Common Event Expression

Larry Shields [lshields@mitre.org]
William Heinbockel [heinbockel@mitre.org]
Organization

■ The Situation
■ The Problem
■ The Goals
■ The Solution
■ The Standard: CEE
  – Common Event Expression Taxonomy (CEET)
  – Common Log Syntax (CLS)
  – Common Log Transport (CLT)
  – Common Event Log Recommendations (CELR)
■ CEE & EMAP
  – Validation
The Situation

- Firewall
- AV
- IDS/HIDS
- SIEM
- Log Infrastructure
The Event Space

- Events
  - Traffic
  - Weather
  - SCADA
  - Computer Generated
    - Firewall
    - IDS
    - OS
  - First Responder
Computer Generated Event Management

EVENT

REPORTS

TRANSPORT

LOG

- Log Repository
- Log Management Solution
- SIEM Product
- SAN / NAS

USER
SYSTEM
SERVICE OR APPLICATION
OTHERS

Electronic System
External Sources

RECORD
What are our logs telling us?
Why Standardize?

- **Cryptic Records**
  Sep 01 08:11:53 Last message repeated 5 times

- **Missing and Inconsistent Event Details**
  - **Problem: Inconsistent Success/Fail**
    Apr 10 12:31:34 host sshd[16682]: error: PAM: Authentication failure for user from remote-pc.mitre.org
    Apr 10 12:31:39 host sshd[16701]: Accepted keyboard-interactive/pam for user from 192.168.0.1 port 2880 ssh2

- **Year?**
- **Time zone?**
- **DNS vs. IP?**
- **Different PAM Notation?**
Why Standardize – Another Example

- Inconsistent Event Descriptions

  Sep 22 10:02:00 myhost login(pam_unix)[808]: session opened for user root by LOGIN(uid=0)

  Sep 26 12:17:32 myhost-- root[808]: ROOT LOGIN ON tty1

  Sep 26 13:00:40 myhost snort: [1:5503:6] POLICY ROOT login attempt [Classification: Misc activity] [Priority: 3]: {TCP} 6.7.8.9:32804 -> 1.2.3.4:23

  Log events are like a box of chocolates, you never know what you’re gonna get…
The Problem (The tl;dr Version)

LOGS ARE PRODUCED FOR THE WRONG AUDIENCE

Humans understand semantics
Systems understand syntactics
The Goals: Format Neutrality
The Goals: Simplicity

3. Find x.

Here it is
The Goals: Extensibility
## Other Attempts to Standardize

<table>
<thead>
<tr>
<th>Standard</th>
<th>Description</th>
<th>Challenges</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CIDF</strong> <em>(Common Intrusion Detection Framework)</em></td>
<td>Started in 1998, LISP-like structure, Protocol &amp; API for intrusion detection information exchange</td>
<td>Specifically focused on intrusion detection, no longer active</td>
</tr>
<tr>
<td><strong>IDMEF</strong> <em>(Intrusion Detection Message Exchange Format)</em></td>
<td>For IDS/IPS systems and management systems that interact with them</td>
<td>Narrow focus on intrusion detection events, XML over BEEP format only</td>
</tr>
<tr>
<td><strong>CEF</strong> <em>(Common Event Format)</em></td>
<td>Created by ArcSight, name/value pair based, can leverage flat files or syslog</td>
<td>Vendor specific, small number of attributes (those needed/used by the product)</td>
</tr>
<tr>
<td><strong>XDAS</strong> <em>(Distributed Audit Services)</em></td>
<td>Start in 1998 as an API for Unix, adopted by SCO. In 2008 work taken by Novell to create v2, and make a more general standard.</td>
<td>Strong focus on audit use-case, Unix-centric API</td>
</tr>
</tbody>
</table>
The Solution
From Events to Logs and Back Again
CEE Building Blocks

- CEE Taxonomy (CEET)
  - Data Dictionary
  - Object-Action-Status (OAS) Taxonomy
- Common Log Syntax (CLS)
- Common Log Transport (CLT)
- Common Event Log Recommendations (CELR)
  - Best Practices
  - Device Profiles
Building Blocks Today

Scenario: An attacker has breached our network - determine if there were any successful logins

What do we search for? (‘log in’, ‘login’, ‘logged on’, etc.)
Event Attributes

- Names are designed to be composable
- Types to aide programming and validation
- Restrictions not enforced – just expected values

---

<table>
<thead>
<tr>
<th>Unique Name</th>
<th>Type</th>
<th>Description</th>
<th>Restrictions</th>
</tr>
</thead>
<tbody>
<tr>
<td>netDstPort</td>
<td>integer</td>
<td>Destination port</td>
<td>0-65535</td>
</tr>
<tr>
<td>logSrcMac</td>
<td>mac</td>
<td>MAC address of the log source</td>
<td></td>
</tr>
<tr>
<td>eventTime</td>
<td>time</td>
<td>The time at which the event occurred</td>
<td></td>
</tr>
<tr>
<td>logTime</td>
<td>time</td>
<td>The time when the event was recorded</td>
<td></td>
</tr>
<tr>
<td>netSrcIpv6</td>
<td>ipv6</td>
<td>The IPv6 address of the network source</td>
<td></td>
</tr>
</tbody>
</table>
CEE Taxonomy – OAS Taxonomy

- **Context**
- **Object**
- **Action**
- **Status**

**Example:**
Sep 26 12:17:32 myhost-- root[808]: ROOT LOGIN ON tty1

**OAS Taxonomy:**
firewall-admin-login-success

**Data Dictionary Elements:**
- logTime
- netSrcHostname
- procName
- procId
- acctName
- ttyName
Common Event Logging Recommendations - Profiles

- Specifies:
  - OAS Taxonomy Events
  - Required Data Elements
  - Recommended Data Elements

- Device specific profiles – guidance of what must/should be logged

- Provides ability to validate logged events to verify CEE compliance
Common Log Syntax Examples

Sep 26 12:17:32 myhost-- root[808]: ROOT LOGIN ON tty1

■ XML Example:

```xml
<event name="firewall-admin-login-success">
  <logTime>2009-09-16T12:17:32</logTime>
  <netSrcHostname>myhost</netSrcHostname>
  <procName>root</procName>
  <procId>808</procId>
  <acctName>root</acctName>
  <ttyName>tty1</ttyName>
</event>
```

■ Plaintext Example:

```plaintext
event="firewall-admin-login-success" logTime="2009-09-16T12:17:32"
netSrcHostname="myhost" procName="root" procId="808" acctName="root"
ttyName="tty1"
```
Common Log Transport

- **Goal:** Let’s not reinvent the wheel!

- Leverage existing technologies based on the syntax desired
- Approve specific transport options for each syntax

- **Examples:**
  - XML  ➔ SOAP
  - Plaintext  ➔ Syslog
Deconstruction of Traditional Logs

CLS (Plaintext)

<37> Sep 26 12:17:32 myhost-- root[808]: ROOT LOGIN ON tty1

logTime logSrcHostname procName procId acctName action ttyName

CEET

CLS (Plaintext)

CLT (syslog)
Putting It Together

Event

Event Attributes

Event Record

CELR

Base Profile

Device Profiles

CLS

Text

Binary

CEET

OAS

Taxonomy

Coding

Best Practices

Approved for Public Release 09-4036
CEE & Event Management Automation Protocol (EMAP)

- NIST Research Effort

- Extend concepts of SCAP to automate the event management space

- CEE is a critical foundation for EMAP

- Need standard way to know:
  - Required information will be present
  - Events in standardized format to aid tool consumption
CEE & EMAP – Automating an OODA Loop

- Observe – Meaningful Logs, Reports, and News
- Orient – Looking for Events of Possible Interest
- Decide – Determine Good, Bad, Unknown, Watch, Ignore…
- Act – Block or Allow? Refine Rules or Policy?
- Feedback – Alter CEE configuration?
CEE & EMAP Validation

- Validate log compliance to a CELR Profile
  - Not necessarily the same one used to configure logs
## Upcoming Timeline

<table>
<thead>
<tr>
<th>Task Summary</th>
<th>Target Date (CY)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Draft Specification</td>
<td>Q3 2009</td>
</tr>
<tr>
<td>XML and Text CLS Support</td>
<td>Q4 2009</td>
</tr>
<tr>
<td>Firewall and IDS CELR</td>
<td>Q1 2010</td>
</tr>
<tr>
<td>Final CEE Draft 1.0 (CEE Specification)</td>
<td>Q1 2010</td>
</tr>
<tr>
<td>Initial CLT Support</td>
<td>Q2 2010</td>
</tr>
<tr>
<td>Initial Public Repository for CEET and CELR Data</td>
<td>Q2 2010</td>
</tr>
</tbody>
</table>

Vendor / device support of CEE possible at end of Q2 2010
More Information Available

- CEE Website: http://cee.mitre.org/
- CEE Working Group Mailing List: http://cee.mitre.org/discussiongroup.html
Questions?

“Those who cannot remember the past are condemned to repeat it.”

– George Santayana