Overview of log file formats

You've learned about how logs record events that happen on a network, or system. In security, logs provide key details about activities that occurred across an organization, like who signed into an application at a specific point in time. As a security analyst, you'll use **log analysis**, which is the process of examining logs to identify events of interest. It's important to know how to read and interpret different log formats so that you can uncover the key details surrounding an event and identify unusual or malicious activity. In this reading, you'll review the following log formats:

- JSON
- Syslog
- XML
- CSV
- CEF

JavaScript Object Notation (JSON)

JavaScript Object Notation (JSON) is a file format that is used to store and transmit data. JSON is known for being lightweight and easy to read and write. It is used for transmitting data in web technologies and is also commonly used in cloud environments. JSON syntax is derived from JavaScript syntax. If you are familiar with JavaScript, you might recognize that JSON contains components from JavaScript including:

- Key-value pairs
- Commas
- Double quotes
- Curly brackets
- Square brackets

Key-value pairs

A **key-value pair** is a set of data that represents two linked items: a key and its corresponding value. A key-value pair consists of a key followed by a colon, and then followed by a value. An example of a key-value pair is `"Alert": "Malware"`.

Note: For readability, it is recommended that key-value pairs contain a space before or after the colon that separates the key and value.

Commas

Commas are used to separate data. For example: `"Alert": "Malware", "Alert code": 1090, "severity": 10`.

Double quotes

Double quotes are used to enclose *text* data, which is also known as a string, for example: `"Alert": "Malware"`. Data that contains numbers *is not* enclosed in quotes, like this: `"Alert code": 1090`.

Curly brackets

Curly brackets enclose an **object**, which is a data type that stores data in a comma-separated list of key-value pairs. Objects are often used to describe multiple properties for a given key. JSON log entries start and end with a curly bracket. In this example, `User` is the object that contains multiple properties:

```
"User": {  "id": "1234",  "name": "user", "role": "engineer" }
```

Square brackets

Square brackets are used to enclose an **array**, which is a data type that stores data in a comma-separated ordered list. Arrays are useful when you want to store data as an ordered collection, for example: `["Administrators", "Users", "Engineering"]`.

Syslog

Syslog is a standard for logging and transmitting data. It can be used to refer to any of its three different capabilities:

1. **Protocol**: The syslog protocol is used to transport logs to a centralized log server for log management. It uses port 514 for plaintext logs and port 6514 for encrypted logs.
2. **Service**: The syslog service acts as a log forwarding service that consolidates logs from multiple sources into a single location. The service works by receiving and then forwarding any syslog log entries to a remote server.
3. **Log format**: The syslog log format is one of the most commonly used log formats that you will be focusing on. It is the native logging format used in Unix® systems. It consists of three components: a header, structured-data, and a message.

Syslog log example

Here is an example of a syslog entry that contains all three components: a header, followed by structured-data, and a message:

```
<236>1 2022-03-21T01:11:11.003Z virtual.machine.com evntslog - ID01
[user@32473 iut="1" eventSource="Application" eventID="9999"] This is a log
entry!
```

Header

The header contains details like the timestamp; the hostname, which is the name of the machine that sends the log; the application name; and the message ID.

- **Timestamp**: The timestamp in this example is `2022-03-21T01:11:11.003Z`, where `2022-03-21` is the date in YYYY-MM-DD format. `T` is used to separate the date and the time. `01:11:11.003` is the 24-hour format of the time and includes the number of milliseconds `003`. `Z` indicates the timezone, which is Coordinated Universal Time (UTC).
- **Hostname**: `virtual.machine.com`

- **Application**: `evntslog`
- **Message ID**: `ID01`

Structured-data

The structured-data portion of the log entry contains additional logging information. This information is enclosed in square brackets and structured in key-value pairs. Here, there are three keys with corresponding values: **`[user@32473 iut="1" eventSource="Application" eventID="9999"]`**.

Message

The message contains a detailed log message about the event. Here, the message is **`This is a log entry!`**.

Priority (PRI)

The priority (PRI) field indicates the urgency of the logged event and is contained with angle brackets. In this example, the priority value is **`<236>`** . Generally, the lower the priority level, the more urgent the event is.

Note: Syslog headers can be combined with JSON, and XML formats. Custom log formats also exist.

XML (eXtensible Markup Language)

XML (eXtensible Markup Language) is a language and a format used for storing and transmitting data. XML is a native file format used in Windows systems. XML syntax uses the following:

- Tags
- Elements
- Attributes

Tags

XML uses tags to store and identify data. Tags are pairs that must contain a start tag and an end tag. The start tag encloses data with angle brackets, for example **`<tag>`**, whereas the end of a tag encloses data with angle brackets and a forward slash like this: **`</tag>`**.

Elements

XML elements include *both* the data contained inside of a tag and the tags itself. All XML entries must contain at least one root element. Root elements contain other elements that sit underneath them, known as child elements.

Here is an example:

```
<Event> <EventID>4688</EventID> <Version>5</Version> </Event>
```

In this example, **<Event>** is the root element and contains two child elements **<EventID>** and **<Version>**. There is data contained in each respective child element.

Attributes

XML elements can also contain attributes. Attributes are used to provide additional information about elements. Attributes are included as the second part of the tag itself and must always be quoted using either single or double quotes.

For example:

```
<EventData> <Data Name='SubjectUserSid'>S-2-3-11-160321</Data> <Data
Name='SubjectUserName'>JSMITH</Data> <Data
Name='SubjectDomainName'>ADCOMP</Data> <Data
Name='SubjectLogonId'>0x1cf1c12</Data> <Data
Name='NewProcessId'>0x1404</Data> </EventData>
```

In the first line for this example, the tag is **<Data>** and it uses the attribute **Name='SubjectUserSid'** to describe the data enclosed in the tag **S-2-3-11-160321**.

CSV (Comma Separated Value)

CSV (Comma Separated Value) uses commas to separate data values. In CSV logs, the position of the data corresponds to its field name, but the field names themselves might not be included in the log. It's critical to understand what fields the source device (like an IPS, firewall, scanner, etc.) is including in the log.

Here is an example:

```
2009-11-24T21:27:09.534255,ALERT,192.168.2.7,
1041,x.x.250.50,80,TCP,ALLOWED,1:2001999:9,"ET MALWARE BTGrab.com Spyware
Downloading Ads",1
```

CEF (Common Event Format)

Common Event Format (CEF) is a log format that uses key-value pairs to structure data and identify fields and their corresponding values. The CEF syntax is defined as containing the following fields:

```
CEF:Version|Device Vendor|Device Product|Device Version|Signature
ID|Name|Severity|Extension
```

Fields are all separated with a pipe character **|**. However, anything in the **Extension** part of the CEF log entry must be written in a key-value format. Syslog is a common method used to transport logs like CEF. When Syslog is used a timestamp and hostname will be prepended to the CEF message. Here is an example of a CEF log entry that details malicious activity relating to a worm infection:

```
Sep 29 08:26:10 host CEF:1|Security|threatmanager|1.0|100|worm successfully
stopped|10|src=10.0.0.2 dst=2.1.2.2 spt=1232
```

Here is a breakdown of the fields:

- **Syslog Timestamp**: `Sep 29 08:26:10`
- **Syslog Hostname**: `host`
- **Version**: `CEF:1`
- **Device Vendor**: `Security`
- **Device Product**: `threatmanager`
- **Device Version**: `1.0`
- **Signature ID**: `100`
- **Name**: `worm successfully stopped`
- **Severity**: `10`
- **Extension**: This field contains data written as key-value pairs. There are two IP addresses, `src=10.0.0.2` and `dst=2.1.2.2`, and a source port number `spt=1232`. Extensions are not required and are optional to add.

This log entry contains details about a `Security` application called `threatmanager` that `successfully stopped a worm` from spreading from the internal network at `10.0.0.2` to the external network `2.1.2.2` through the port `1232`. A high severity level of `10` is reported.

Note: Extensions and syslog prefix are optional to add to a CEF log.

Key takeaways

There is no standard format used in logging, and many different log formats exist. As a security analyst, you will analyze logs that originate from different sources. Knowing how to interpret different log formats will help you determine key information that you can use to support your investigations.