COMMON EVENT EXPRESSION (CEE) OVERVIEW

The CEE Editorial Board
4 November 2010

VERSION: 1.0

EDITORS:

Eric Fitzgerald, Microsoft Corporation

Dr. Anton Chuvakin, Security Warrior Consulting

Bill Heinbockel, MITRE Corporation

Dominique Karg, Alienvault

Raffael Marty, Loggly

Contents

Overview

This document provides an introductory overview of the Common Event Expression (CEE) standard effort. CEE is an interoperability standard for electronic systems. CEE standardizes the representation of event records in logs to achieve interoperability by providing a solution for four problems:

- *What are the fields in the event and what do they mean?*
  CEE proposes a common, extensible event record **syntax.**
- *What does an event record mean?*
  CEE proposes a common, extensible **taxonomy** for events.
- *As an event producer, what events should I log and what data should I include in those events?*
  CEE proposes a common, extensible set of logging **recommendations.**
- *How can an event record be moved between systems while still preserving the solutions for the above problems?*
  CEE proposes required characteristics for common log **transport** (but does not mandate a particular protocol.

Standard Terminology

An ***event*** is an occurrence that is observable by an IT system. Most events involve an attempted or actual state change of an IT system. Examples of events include user logins, network connections, disk failures and function calls during program execution, as well as physical events such as building access or changes in pump pressure.

An ***event field***, or just ***field***, describes one characteristic of an event. Examples of an event field include date/time, user name, source IP address, device identifiers, function names in source code, and building door identifiers.

An ***event record*** is a collection of event fields that, together, describe a single event. Terms synonymous to event record include "audit record" and "log entry".

An ***event producer*** is a device or software component which observes an event and generates an event record containing the details about the event.

An ***event consumer*** is a device or software component which receives event records from an event producer or intermediate event record handling system.

An ***event log***, or just ***log***, is an ordered collection of event records. Terms such as "data log," "activity log" and "log file" are often used to mean “event log”. Terms such as "audit log" and "audit trail" are used to describe specific types of event logs which contain security-relevant event records. Event logs might be persistent, as in a file stored on disk or a printout, or they might be ephemeral, as in a stream of event records provided to a subscriber over a network.

Recording is the act of capturing the details regarding a single event and forming an event record for the event.

Logging is the act of storing event records persistently or providing them to an event consumer for storage.

What problems is CEE trying to solve?

CEE proposes to increase the utility of event logs by making event records unambiguous and understandable by both people and IT systems.

Achieving this goal requires solutions to several technical problems. Below is a discussion of each of the technical problems and the approach that CEE takes to a solution.

Common Log Syntax (CLS)

PROBLEM: People and systems must be able to extract information from the event records they receive.

In order for automation to efficiently extract information from and correlate event records, a few fundamental problems must be solved. For example, in order for an event consumer to extract and/or analyze a specific field of a specific event record, the event consumer must be able to understand where event records begin and end, and within each event record, where each fields begins and ends.

Trivially, each event record could be treated as a string and the event consumer could perform a full text search on the log and hope for consistency. But for reliable automated event correlation, it's necessary that event consumers understand and agree on the syntax of the event record.

CEE solves these problems by requiring unambiguous field delimitation and other structural requirements. CEE does not mandate a particular log format and in particular CEE is designed to work with flat delimited log formats (such as CSV and RFC 5424 structured syslog) as well as with hierarchical formats such as XML and JSON.

Although people are more flexible than automation in this regard, people typically use tools even in manual log analysis. And even with flexible tools like regular expressions, non-uniform delimiting and other structural problems might prevent proper analysis or at the very least dramatically increase analyst workload.

PROBLEM: People and systems must be able to extract the information from the fields of the event records that they receive.

Each field of every event record contains some information in some representation.

Consider timestamps. Although perhaps obvious to a human being, it's not at all clear that "Sun 10 3 2010 3:55pm" and "2010-10-03T11:55:00Z" represent the same time. In fact in the first example, the time ambiguous due to lack of time zone information, and the date is ambiguous due to regional date order preference concerns.

In order for an event consumer to understand the time at which an event occurred, the timestamp in the event record must be represented in an understandable format, and the receiver must understand the representation of the timestamp – it must understand whether a time zone offset has been applied to the timestamp, and whether a daylight savings time offset has been applied.

CEE solves this problem with a "data dictionary" which defines types and their associated syntaxes. When event records are defined, each field in each event is associated with a type from the data dictionary. Fields containing timestamps can have a declared syntax of "ISO 8601". Other fields can have other declared syntaxes.

The data dictionary is contained in machine readable (XML) profile documents. The CEE base data dictionary will be published by MITRE when the CEE standard is finalized. The data dictionary can be expanded by event producers if necessary. Extension dictionaries must be published by the event producer in such a way that event consumers can discover and obtain the dictionary via automation.

PROBLEM: People and systems must be able to understand the context of the data in the fields of the event records that they receive.

Sometimes event records have multiple fields containing data of the same type. The simplest fields might only contain a field identifier and a string value. However, in order for CEE to enable better automation and correlation, each field value should be of an agreed upon value type (e.g., integer, date/time, IPv4 dotted-decimal address).

As an example, a field containing a piece of information such as an IPv4 address can be represented in a standard syntax, but understanding the syntax of the data in the event record still does not allow full understanding of the field in the context of the event. For example, a single event record might contain more than one IPv4 address, and each address might have a different relationship with the underlying event, such as firewall event records containing “source” and “destination” IP addresses.

CEE solves this problem by declaring field types in data dictionaries. In addition to defining field types, such as “IPv4 address”, the data dictionary allows declaration of named instances of those types, such as “*source* IPv4 address”. These field names are used in the declaration of event records, which will be described shortly.

Event Taxonomy (ET)

PROBLEM: People and systems must be able to understand the event that an event record represents.

For example, the act of a human being presenting credentials to a system, having the system validate those credentials, and having the system create a session for the user account associated with those credentials, is commonly called a "logon". However some systems call this a "login", or say that the user "logged in". Some systems have events for the entire process of logging on, and some systems have events for constituent parts of the logon process such as the validation of credentials.

CEE solves this problem with "tags". Tags are typed metadata that carry meaning about the event. For example, an event record representing a logon event, could carry tags that indicate that the event record refers to a "account" object, and that the type of action or activity the event represents is a "logon", and that the logon was successful. Tags can carry any sort of metadata about the event record.

The CEE working group recommends a minimal set of 3 tags for every event: an "object" tag, an "action" tag, and a "status" tag. Colloquially the CEE working group refers to this base set of tags as "OAS". The OAS set of tags, taken as a unit, provide a meaningful, unambiguous descriptor of the activity represented by the event record.

As with field types and field names, tags are declared in a data dictionary, and event producers can declare new tags when defining their own event records. However, as with field types and field names, declaration of new tags reduces the ability of event consumers to correlate event records.

The CEE working group considered but rejected the creation and maintenance of a formal, hierarchical event taxonomy or a master list of all event records that can be logged.

No matter how such a taxonomy is organized, many event records would likely map into multiple places in the taxonomy, and many events which seem closely related to each other would likely appear at different places in the taxonomy as a result.

For example, should logon records for an application be associated with the application, or should all logon records be associated together in a security-related grouping?

Common Event Log Recommendations (CELR)

PROBLEM: IT purchasers need a way of communicating required event functionality to software producers.

PROBLEM: Event producers need a way of communicating to system operators and event consumers the set of event records and fields that a product generates.

How does a software vendor (event producer) know what event records to generate? How does an event producer know what fields to put in each of those event records in order for the event to be useful to event consumers?

Once a product is built, how does it describe to event consumers, what event records it can generate and what fields are in those event records?

CEE solves these problems with a document called a profile.

Structurally, a profile is an XML document that conforms to a schema which will be published as part of the CEE standard. Note that while the profile is an XML document, the event records described in the profile are never required to be represented in XML. A profile can describe event records that will be generated and transported in syslog or CSV other flat text formats, as well as XML- and JSON-formatted hierarchical formats.

A profile document contains two major sections:

1. A data dictionary, which defines field types and their associated syntaxes (ex: IPv4 address), and field names that instantiate those types in particular semantic contexts (ex: source IPv4 address).
2. A set of event definitions, which contain a minimal and recommended set of tags for each event record, and a minimal and recommended set of fields of each event record

Automated event consumer tools will be able to read a profile, and then will be able to correctly parse event records that conform to the profile.

System operators, with only standard text file tools, will be able to understand the set of event records that a product generates, including the field set and meaning of each event record.

As part of the CEE standard, a CEE Base Event profile will be published which will define common field types and names, and will define the basic structure of CEE events. For example, things like timestamp, log source type, definition of the OAS tags, must be present in the CEE Base Event profile.

The CEE working group will then work with industry experts in narrow fields of interest to event consumers, and develop profiles of a “best practices” sort that describe the set of event records and fields needed by event consumers to analyze particular functional areas. These types of profiles are called “functional” profiles. As of this writing, the CEE working group plans to publish functional profiles for firewall events and for security auditing events for operating systems, as soon as possible after the CEE standard is published.

An event producing product doesn’t have to conform to any published profile. As long as the event records generated by a product conform to CEE’s CLS requirements, an event producer may create and publish their own profile document that describes the events that their product generates. CEE does require that the profile be published and available via automation to event consumers, but does not require that any particular event record conform to any particular profile other than the CEE base profile.

Event Transport

PROBLEM: Event consumers must receive event records from event producers, with all CEE-required metadata intact.

There are many event transport protocols in existence- syslog and variants, WS-Management, and numerous proprietary and product-specific protocols.

The CEE working group has taken the approach of describing protocol requirements rather than mandating use of a particular protocol.

CEE Transport requirements will address areas such as:

- Preservation of semantic metadata
- Syntax preservation of individual event records and fields, and of the data in the fields
- Sequencing
- Encoding
- Integrity (of individual event records and of the event stream as a whole)

The transport requirements will therefore not require the use of a particular protocol, but may mandate that existing protocols be used in particular ways.

For example, to conform to CEE standards, a product which uses syslog to transport events might have to use syslog STRUCTURED-DATA features from RFC 5424, in order to preserve event field delimiting and event field metadata, and to standardize the time stamp .

How do we get there from here?

PROBLEM: There has never been a widely successful event log standard. How will CEE succeed where others have failed?

The CEE standard, when published, will have “conformance levels”. Instead of requiring every vendor of every product to change all their products in fundamental ways, CEE will offer a way for vendors to make incremental steps to conformance, with each step conferring a benefit on both the vendor and the event consumers who use the vendor’s product.

For example, CEE might propose a “basic” conformance level which requires proper field delimiting and time stamping and other such low-level syntax requirements. This does not require that products immediately start logging new events or new information in existing events.

Higher levels of conformance would require events to conform to vendor-specific or mandatory requirements in a CEE standard functional profile, and the highest levels of conformance would require conformance to all requirements of a CEE standard functional profile.

At each level of conformance, the amount of effort by the event producer is greater, but the corresponding value to event consumers is also commensurately greater.

Since he CEE specification, data dictionary, taxonomy and all functional requirements are published in profile documents, then it becomes possible to write an automated test tool which consumes an event stream and measures conformance to a profile.

Functional profiles enable purchasers of event producing software to communicate their needs to event producers. For example, in a Request for Proposal, a purchasing agent could specify the desire to purchase a firewall product that conforms to the “CEE Firewall Functional Profile 1.0”. Firewall vendors are therefore incentivized to produce products which emit events that conform to that profile.

Another way that the CEE working group hopes to promote adoption is by encouraging “middleware” or community products to assist with conformance.

Many companies and some private individuals and organizations have done extensive work doing field mappings and other exercises to normalize and enhance the meaning of events from other companies’ products.

If a product emits a useful but non-conformant event stream, then anyone who has the knowledge can write a product which consumes the original event stream, normalizes it to CEE requirements, add semantic information, and publishes a normalized CEE event stream for other CEE-compliant products to consume. Such a product can publish its own profile of the normalized events and gain CEE conformance on its own.

In this fashion, it is the hope of the CEE working group that popular products will effectively become CEE conformant even before the vendors of those products make any change.

Conclusion

You should now have a clear understanding of how the CEE effort is trying to solve the various problems with current event and audit standards, as well as the approach that the working group is taking to each problem:

- The CEE Common Log Syntax (CLS) allows event consumers to extract information from event records and the fields in those records? How do consumers extract meaning from the information in event record fields? The Common Log Syntax (CLS) effort solves these problems by associating an explicitly declared syntax and semantic meaning with every field of every event record.

- The CEE Event Taxonomy (ET) allows event consumers understand what each event record means by associating a set of tags with every event record which conveys the semantic meaning of the event, and recommending a standard set of tags ("OAS") for every event record.

- The CEE Common Event Log Recommendations (CELR) allow event consumers tell product vendors what event records to generate and what to put in these records by defining a "profile" document that describes the desired event record format. These profiles may leverage standard profiles, published by the CEE community, that reflect industry best practices.

- The CEE Common Event Log Recommendations (CELR) allow event producing products describe the events that they produce to event consumers by requiring that event producers declare their events in a profile document which is programmatically available to event consumers.

- The CEE Event Transport specification enables event consumers to be sure that structure and meaning are preserved when event records are moved between systems by providing specific requirements for event transports that ensure that the received event record is still CEE conformant if transported over a channel that complies with the specification.

- CEE will succeed in generating industry adoption where other efforts have failed because CEE provides defined, testable conformance levels, where each level of conformance corresponds with both the level of effort required by the event provider, and the value to the event consumer. In turn, conformance levels and profiles allow a customer to precisely specify to a product vendor, what set of event records is expected from a product. Additionally, vendors or private parties can author automation and profiles to make other vendors' products conformant to CEE

The CEE editorial board would like to invite you to review our draft specifications and provide feedback, as well as participate in a general discussion about the CEE standard. For more information on CEE and how to participate, please visit the CEE web site at http://cee.mitre.org.