Introduction

The Cisco Event Streamer (also known as eStreamer) allows you to stream Firepower System events to external client applications. While eStreamer continues to support the binary streaming of events, you may also request fully-qualified events. These events are in a clear text format with metadata resolved. This guide describes how to request these fully-qualified events from the eStreamer service.

Connection events, intrusion events, intrusion packets, and file events are available as fully-qualified events from a Management Center.

Note that eStreamer is not supported on NGIPSv, Firepower Services, Firepower Threat Defense Virtual, and Firepower Threat Defense. To stream events from these devices, you can configure eStreamer on the Management Center that the device reports to.

eStreamer uses a custom application layer protocol to communicate with connected client applications. As the purpose of eStreamer is simply to return data that the client requests, the majority of this guide describes the eStreamer formats for the requested data.

There are four major steps to creating and integrating an eStreamer client with a Firepower System:

1. Write a client application that exchanges messages with the Management Center or managed device using the eStreamer application protocol.

2. Configure a Management Center or device to send the required type of events to your client application.

3. Connect your client application to the Management Center or device.

4.Specify the data and format you want and begin exchanging data.

This guide provides the information you need to successfully create and run an eStreamer client application which receives fully-qualified events.

- Using this Guide, on page 1
- Prerequisites, on page 2
- IP Addresses, on page 2
- Best Practices, on page 2

Using this Guide

- At the highest level, the eStreamer service is a mechanism for streaming data from the Secure Firewall System to a requesting client. The service can stream the following fully-qualified events:
  - Connection Events

- Intrusion Events
- Intrusion Packets
- File Events

Descriptions of the data structures returned by eStreamer make up the majority of this book. The chapters in the book are:

Prerequisites

To understand the information in this guide, you should be familiar with the features and nomenclature of the Secure Firewall System and the function of its components in general, and with the different types of event data these components generate in particular. Definitions of unfamiliar or product-specific terms can frequently be obtained from the *Secure Firewall eStreamer Integration Guide.*

You should also be familiar with CSV (comma-separated values) or JSON (JavaScript Object Notation) file format and be able to create a program or script which can use one of these formats.

IP Addresses

The Cisco database stores IPv4 and IPv6 addresses in the same fields in a hexadecimal format. To get IPv6 addresses, convert to hex notation, for example: `20010db800000000000000000000004321`. The database follows the RFC for storing IPv4 addresses by filling in bits 80-95 with 1's, which yields an invalid IPv6 address. For example, the IPv4 address `10.5.15.1` would be stored as `00000000000000000000FFFF0A050F01`.

Best Practices

When working with eStreamer, Cisco recommends the following for best use of the API.

Design

- Build your eStreamer client to support everything the API can provide, as every bit of the schema is important to at least some small part of the customer base.
- Implement robust error handling and logging so that when something goes wrong, you can see the message and situation that caused the problem without necessarily needing to reproduce the error.
- Pick your language carefully. Parsing might not seem that computationally expensive but when there are thousands of events per second, everything counts. Compiled languages such as C, C++, Go will be faster than Python / JavaScript. The downside of such an approach is lack of portability.
- Look at the existing eStreamer implementations to see how others have accomplished your goals in the past. Some resources:

  https://splunkbase.splunk.com and search for eStreamer

  https://software.cisco.com/download/home/ next to "Select a Product" select "Browse All", then "Security", followed by "Firewalls", "Firewall Management", "Firepower Management Center Virtual Appliance", then "Firepower System Tools and APIs".

https:/community.cisco.com and search for “eNcoreCLI”.

- Make sure to work with the Cisco Security Technical Alliance team to keep up with changes to eStreamer and other aspects of integrating with Cisco Firepower. You can contact them at ask-csta-pm@cisco.com

Testing

- When Cisco introduces a new version of Firepower, promptly test your client against it to make sure the data collected by your client does not change.
- Have a good test bed so you can test easily and frequently.
- If you prefer to not build your own test bed, use the dcloud sandbox test bed. The Cisco Security Technical Alliance will provide resources to assist in setting up and using it. Dcloud is free and enables comprehensive testing. However, it is not necessarily complete for your use and does not have 100% event coverage. Also, instances are only available for short periods of time. For more information about dcloud, go to https://dcloud2-rtp.cisco.com