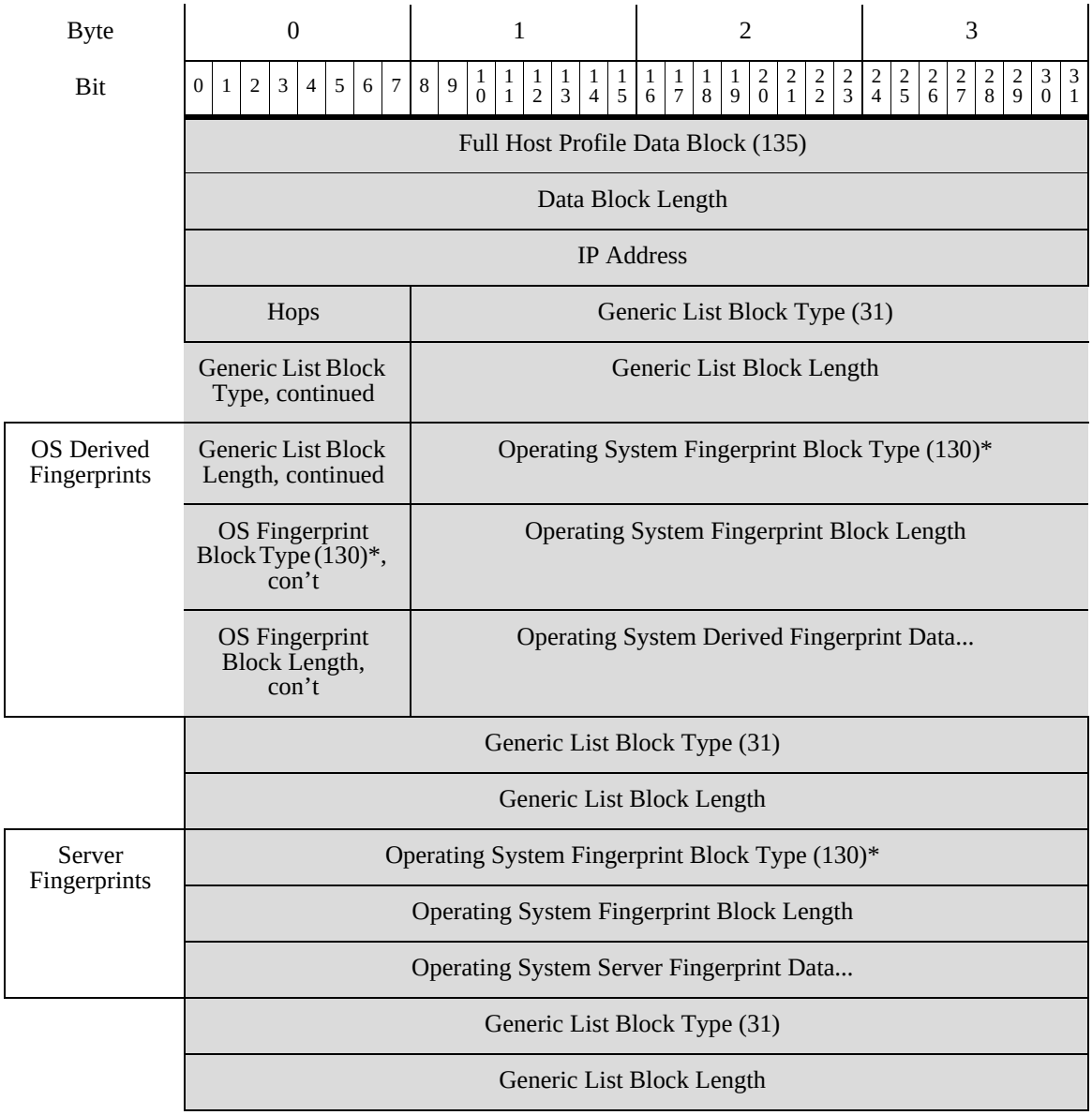
*Table B-57 Full Host Profile Record 5.0 - 5.0.2 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for host notes. This value is always `0`. |
| String Block Length | uint32 | Number of bytes in the notes String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the notes string. |
| Notes | string | Contains the contents of the Notes host attribute for the host. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Host Vulnerability data blocks conveying VDB vulnerability data. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated data blocks. |
| (VDB) Host Vulnerability Data Blocks * | variable | List of Host Vulnerability data blocks for vulnerabilities identified in the Cisco vulnerability database (VDB). See Host Vulnerability Data Block 4.9.0+, page 4-112 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Host Vulnerability data blocks conveying third-party scan vulnerability data. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated data blocks. |
| (Third Party/VDB) Host Vulnerability Data Blocks * | variable | Host Vulnerability data blocks sourced from a third party scanner and containing information about host vulnerabilities cataloged in the Cisco vulnerability database (VDB). See Host Vulnerability Data Block 4.9.0+, page 4-112 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Host Vulnerability data blocks conveying third party scan vulnerability data. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated data blocks. |
| (Third Party Scan) Host Vulnerability Data Blocks * | variable | Host Vulnerability data blocks sourced from a third party scanner. Note that the host vulnerability IDs for these data blocks are the third party scanner IDs, not Cisco-detected IDs. See Host Vulnerability Data Block 4.9.0+, page 4-112 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Attribute Value data blocks conveying attribute data. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the List data block, including the list header and all encapsulated data blocks. |
| Attribute Value Data Blocks * | variable | List of Attribute Value data blocks. See Attribute Value Data Block, page 4-82 for a description of the data blocks in this list. |

# Full Host Profile Data Block 5.1.1

The Full Host Profile data block for version 5.1.1 contains a full set of data describing one host. It has the format shown in the graphic below and explained in the following table. Note that, except for List data blocks, the graphic does not show the fields of the encapsulated data blocks. These encapsulated data blocks are described separately in Understanding Discovery & Connection Data Structures, page 4-1. The Full Host Profile data block a block type value of 135. It deprecates data block 111.

**Note** An asterisk(*) next to a block name in the following diagram indicates that multiple instances of the data block may occur.



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Client Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Client Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VDB Native Fingerprints 1 | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System VDB Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VDB Native Fingerprints 2 | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System VDB Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System User Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Scan Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Scan Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Application Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Application Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Conflict Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Conflict Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (TCP) Full Server Data | List Block Type (11)... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (TCP) Full Server Data Blocks (104)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (UDP) Full Server Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (UDP) Full Server Data Blocks (104)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Network Protocol Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (Network) Protocol Data Blocks (4)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transport Protocol Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (Transport) Protocol Data Blocks (4)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| MAC Address Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Host MAC Address Data Blocks (95)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Last Seen | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Host Type | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Business Criticality | | | | | | | | | | | | | | | | VLAN ID | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | VLAN Type | | | | | | | | VLAN Priority | | | | | | | | Generic List Block Type (31) | | | | | | | | | | | | | | | | |
| Host Client Data | Generic List Block Type, continued | | | | | | | | | | | | | | | | Generic List Block Length | | | | | | | | | | | | | | | |
| | Generic List Block Length, continued | | | | | | | | | | | | | | | | Full Host Client Application Data Blocks (112)* | | | | | | | | | | | | | | | |
| NetBIOS Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | NetBIOS Name String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Notes Data | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Notes String.... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (VDB) Host Vulns | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (VDB) Host Vulnerability Data Blocks (85)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3rd Pty/VDB) Host Vulns | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (Third Party/VDB) Host Vulnerability Data Blocks (85)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3rd Pty Scan Host Vulns | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (Third Party Scan) Host Vulnerability Data Blocks with Original Vuln IDs (85)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Attribute Value Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Attribute Value Data Blocks * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Mobile | | | | | | | | Jailbroken | | | | | | | | VLAN Presence | | | | | | | | | | | | | | | |

The following table describes the components of the Full Host Profile for 5.1.1 record.

*Table B-58 Full Host Profile Record 5.1.1 Fields*

| Field | Data Type | Description |
|---|---|---|
| IP Address | uint8[4] | IP address of the host, in IP address octets. |
| Hops | uint8 | Number of network hops from the host to the device. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data derived from the existing fingerprints for the host. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Derived Fingerprint Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host derived from the existing fingerprints for the host. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a server fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Server Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a server fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a client fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Client Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a client fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a Cisco VDB fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (VDB) Native Fingerprint 1) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using the fingerprints in the Cisco vulnerability database (VDB). See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |

*Table B-58 Full Host Profile Record 5.1.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a Cisco VDB fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (VDB) Native Fingerprint 2) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using the fingerprints in the Cisco vulnerability database (VDB). See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data added by a user. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (User Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host added by a user. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data added by a vulnerability scanner. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Scan Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host added by a vulnerability scanner. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data added by an application. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Application Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host added by an application. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data selected through fingerprint conflict resolution. This value is always 31. |

*Table B-58 Full Host Profile Record 5.1.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Conflict Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host selected through fingerprint conflict resolution. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Full Server data blocks conveying TCP service data. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus the length of all encapsulated Full Server data blocks. |
| (TCP) Full Server Data Blocks * | variable | List of Full Server data blocks conveying data about the TCP services on the host. See Full Host Server Data Block 4.10.0+, page 4-141 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Full Server data blocks conveying UDP service data. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus the length of all encapsulated Full Server data blocks. |
| (UDP) Full Server Data Blocks * | variable | List of Full Server data blocks conveying data about the UDP sub-servers on the host. See Full Host Server Data Block 4.10.0+, page 4-141 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Protocol data blocks conveying network protocol data. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus the length of all encapsulated Protocol data blocks. |
| (Network) Protocol Data Blocks * | variable | List of Protocol data blocks conveying data about the network protocols on the host. See Protocol Data Block, page 4-75 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Protocol data blocks conveying transport protocol data. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus the length of all encapsulated Protocol data blocks. |
| (Transport) Protocol Data Blocks * | variable | List of Protocol data blocks conveying data about the transport protocols on the host. See Protocol Data Block, page 4-75 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block containing Host MAC Address data blocks. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the list, including the list header and all encapsulated Host MAC Address data blocks. |

*Table B-58 Full Host Profile Record 5.1.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Host MAC Address Data Blocks * | variable | List of Host MAC Address data blocks. See Host MAC Address 4.9+, page 4-115 for a description of this data block. |
| Last Seen | uint32 | UNIX timestamp that represents the last time the system detected host activity. |
| Host Type | uint32 | Indicates host type. Values include:<br>• 0 — Host<br>• 1 — Router<br>• 2 — Bridge<br>• 3 — NAT (network address translation device)<br>• 4 — LB (load balancer) |
| Business Criticality | uint16 | Indicates criticality of host to business. |
| VLAN ID | uint16 | VLAN identification number that indicates which VLAN the host is a member of. |
| VLAN Type | uint8 | Type of packet encapsulated in the VLAN tag. |
| VLAN Priority | uint8 | Priority value included in the VLAN tag. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Host Vulnerability data blocks conveying Client Application data. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Client Application data blocks. |
| Full Host Client Application Data Blocks * | variable | List of Client Application data blocks. See Full Host Client Application Data Block 5.0+, page 4-155 for a description of this data block. |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for host notes. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the notes String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the notes string. |
| Notes | string | Contains the contents of the Notes host attribute for the host. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Host Vulnerability data blocks conveying VDB vulnerability data. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated data blocks. |

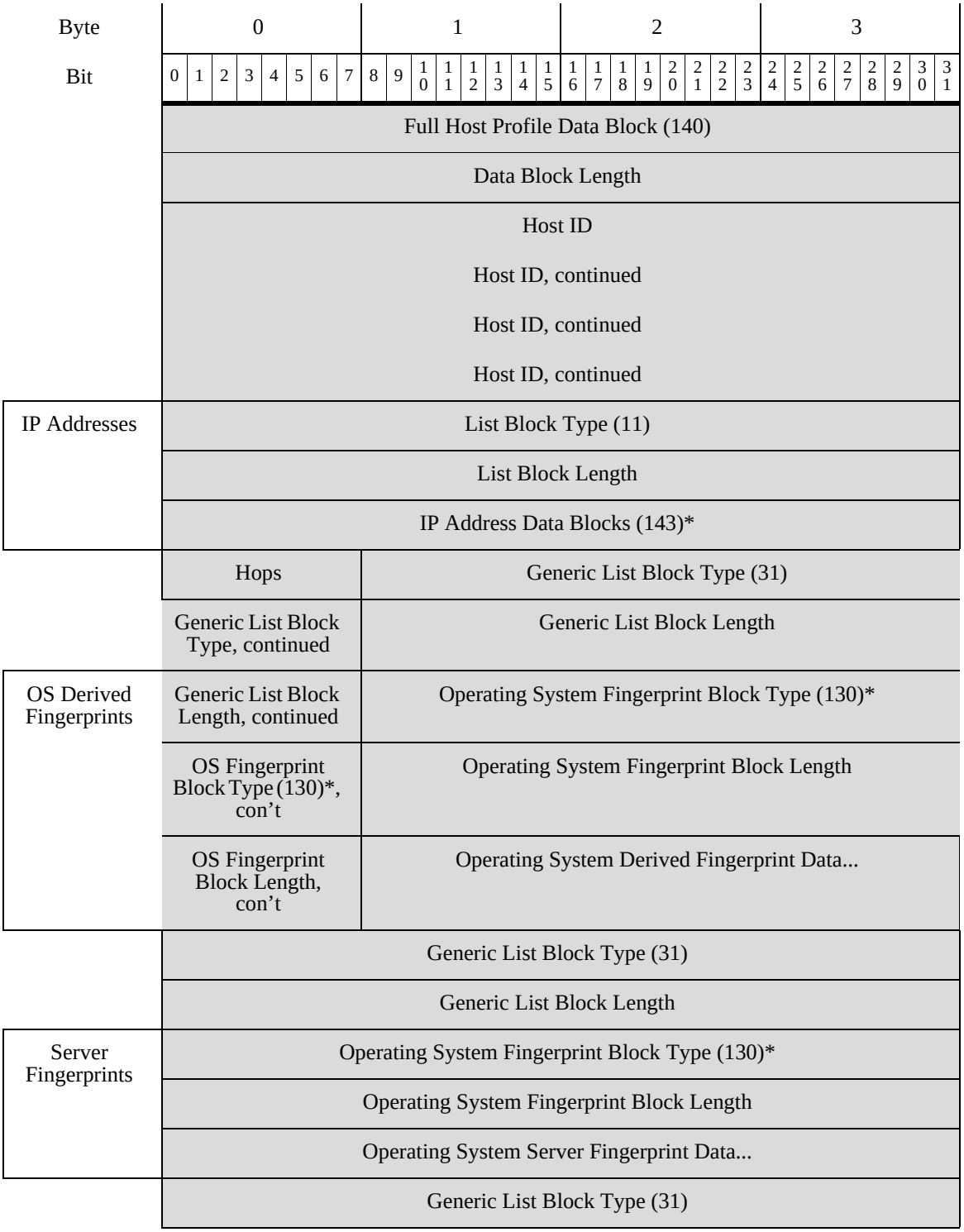
*Table B-58 Full Host Profile Record 5.1.1 Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| (VDB) Host Vulnerability Data Blocks * | variable | List of Host Vulnerability data blocks for vulnerabilities identified in the Cisco vulnerability database (VDB). See Host Vulnerability Data Block 4.9.0+, page 4-112 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Host Vulnerability data blocks conveying third-party scan vulnerability data. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated data blocks. |
| (Third Party/VDB) Host Vulnerability Data Blocks * | variable | Host Vulnerability data blocks sourced from a third party scanner and containing information about host vulnerabilities cataloged in the Cisco vulnerability database (VDB). See Host Vulnerability Data Block 4.9.0+, page 4-112 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Host Vulnerability data blocks conveying third party scan vulnerability data. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated data blocks. |
| (Third Party Scan) Host Vulnerability Data Blocks * | variable | Host Vulnerability data blocks sourced from a third party scanner. Note that the host vulnerability IDs for these data blocks are the third party scanner IDs, not Cisco-detected IDs. See Host Vulnerability Data Block 4.9.0+, page 4-112 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Attribute Value data blocks conveying attribute data. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the List data block, including the list header and all encapsulated data blocks. |
| Attribute Value Data Blocks * | variable | List of Attribute Value data blocks. See Attribute Value Data Block, page 4-82 for a description of the data blocks in this list. |
| Mobile | uint8 | A true-false flag indicating whether the operating system is running on a mobile device. |
| Jailbroken | uint8 | A true-false flag indicating whether the mobile device operating system is jailbroken. |
| VLAN Presence | uint8 | Indicates whether a VLAN is present:<br>• 0 — Yes<br>• 1 — No |

# Full Host Profile Data Block 5.2.x

The Full Host Profile data block for version 5.2.x contains a full set of data describing one host. It has the format shown in the graphic below and explained in the following table. Note that, except for List data blocks, the graphic does not show the fields of the encapsulated data blocks. These encapsulated data blocks are described separately in Understanding Discovery & Connection Data Structures, page 4-1. The Full Host Profile data block a block type value of 140. It supersedes the prior version, which has a block type of 135.

**Note** An asterisk (*) next to a block name in the following diagram indicates that multiple instances of the data block may occur.



| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Client Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Client Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VDB Native Fingerprints 1 | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System VDB Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| VDB Native Fingerprints 2 | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System VDB Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System User Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Scan Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Scan Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Application Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Application Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Conflict Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Conflict Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Mobile Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Mobile Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| IPv6 Server Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System IPv6 Server Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Ipv6 Client Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Ipv6 Client Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Ipv6 DHCP Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System IPv6 DHCP Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| User Agent Fingerprints | Operating System Fingerprint Block Type (130)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System Fingerprint Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Operating System User Agent Fingerprint Data... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (TCP) Full Server Data | List Block Type (11)... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (TCP) Full Server Data Blocks (104)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (UDP) Full Server Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (UDP) Full Server Data Blocks (104)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Network Protocol Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (Network) Protocol Data Blocks (4)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transport Protocol Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (Transport) Protocol Data Blocks (4)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| MAC Address Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Host MAC Address Data Blocks (95)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Last Seen | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Host Type | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Business Criticality | | | | | | | | | | | | | | | | VLAN ID | | | | | | | | | | | | | | | |

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | VLAN Type | | | | | | | | VLAN Priority | | | | | | | | Generic List Block Type (31) | | | | | | | | | | | | | | | | |
| Host Client Data | Generic List Block Type, continued | | | | | | | | | | | | | | | | Generic List Block Length | | | | | | | | | | | | | | | |
| | Generic List Block Length, continued | | | | | | | | | | | | | | | | Full Host Client Application Data Blocks (112)* | | | | | | | | | | | | | | | |
| NetBios Name | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Name | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | NetBIOS Name String... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Notes Data | String Block Type (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | String Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Notes String.... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (VDB) Host Vulns | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (VDB) Host Vulnerability Data Blocks (85)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3rd Pty/VDB) Host Vulns | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (Third Party/VDB) Host Vulnerability Data Blocks (85)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3rd Pty Scan Host Vulns | Generic List Block Type (31) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Generic List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | (Third Party Scan) Host Vulnerability Data Blocks with Original Vuln IDs (85)* | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Attribute Value Data | List Block Type (11) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | List Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Attribute Value Data Blocks * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Mobile | | | | | | | | Jailbroken | | | | | | | | | | | | | | | | | | | | | | | |

The following table describes the components of the Full Host Profile for 5.2.x record.

*Table B-59 Full Host Profile Record 5.2.x Fields*

| Field | Data Type | Description |
|---|---|---|
| Host ID | uint8[16] | Unique ID number of the host. This is a UUID. |
| List Block Type | uint32 | Initiates a List data block comprising IP address data blocks conveying TCP service data. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus the length of all encapsulated IP address data blocks. |
| IP Address | variable | IP addresses of the host and when each IP address was last seen. See Host IP Address Data Block, page 4-97 for a description of this data block. |
| Hops | uint8 | Number of network hops from the host to the device. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data derived from the existing fingerprints for the host. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Derived Fingerprint Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host derived from the existing fingerprints for the host. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a server fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Server Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a server fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a client fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Client Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a client fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a Cisco VDB fingerprint. This value is always 31. |

*Table B-59 Full Host Profile Record 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (VDB) Native Fingerprint 1) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using the fingerprints in the Cisco vulnerability database (VDB). See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a Cisco VDB fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (VDB) Native Fingerprint 2) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using the fingerprints in the Cisco vulnerability database (VDB). See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data added by a user. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (User Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host added by a user. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data added by a vulnerability scanner. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Scan Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host added by a vulnerability scanner. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data added by an application. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |

*Table B-59 Full Host Profile Record 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Operating System Fingerprint (Application Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host added by an application. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data selected through fingerprint conflict resolution. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Conflict Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host selected through fingerprint conflict resolution. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying mobile device fingerprint data. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Mobile) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a mobile device host. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using an IPv6 server fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (IPv6 Server Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using an IPv6 server fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using an IPv6 client fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |

*Table B-59 Full Host Profile Record 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Operating System Fingerprint (IPv6 Client Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using an IPv6 client fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using an IPv6 DHCP fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (IPv6 DHCP) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using an IPv6 DHCP fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a user agent fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (User Agent) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a user agent fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Full Server data blocks conveying TCP service data. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus the length of all encapsulated Full Server data blocks. |
| (TCP) Full Server Data Blocks * | variable | List of Full Server data blocks conveying data about the TCP services on the host. See Full Host Server Data Block 4.10.0+, page 4-141 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Full Server data blocks conveying UDP service data. This value is always 11. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus the length of all encapsulated Full Server data blocks. |
| (UDP) Full Server Data Blocks * | variable | List of Full Server data blocks conveying data about the UDP sub-servers on the host. See Full Host Server Data Block 4.10.0+, page 4-141 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Protocol data blocks conveying network protocol data. This value is always 11. |

*Table B-59 Full Host Profile Record 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus the length of all encapsulated Protocol data blocks. |
| (Network) Protocol Data Blocks * | variable | List of Protocol data blocks conveying data about the network protocols on the host. See Protocol Data Block, page 4-75 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Protocol data blocks conveying transport protocol data. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus the length of all encapsulated Protocol data blocks. |
| (Transport) Protocol Data Blocks * | variable | List of Protocol data blocks conveying data about the transport protocols on the host. See Protocol Data Block, page 4-75 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block containing Host MAC Address data blocks. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the list, including the list header and all encapsulated Host MAC Address data blocks. |
| Host MAC Address Data Blocks * | variable | List of Host MAC Address data blocks. See Host MAC Address 4.9+, page 4-115 for a description of this data block. |
| Last Seen | uint32 | UNIX timestamp that represents the last time the system detected host activity. |
| Host Type | uint32 | Indicates host type. Values include:<br>• `0` — Host<br>• `1` — Router<br>• `2` — Bridge<br>• `3` — NAT (network address translation device)<br>• `4` — LB (load balancer) |
| Business Criticality | uint16 | Indicates criticality of host to business. |
| VLAN ID | uint16 | VLAN identification number that indicates which VLAN the host is a member of. |
| VLAN Type | uint8 | Type of packet encapsulated in the VLAN tag. |
| VLAN Priority | uint8 | Priority value included in the VLAN tag. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Host Vulnerability data blocks conveying Client Application data. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Client Application data blocks. |

*Table B-59 Full Host Profile Record 5.2.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Full Host Client Application Data Blocks * | variable | List of Client Application data blocks. See Full Host Client Application Data Block 5.0+, page 4-155 for a description of this data block. |
| String Block Type | uint32 | Initiates a String data block for the host NetBIOS name. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the NetBIOS name string. |
| NetBIOS Name | string | Host NetBIOS name string. |
| String Block Type | uint32 | Initiates a String data block for host notes. This value is always 0. |
| String Block Length | uint32 | Number of bytes in the notes String data block, including eight bytes for the string block type and length fields, plus the number of bytes in the notes string. |
| Notes | string | Contains the contents of the Notes host attribute for the host. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Host Vulnerability data blocks conveying VDB vulnerability data. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated data blocks. |
| (VDB) Host Vulnerability Data Blocks * | variable | List of Host Vulnerability data blocks for vulnerabilities identified in the Cisco vulnerability database (VDB). See Host Vulnerability Data Block 4.9.0+, page 4-112 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Host Vulnerability data blocks conveying third-party scan vulnerability data. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated data blocks. |
| (Third Party/VDB) Host Vulnerability Data Blocks * | variable | Host Vulnerability data blocks sourced from a third party scanner and containing information about host vulnerabilities cataloged in the Cisco vulnerability database (VDB). See Host Vulnerability Data Block 4.9.0+, page 4-112 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Host Vulnerability data blocks conveying third party scan vulnerability data. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated data blocks. |
| (Third Party Scan) Host Vulnerability Data Blocks * | variable | Host Vulnerability data blocks sourced from a third party scanner. Note that the host vulnerability IDs for these data blocks are the third party scanner IDs, not Cisco-detected IDs. See Host Vulnerability Data Block 4.9.0+, page 4-112 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Attribute Value data blocks conveying attribute data. This value is always 11. |

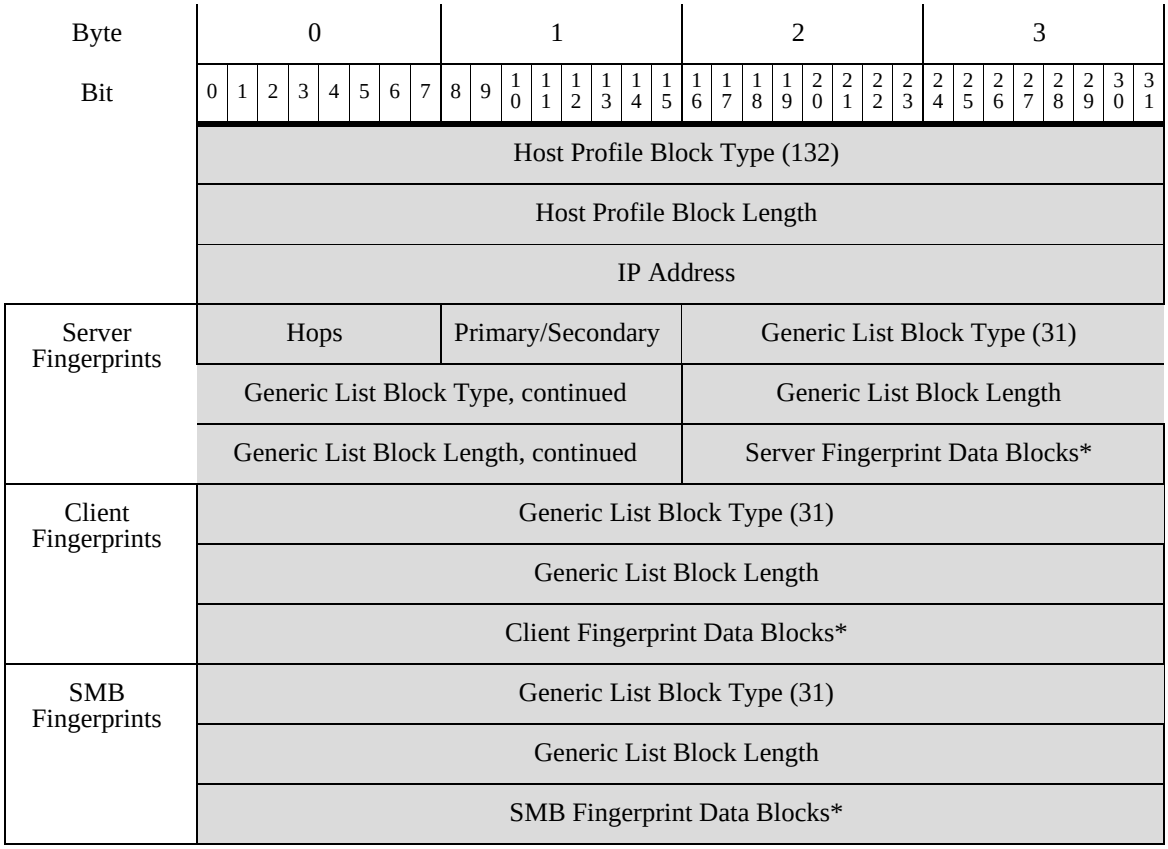
*Table B-59 Full Host Profile Record 5.2.x Fields (continued)*

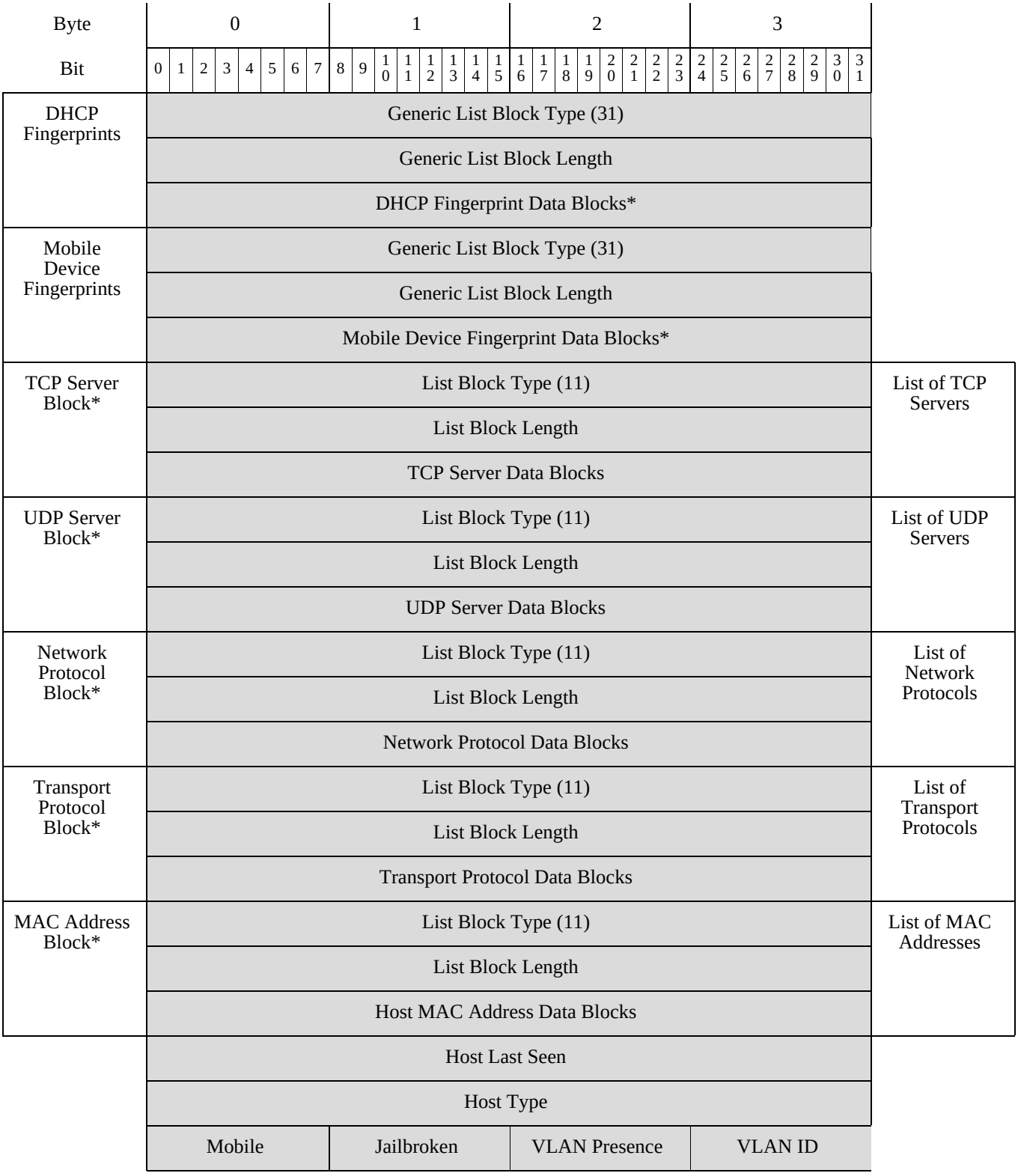
| Field | Data Type | Description |
|---|---|---|
| List Block Length | uint32 | Number of bytes in the List data block, including the list header and all encapsulated data blocks. |
| Attribute Value Data Blocks * | variable | List of Attribute Value data blocks. See Attribute Value Data Block, page 4-82 for a description of the data blocks in this list. |
| Mobile | uint8 | A true-false flag indicating whether the operating system is running on a mobile device. |
| Jailbroken | uint8 | A true-false flag indicating whether the mobile device operating system is jailbroken. |

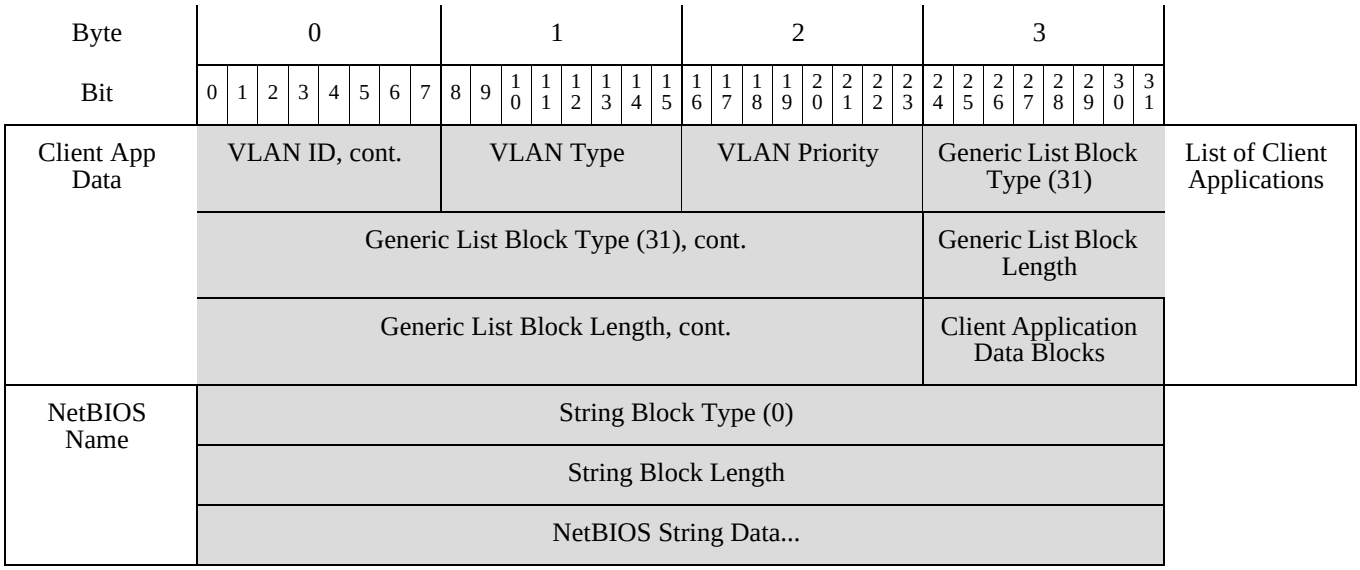
# Host Profile Data Block for 5.1.x

The following diagram shows the format of a Host Profile data block. The data block also does not include a host criticality value, but does include a VLAN presence indicator. In addition, a data block can convey a NetBIOS name for the host. The Host Profile data block has a block type of 132.

**Note** An asterisk(*) next to a block type field in the following diagram indicates the message may contain zero or more instances of the series 1 data block.




Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
DHCP Fingerprints
Generic List Block Type (31)
Generic List Block Length
DHCP Fingerprint Data Blocks*
Mobile Device Fingerprints
Generic List Block Type (31)
Generic List Block Length
Mobile Device Fingerprint Data Blocks*
TCP Server Block*
List Block Type (11)
List Block Length
TCP Server Data Blocks
List of TCP Servers
UDP Server Block*
List Block Type (11)
List Block Length
UDP Server Data Blocks
List of UDP Servers
Network Protocol Block*
List Block Type (11)
List Block Length
Network Protocol Data Blocks
List of Network Protocols
Transport Protocol Block*
List Block Type (11)
List Block Length
Transport Protocol Data Blocks
List of Transport Protocols
MAC Address Block*
List Block Type (11)
List Block Length
Host MAC Address Data Blocks
List of MAC Addresses
Host Last Seen
Host Type
Mobile
Jailbroken
VLAN Presence
VLAN ID




The following table describes the fields of the host profile data block returned by version 5.1.x

***Table B-60 Host Profile Data Block 5.1.x Fields***

| Field | Data Type | Description |
|---|---|---|
| Host Profile Block Type | uint32 | Initiates the Host Profile data block for 5.1.x. This value is always `132`. |
| Host Profile Block Length | uint32 | Number of bytes in the Host Profile data block, including eight bytes for the host profile block type and length fields, plus the number of bytes included in the host profile data that follows. |
| IP Address | uint8[4] | IP address of the host described in the profile, in IP address octets. |
| Hops | uint8 | Number of hops from the host to the device. |
| Primary/ Secondary | uint8 | Indicates whether the host is in the primary or secondary network of the device that detected it:<br>• `0` — Host is in the primary network.<br>• `1` — Host is in the secondary network. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a server fingerprint. This value is always `31`. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Server Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a server fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |

*Table B-60 Host Profile Data Block 5.1.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a client fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (Client Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a client fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using an SMB fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (SMB Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using an SMB fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a DHCP fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |
| Operating System Fingerprint (DHCP Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a DHCP fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Operating System Fingerprint data blocks conveying fingerprint data identified using a DHCP fingerprint. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated Operating System Fingerprint data blocks. |

*Table B-60 Host Profile Data Block 5.1.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Operating System Fingerprint (Mobile Device Fingerprint) Data Blocks * | variable | Operating System Fingerprint data blocks containing information about the operating system on a host identified using a mobile device fingerprint. See Operating System Fingerprint Data Block 5.1+, page 4-160 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Server data blocks conveying TCP server data. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus all encapsulated Server data blocks.<br>This field is followed by zero or more Server data blocks. |
| TCP Server Data Blocks | variable | Host server data blocks describing a TCP server (as documented for earlier versions of the product). |
| List Block Type | uint32 | Initiates a List data block comprising Server data blocks conveying UDP server data. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus all encapsulated Server data blocks.<br>This field is followed by zero or more Server data blocks. |
| UDP Server Data Blocks | uint32 | Host server data blocks describing a UDP server (as documented for earlier versions of the product). |
| List Block Type | uint32 | Initiates a List data block comprising Protocol data blocks conveying network protocol data. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus all encapsulated Protocol data blocks.<br>This field is followed by zero or more Protocol data blocks. |
| Network Protocol Data Blocks | uint32 | Protocol data blocks describing a network protocol. See Protocol Data Block, page 4-75 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising Protocol data blocks conveying transport protocol data. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the list. This number includes the eight bytes of the list block type and length fields, plus all encapsulated Protocol data blocks.<br>This field is followed by zero or more transport protocol data blocks. |
| Transport Protocol Data Blocks | uint32 | Protocol data blocks describing a transport protocol. See Protocol Data Block, page 4-75 for a description of this data block. |
| List Block Type | uint32 | Initiates a List data block comprising MAC Address data blocks. This value is always `11`. |
| List Block Length | uint32 | Number of bytes in the list, including the list header and all encapsulated MAC Address data blocks. |

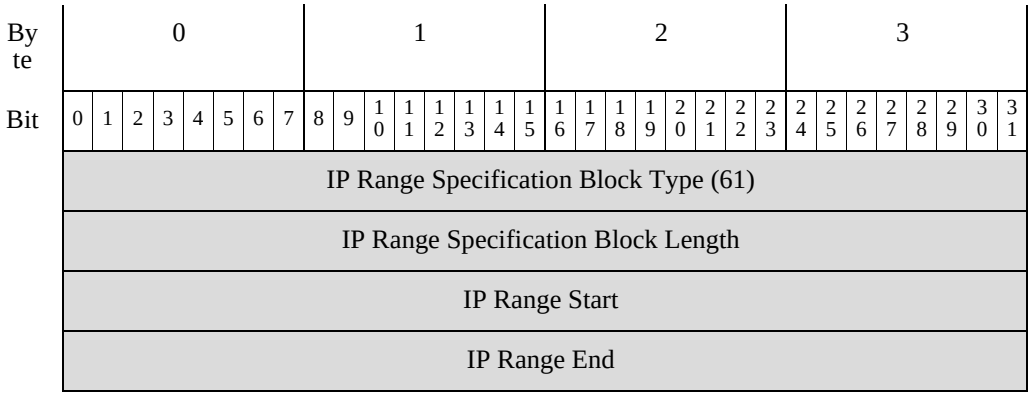
*Table B-60 Host Profile Data Block 5.1.x Fields (continued)*

| Field | Data Type | Description |
|---|---|---|
| Host MAC Address Data Blocks | uint32 | Host MAC Address data blocks describing a host MAC address. See Host MAC Address 4.9+, page 4-115 for a description of this data block. |
| Host Last Seen | uint32 | UNIX timestamp that represents the last time the system detected host activity. |
| Host Type | uint32 | Indicates the host type. The following values may appear:<br>• 0 — Host<br>• 1 — Router<br>• 2 — Bridge<br>• 3 — NAT device<br>• 4 — LB (load balancer) |
| Mobile | uint8 | True-false flag indicating whether the host is a mobile device. |
| Jailbroken | uint8 | True-false flag indicating whether the host is a mobile device that is also jailbroken. |
| VLAN Presence | uint8 | Indicates whether a VLAN is present:<br>• 0 — Yes<br>• 1 — No |
| VLAN ID | uint16 | VLAN identification number that indicates which VLAN the host is a member of. |
| VLAN Type | uint8 | Type of packet encapsulated in the VLAN tag. |
| VLAN Priority | uint8 | Priority value included in the VLAN tag. |
| Generic List Block Type | uint32 | Initiates a Generic List data block comprising Client Application data blocks conveying client application data. This value is always 31. |
| Generic List Block Length | uint32 | Number of bytes in the Generic List data block, including the list header and all encapsulated client application data blocks. |
| Client Application Data Blocks | uint32 | Client application data blocks describing a client application. See Full Host Client Application Data Block 5.0+, page 4-155 for a description of this data block. |
| String Block Type | uint32 | Initiates a string data block for the NetBIOS name. This value is set to 0 to indicate string data. |
| String Block Length | uint32 | Indicates the number of bytes in the NetBIOS name data block, including eight bytes for the string block type and length, plus the number of bytes in the NetBIOS name. |
| NetBIOS String Data | Variable | Contains the NetBIOS name of the host described in the host profile. |

# IP Range Specification Data Block for 5.0 - 5.1.1.x

The IP Range Specification data block conveys a range of IP addresses. IP Range Specification data blocks are used in User Protocol, User Client Application, Address Specification, User Product, User Server, User Hosts, User Vulnerability, User Criticality, and User Attribute Value data blocks. The IP Range Specification data block has a block type of 61.

The following diagram shows the format of the IP Range Specification data block:




The following table describes the components of the IP Range Specification data block.

***Table B-61 IP Range Specification Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| IP Range Specification Block Type | uint32 | Initiates a IP Range Specification data block. This value is always `61`. |
| IP Range Specification Block Length | uint32 | Total number of bytes in the IP Range Specification data block, including eight bytes for the IP Range Specification block type and length fields, plus the number of bytes of IP range specification data that follows. |
| IP Range Specification Start | uint32 | The starting IP address for the IP address range. |
| IP Range Specification End | uint32 | The ending IP address for the IP address range. |

# Access Control Policy Rule Reason Data Block

The eStreamer service uses the Access Control Rule Policy Rule Reason Data block to contain information about access control policy rule IDs. This data block has a block type of 21 in series 2.

The following diagram shows the structure of the Access Control Policy Rule ID metadata block.

| Byte | 0 | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| | Access Control Policy Rule Reason Data Block Type (21) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Access Control Policy Rule Reason Data Block Length | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Description | Reason | | | | | | | | | | | | | | | | String Block Type (0) | | | | | | | | | | | | | | | |
| | String Block Type (0), continued | | | | | | | | | | | | | | | | String Block Length | | | | | | | | | | | | | | | |
| | String Block Length, continued | | | | | | | | | | | | | | | | Description... | | | | | | | | | | | | | | | |

The following table describes the fields in the Access Control Policy Rule ID metadata block.

***Table B-62 Access Control Policy Rule Reason Data Block Fields***

| Field | Data Type | Description |
|---|---|---|
| Access Control Policy Rule Reason Data Block Type | uint32 | Initiates an Access Control Policy Rule Reason data block. This value is always `21`. |
| Access Control Policy Rule Reason Data Block Length | uint32 | Total number of bytes in the Access Control Policy Rule Reason data block, including eight bytes for the Access Control Policy Rule Reason data block type and length fields, plus the number of bytes of data that follows. |
| Reason | uint16 | The number of the reason for the rule that triggered the event. |
| String Block Type | uint32 | Initiates a String data block containing the description of the access control policy rule reason. This value is always `0`. |
| String Block Length | uint32 | The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Description field. |
| Description | string | Description of the reason for the rule. |