# The June CDM Webinar
# We will begin at 12:00PM EDT

## Welcome to the CDM Webinar – Automating Software Asset Management

## While you wait, check out:

**Continuous Diagnostics and Mitigation (CDM)**

Welcome to the DHS Continuous Diagnostics and Mitigation (CDM) Training Program website.

The information on this website is intended for government cybersecurity professionals who are participating in the DHS CDM Program and for cybersecurity professionals who would like more information on implementing a continuous monitoring program.

The CDM Training section provides information on upcoming workshops and webinars.

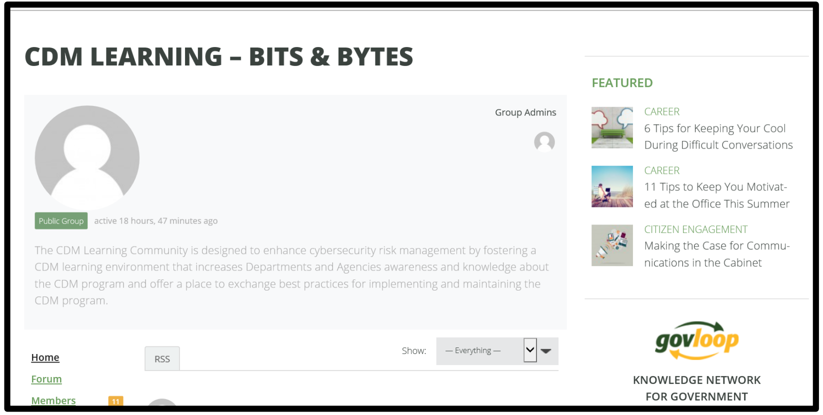
The CDM Guides section provides readiness and implementation training information.

The CDM FAQ section provides a list of questions and answers regarding the CDM Program.

The CDM Resources section provides a list of documents and other helpful resources related to the CDM Program.

**Our CDM Homepage**

https://www.us-cert.gov/cdm/training


Group Admins
Public Group
active 18 hours, 47 minutes ago
Home
Forum
Members
11
RSS
Show:
— Everything —
FEATURED
CAREER
6 Tips for Keeping Your Cool During Difficult Conversations
CAREER
11 Tips to Keep You Motivated at the Office This Summer
CITIZEN ENGAGEMENT
Making the Case for Communications in the Cabinet
govloop
KNOWLEDGE NETWORK FOR GOVERNMENT

**CDM LEARNING – BITS & BYTES**

The CDM Learning Community is designed to enhance cybersecurity risk management by fostering a CDM learning environment that increases Departments and Agencies awareness and knowledge about the CDM program and offer a place to exchange best practices for implementing and maintaining the CDM program.

**Our CDM Bits and Bytes Blog**

https://www.govloop.com/groups/cdm-learning-bits-bytes/

*Have a topic suggestion for a future event or blog post? Want to join our membership list? Please reach out to cdmlearning@hq.dhs.gov*

# Automating Software Asset Management

June 16, 2016
12:00 pm – 1:00 pm

A CDM Learning Webinar

END
START
START
END
MANAGE EVENTS
MANAGE SECURITY LIFECYCLE
END
START
MANAGE ACCOUNTS FOR PEOPLE AND SERVICES
MANAGE ASSETS
START
END

# Event Goal

The goal is to discuss the automation of Software Asset Management (SWAM), focusing on:

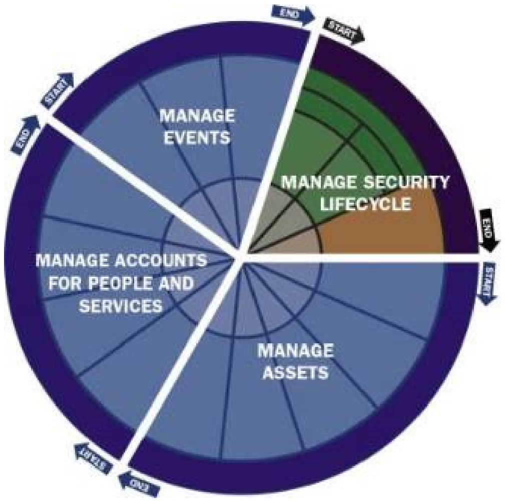
- the NCCoE building block supporting the SWAM cybersecurity capability
- creating a SWID environment

# CDM and Software Asset Management

## Software Asset Management

- What steps can you take to promote development of a SWID environment?
- Who are the stakeholders?
- What in-house (custom) applications need SWID tags?
- Do you have a “whitelist” – lists of allowed products, with all others prohibited?
- Do you have a “blacklist” - lists of prohibited products, with all unlisted products implicitly allowed

CDM SWAM
Security
Capability


MANAGE EVENTS
MANAGE SECURITY LIFECYCLE
MANAGE ACCOUNTS FOR PEOPLE AND SERVICES
MANAGE ASSETS
START
END
START
END
START
END
START
END

# Today’s Speaker: David Waltermire

**Security Automation Architect, National Institute of Standards and Technology (NIST) Computer Security Division (CSD)**

- Leads development of the Building Block for Continuous Monitoring and Software Asset Management.
- Leads research at NIST on the Security Content Automation Protocol (SCAP) which includes CPEs, CVEs (and CVSS scores), and CCE

# Use of SWID Tags for Continuous Monitoring of IT Assets

David Waltermire

National Institute of Standards and Technology

NIST
National Institute of
Standards and Technology
U.S. Department of Commerce

# Agenda

- Software Management Challenges
- Software Identification (SWID) Tag Concepts
- Role of Trusted Network Communications (TNC) Standards
- Ongoing IETF Standards Work

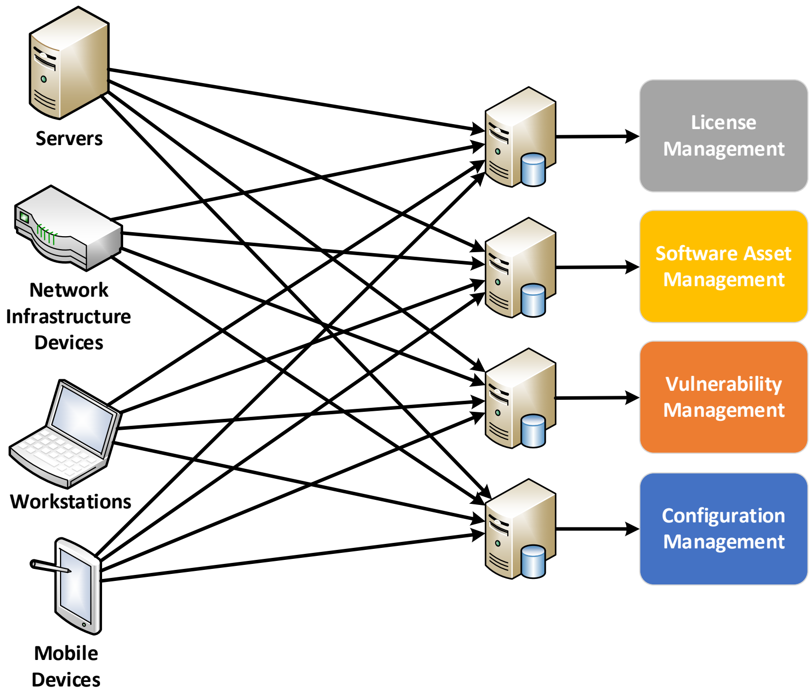
# Background

- On September 16, 2015 NIST published the Software Asset Management Building Block[1]
- This building block identified the following capabilities:
  - Capability 0 – Establish SWID Tag Environment
  - Capability 1 – Publish Installed SWID Tag Data
  - Capability 2 – Media Verification Using SWID Tags
  - Capability 3 – Execution Authorization Using Installed SWID Data
  - Capability 4 – Network-Based Policy Enforcement Based on SWID Information
- Work to-date has focused on Capabilities 0 through 2 by developing supporting standards and guidance

[1] https://nccoe.nist.gov/projects/building_blocks/software_asset_management

# The Software Management Challenge

# The Tower of Babel



- Inconsistent software information collection methods
  - Different identifiers for the same installed software
  - Data cannot be cross-correlated
  - Redundant data collection
  - Extra load on devices
  - Increased attack surface
- Automation limited to a specific tool and/or platform
- **Common software identifiers are needed to unify business processes across platforms**

# Endpoints are Unpatched

Organizations need to identify unpatched, vulnerable software on computing devices

- 99.9% of exploited vulnerabilities were compromised more than a year after they were disclosed

Source: Verizon Enterprise, 2015 Data Breach Investigations Report

- 58% of businesses don’t have “mature” patch management processes

Source: Trustwave, 2014 State of Risk Report

**Software identification and characterization data is need for vulnerability-to-software mappings**

# Correlating Shared Cybersecurity Information

Collection, analysis, sharing, and use of cybersecurity information needs to occur in cyber-relevant time

- 75% of attacks spread from Victim 0 to Victim 1 within one day, with spread to a second organization in 1 hour

  Source: Verizon Enterprise, 2015 Data Breach Investigations Report

**Standardized software identifiers are needed to support rapid correlation of shared software-related information**

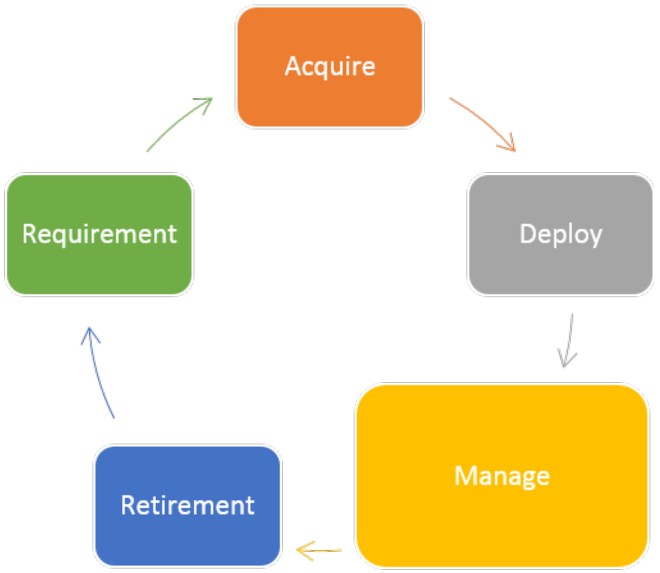
- **Vulnerability Bulletins and Alerts**
- **Shared Courses of Action (CoA)**
- **Configuration Baselines**

# The Common Platform Enumeration (CPE)

- Used to identify 11 metadata attributes for software (i.e., vendor, product name, version, update, edition)
  - Part of the Security Content Automation Protocol (SCAP)
  - No support for software patches
- **Untimely:** CPE Names are often created when a vulnerability is found in a software application
- **Centralized:** Most CPE Names are created by the National Vulnerability Database analysts

What about license, configuration, and software inventory use cases that need an identifier earlier?

# Typically Software is Identified while Managing



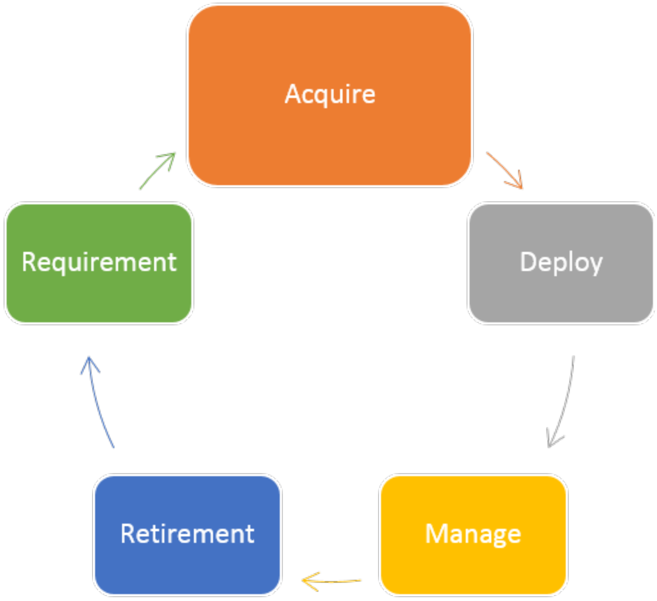
Identifiers used for:

- Software Asset Management
- Vulnerability Management
- Configuration Management

- Software is identified by 3rd-party tools and services supporting these management functions

- Often tools identity the same software differently

# Software Needs to be Identified by the Software Provider



- Software must be identified at the point of software publication
- Use the same software identity for all business processes to:
  - Integrate business processes
  - Correlate data provided by tools

# Key Security Questions to Address

Security automation solutions must help organizations to know:

- What endpoints are connected to the network?
- **What software versions and patches are deployed on a given endpoint?**
- **Has an important change in the software load or configuration occurred?**
- Is this software authorized for use?
- Is this software properly configured?
- What implication does this have for the observed behavior of the endpoint?

# Software Identification (SWID) Tags

Based on ISO/IEC 19770-2:2015

# What’s a SWID Tag?

## Problem Statement

- Software is critical infrastructure
- Software Asset Management (SAM) is a critical business function
  - control and protection of software and related assets
  - control and protection of information needed to control and protect software assets
- Critical challenges:
  - Installation media are hard to verify
  - Once installed, software is hard to *discover*, *identify*, and *characterize*
    - Patches and upgrades too!
  - Relationships among software applications, components, libraries, files, and other system resources are unclear

# What’s a SWID Tag?

## A minimal example

**An XML Document – Conforms to ISO/IEC 19770-2:2015**

```
<SoftwareIdentity
  name="ACME Roadrunner Detector 2013 Coyote Edition"
  tagId="com.acme.rrd2013-ce-sp1-v4-1-5-0" version="4.1.5">

  <Entity name="The ACME Corporation"
          regid="acme.com" role="tagCreator"/>
</SoftwareIdentity>
```

**Most elements and attributes are optional**

# What’s a SWID Tag?

## A more complete example

### Created by an authoritative source

```
<SoftwareIdentity
  name="ACME Roadrunner Detector 2013 Coyote Edition"
  tagId="com.acme.rrd2013-ce-sp1-v4-1-5-0" version="4.1.5">

  <Entity name="The ACME Corporation"
          regid="acme.com" role="tagCreator softwareCreator"/>
  <Entity name="Coyote Services, Inc."
          regid="mycoyote.com" role="distributor"/>

  <Link rel="license" href="www.gnu.org/licenses/gpl.txt/">

  <Meta activationStatus="trial" product="Roadrunner Detector"
        colloquialVersion="2013" edition="coyote"/>

  <Payload>
    <Directory root="%programdata%" location="rrdetector">
      <File name="rrdetector.exe" size="532712"
          SHA256:hash="a314fc2dc663ae7a6b6bc6787594057396e6b3f56
                                  9cd50fd5ddb4d1bbafd2b6a"/>
    </Directory>
  </Payload>
</SoftwareIdentity>
```

# What’s a SWID Tag?

## Anatomy

**Unique identification of the software**

```
<SoftwareIdentity
  name="ACME Roadrunner Detector 2013 Coyote Edition"
  tagId="com.acme.rrd2013-ce-sp1-v4-1-5-0" version="4.1.5">

  <Entity name="The ACME Corporation"
          regid="acme.com" role="tagCreator softwareCreator"/>
  <Entity name="Coyote Services, Inc."
          regid="mycoyote.com" role="distributor"/>

  <Link rel="license" href="www.gnu.org/licenses/gpl.txt/">

  <Meta activationStatus="trial" product="Roadrunner Detector"
        colloquialVersion="2013" edition="coyote"/>

  <Payload>
    <Directory root="%programdata%" location="rrdetector">
      <File name="rrdetector.exe" size="532712"
          SHA256:hash="a314fc2dc663ae7a6b6bc6787594057396e6b3f56
                          9cd50fd5ddb4d1bbafd2b6a"/>
    </Directory>
  </Payload>
</SoftwareIdentity>
```

# What’s a SWID Tag?

## Anatomy

### Association with software and tagging roles

```
<SoftwareIdentity
  name="ACME Roadrunner Detector 2013 Coyote Edition"
  tagId="com.acme.rrd2013-ce-sp1-v4-1-5-0" version="4.1.5">

  <Entity name="The ACME Corporation"
          regid="acme.com" role="tagCreator softwareCreator"/>
  <Entity name="Coyote Services, Inc."
          regid="mycoyote.com" role="distributor"/>

  <Link rel="license" href="www.gnu.org/licenses/gpl.txt/">

  <Meta activationStatus="trial" product="Roadrunner Detector"
        colloquialVersion="2013" edition="coyote"/>

  <Payload>
    <Directory root="%programdata%" location="rrdetector">
      <File name="rrdetector.exe" size="532712"
          SHA256:hash="a314fc2dc663ae7a6b6bc6787594057396e6b3f56
                          9cd50fd5ddb4d1bbafd2b6a"/>
    </Directory>
  </Payload>
</SoftwareIdentity>
```

# What’s a SWID Tag?

## Anatomy

### Relating information

```
<SoftwareIdentity
  name="ACME Roadrunner Detector 2013 Coyote Edition"
  tagId="com.acme.rrd2013-ce-sp1-v4-1-5-0" version="4.1.5">

  <Entity name="The ACME Corporation"
          regid="acme.com" role="tagCreator softwareCreator"/>
  <Entity name="Coyote Services, Inc."
          regid="mycoyote.com" role="distributor"/>

  <Link rel="license" href="www.gnu.org/licenses/gpl.txt/">

  <Meta activationStatus="trial" product="Roadrunner Detector"
        colloquialVersion="2013" edition="coyote"/>

  <Payload>
    <Directory root="%programdata%" location="rrdetector">
      <File name="rrdetector.exe" size="532712"
          SHA256:hash="a314fc2dc663ae7a6b6bc6787594057396e6b3f56
                          9cd50fd5ddb4d1bbafd2b6a"/>
    </Directory>
  </Payload>
</SoftwareIdentity>
```

# What’s a SWID Tag?

## Anatomy

### Expressing descriptive metadata

```
<SoftwareIdentity
  name="ACME Roadrunner Detector 2013 Coyote Edition"
  tagId="com.acme.rrd2013-ce-sp1-v4-1-5-0" version="4.1.5">

  <Entity name="The ACME Corporation"
          regid="acme.com" role="tagCreator softwareCreator"/>
  <Entity name="Coyote Services, Inc."
          regid="mycoyote.com" role="distributor"/>

  <Link rel="license" href="www.gnu.org/licenses/gpl.txt/">

  <Meta activationStatus="trial" product="Roadrunner Detector"
        colloquialVersion="2013" edition="coyote"/>

  <Payload>
    <Directory root="%programdata%" location="rrdetector">
      <File name="rrdetector.exe" size="532712"
          SHA256:hash="a314fc2dc663ae7a6b6bc6787594057396e6b3f56
                          9cd50fd5ddb4d1bbafd2b6a"/>
    </Directory>
  </Payload>
</SoftwareIdentity>
```

# What’s a SWID Tag?

## Anatomy

### Characterizing the software’s composition

```
<SoftwareIdentity
  name="ACME Roadrunner Detector 2013 Coyote Edition"
  tagId="com.acme.rrd2013-ce-sp1-v4-1-5-0" version="4.1.5">

  <Entity name="The ACME Corporation"
          regid="acme.com" role="tagCreator softwareCreator"/>
  <Entity name="Coyote Services, Inc."
          regid="mycoyote.com" role="distributor"/>

  <Link rel="license" href="www.gnu.org/licenses/gpl.txt/">

  <Meta activationStatus="trial" product="Roadrunner Detector"
        colloquialVersion="2013" edition="coyote"/>

  <Payload>
    <Directory root="%programdata%" location="rrdetector">
      <File name="rrdetector.exe" size="532712"
          SHA256:hash="a314fc2dc663ae7a6b6bc6787594057396e6b3f56
                          9cd50fd5ddb4d1bbafd2b6a"/>
    </Directory>
  </Payload>
</SoftwareIdentity>
```

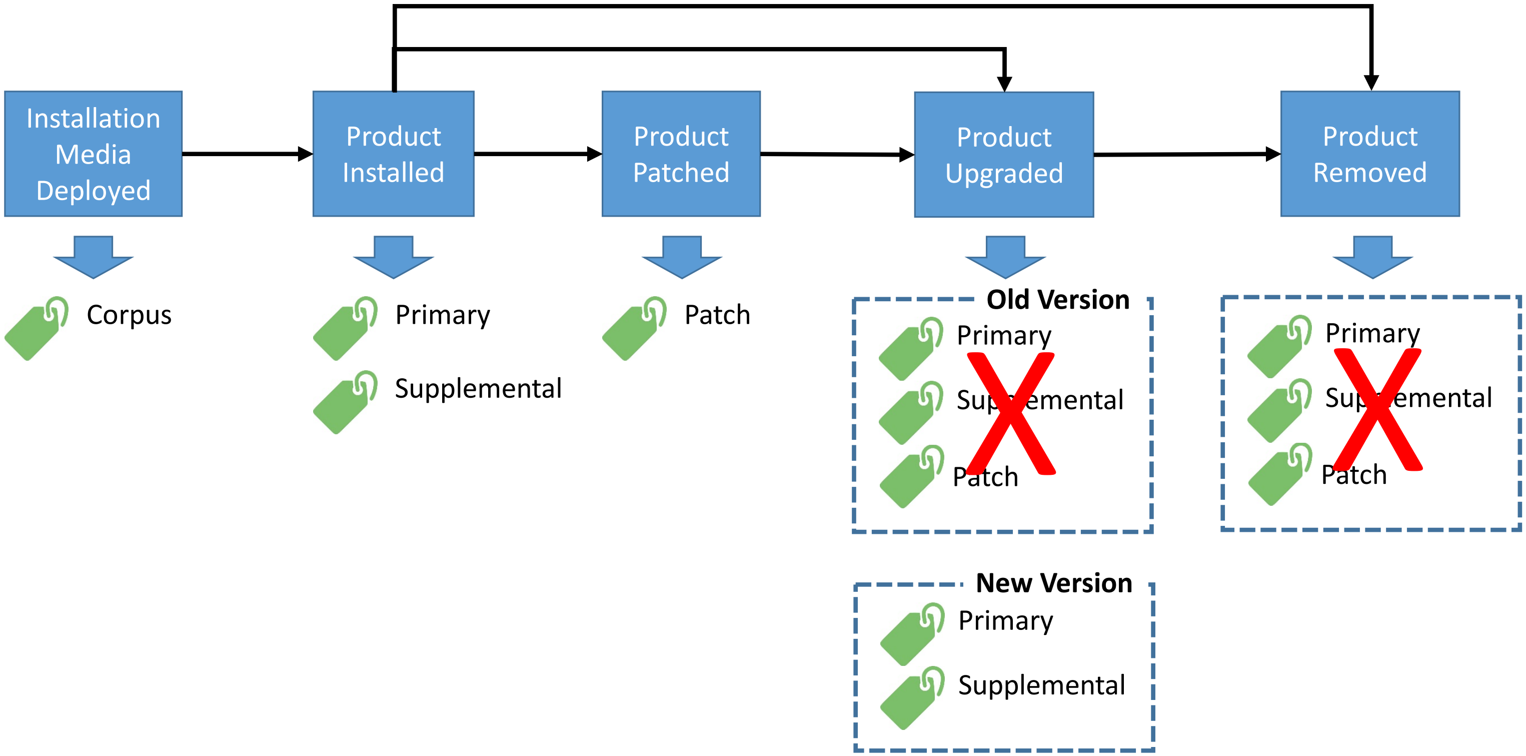
# Types of SWID Tags

- Corpus – Describes the contents of an installation package or media
- Primary – Describes a base software install
- Patch – Describes an incremental software update
- Supplemental – Provides extensible metadata about another tag

- Relationships associate tags
  - Patch -> Primary
  - Supplemental -> Corpus, Primary, or Patch

# What’s a SWID Tag?

## Use in the Software Lifecycle

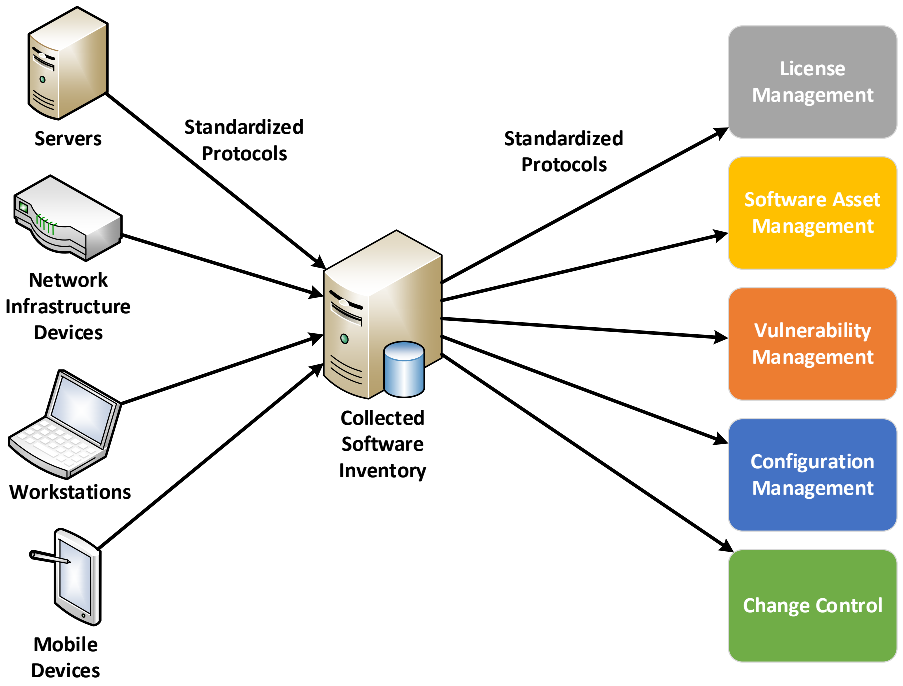


# SWID Tags Integrate Process Verticals

| Release | Install | Maintain | Uninstall |
|---|---|---|---|
| Publisher releases software | Organization installs new software | Installation of patches and updates | Organization retires software |

**SWID tags enable:**

- High-fidelity software metadata provided by vendors
- Platform-neutral, standardized software inventory
- Integration of data and process verticals
- Automation and innovation supporting risk-based management of software



# Guidance on the Creation of Interoperable SWID Tags

NIST published NISTIR 8060: *Guidelines for the Creation of Interoperable Software Identification (SWID) Tags in April 2016*

- *Sections 2 and 3 provide an overview of SWID tags and their operational use*
- *Sections 4 and 5 provide* guidelines that supplement the ISO/IEC SWID tag specification to support cybersecurity use cases
- Section 6 describes cybersecurity usage scenarios for SWID tags

# Uses of SWID Tag Data

- **Package Verification:** Verification of installation packages using Corpus tags improving software assurance
- **Software Inventory:** Reporting of software inventory using the tagId from Primary and Patch tags
- **Software Integrity:** Verify software and patch installs using Payload information from Primary and Patch Tags
    - Libraries, scripts, and executables can be checked for unauthorized changes at runtime
    - Filesystems can be monitored for changes to key applications
- **Digital Policy Definition:** Tag metadata can be used for endpoint protection policies
    - Installation and execution whitelists/blacklists

# The Chicken and the Egg

- Software providers want clear demand signals from customers desiring SWID tags
- Tool providers want more available SWID tags in use

- Some major software providers and tool vendors are already adopting
- More is needed!
- Software discovery tools can fill the gaps

- Consumer demand is key to accelerating wider adoption

# What NIST is Doing to Increase SWID Tag Adoption

- **Guidance:** Providing guidlines for SWID tag creators
- **Standardization:** Working with standards organizations to integrate the use of SWID tags into management and security standards
- **Education:** Informing software providers, tool vendors, and end-users
- **Integration with other NIST efforts:**
  - Inclusion in SCAP
    - SCAP 1.2 tools can look for SWID tags as installation evidence using OVAL
    - SCAP 1.3 adds SWID tags as a component specification describing the OVAL approach
    - SCAP 2.0 will provide native support for SWID tags as a replacement for CPE
  - Integration into the National Vulnerability Database

# What You Can Do to Increase SWID Tag Adoption

- **Increase Demand:** Encourage software providers to include SWID tags with their software as a competitive differentiator
- **Require Tagged Patches:** Ask software providers to include SWID tags with patches
- **Adopt SWID Tags for Custom Software:** In-house developed applications can be managed using SWID tags in the same way as commercial applications
- **Require Compliance with NISTIR 8060:** The guidelines in NISTIR 8060 ensure that SWID tags provide the data needed to manage software for security and other use cases
- **Require use of SWID tags in Tools:** SWID tags provide valuable, authoritative identifiers and metadata data

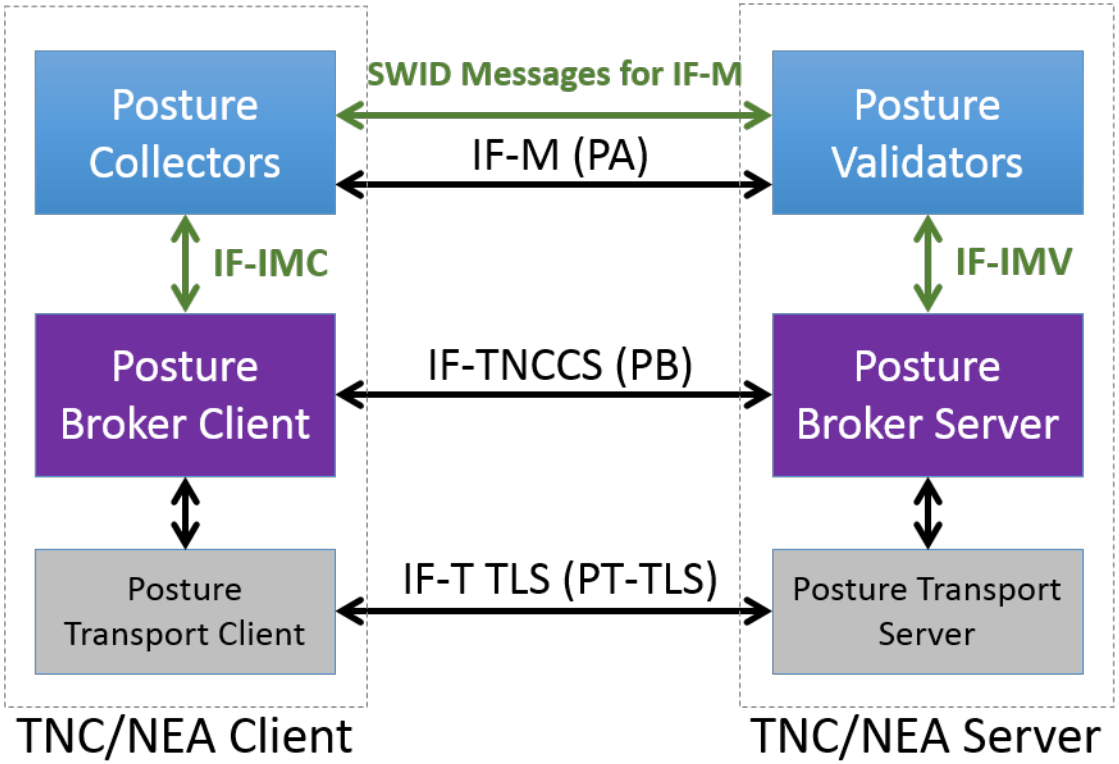
# The Trusted Network Communications (TNC) Standards

Developed by the Trusted Computing Group (TCG)

# The Trusted Network Communication Standards

- Provide ongoing awareness over the constantly changing state of endpoints
- Detect endpoint software changes in cyber-relevant time
- Enable creation and use of shared information within organizations:
  - Support multiple operational and security processes using a single point of data collection
  - Inform courses of action (e.g., patch, configure, block)
  - Identify indicators of compromise – Find and prevent malicious software from executing
- Define standard protocols and data formats for architectural components
  - Leverage existing standards where possible

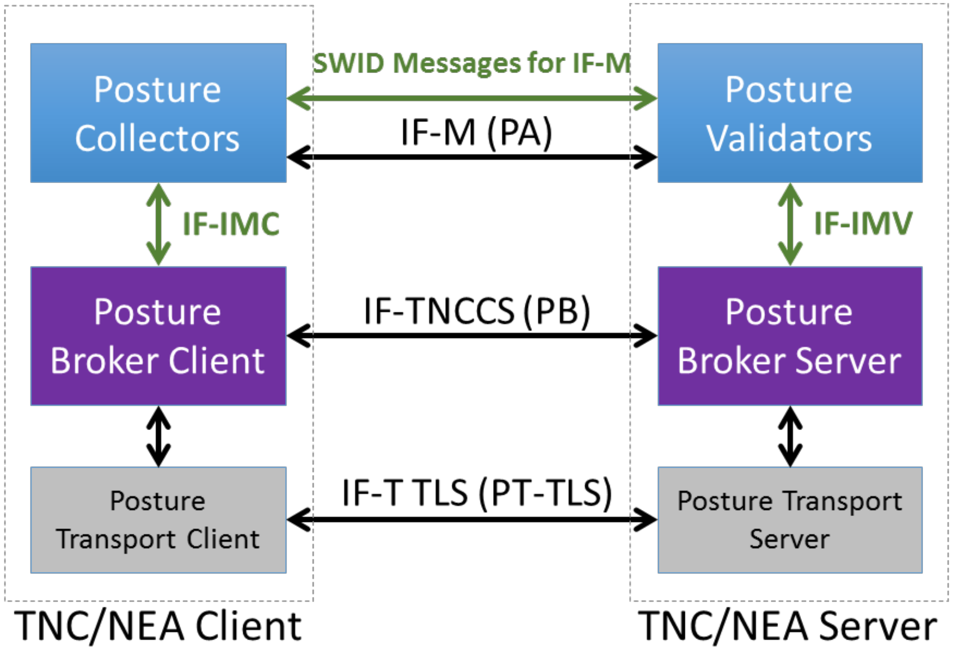
# Building on Existing Standards: The TCG TNC and IETF NEA Architectures



Note: Equivalent specifications are published by the TCG and the IETF.

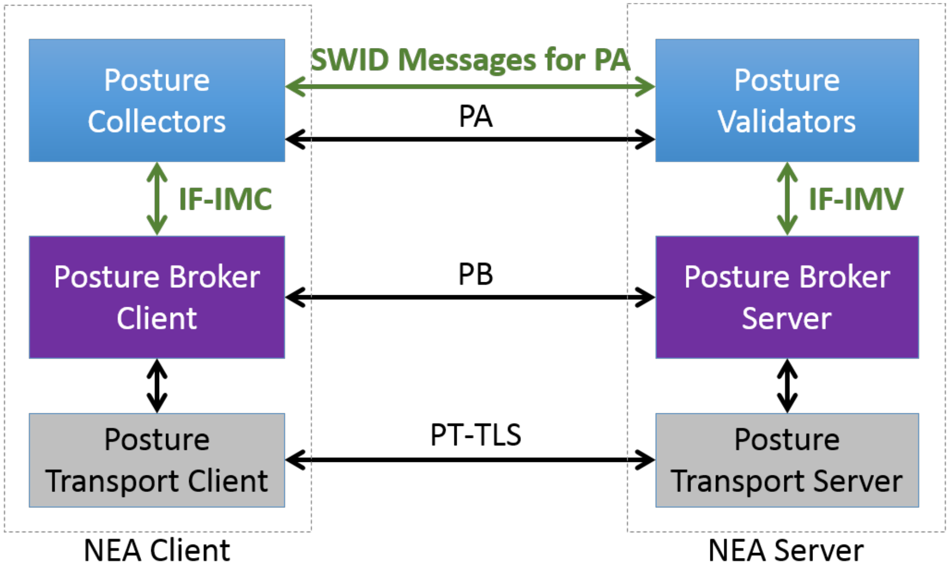
# SWID Messages and Attributes for IF-M

- Supports the collection of software inventory using SWID data
- Allows full and delta software inventories
- Supports subscriptions to monitor an endpoint’s software inventory
    - Detects updates to SWID tags on an endpoint and updates server
    - Software changes can be reported as they occur



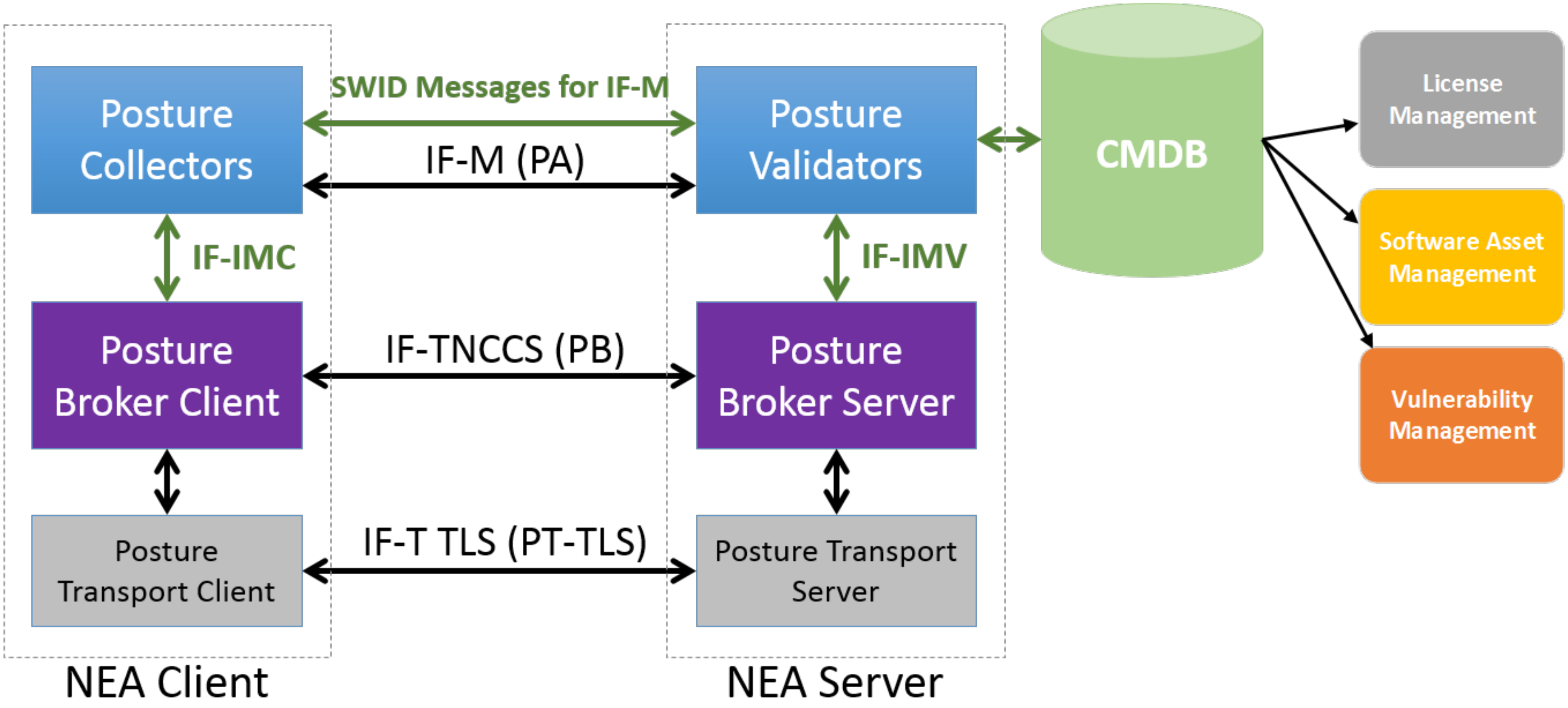
# Use of IF-IMC & IF-IMV

- IF-IMC and IF-IMV provide pluggable interfaces
- Enable plugins for collection and evaluation
- Plugins support different kinds of posture information



# The Enterprise Compliance Profile Supports SWAM and VUL

SWID Messages supports SWAM and software inventory for VUL.



See: http://www.trustedcomputinggroup.org/resources/tnc_endpoint_compliance_profile_specification

# What You Can Do to Increase TNC, NEA, and SWID Messages Adoption

In addition to adoption of SWID tags:

- **Increase Demand:** Encourage security tool vendors to implement the Enterprise Compliance Profile
- **Encourage Adoption of SWID Messages:** For existing tools that support the TNC standards, request support for SWID Messages and Attributes for IF-M
- **Encourage Adoption of IF-T TLS:** For existing tools that support IF-T EAP, request support for IF-T TLS

# Ongoing Standards Work in the Internet Engineering Task Force (IETF)

The Security Automation and Continuous Monitoring (SACM) Working Group

# Current Focus: Software Inventory and Vulnerability Assessment

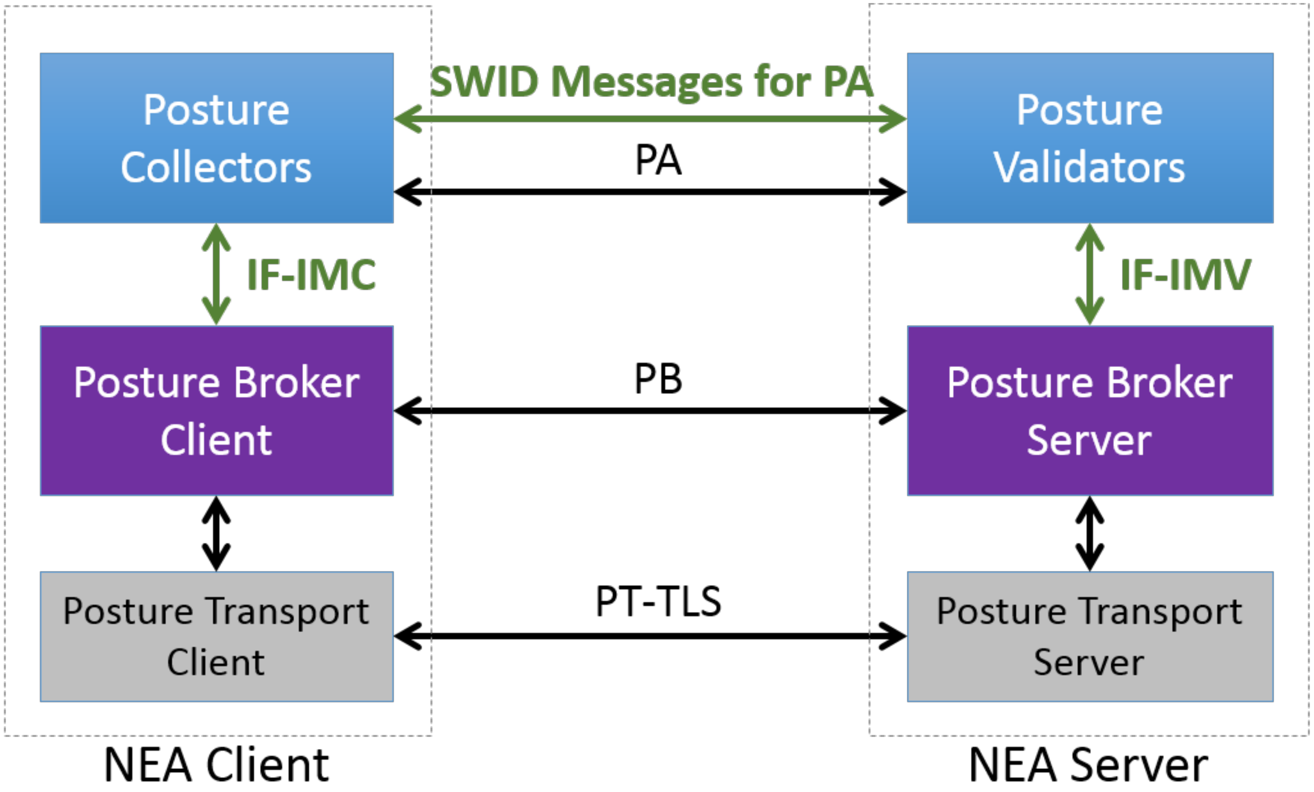
- Mechanisms to support online collection of endpoint software inventory
- Supports management of software patches and updates

Needed capabilities:

- Endpoint Identification
- Ongoing exchange of software inventory, open ports, enabled services
- Use of vulnerability alerts to determine vulnerable endpoints based on software load

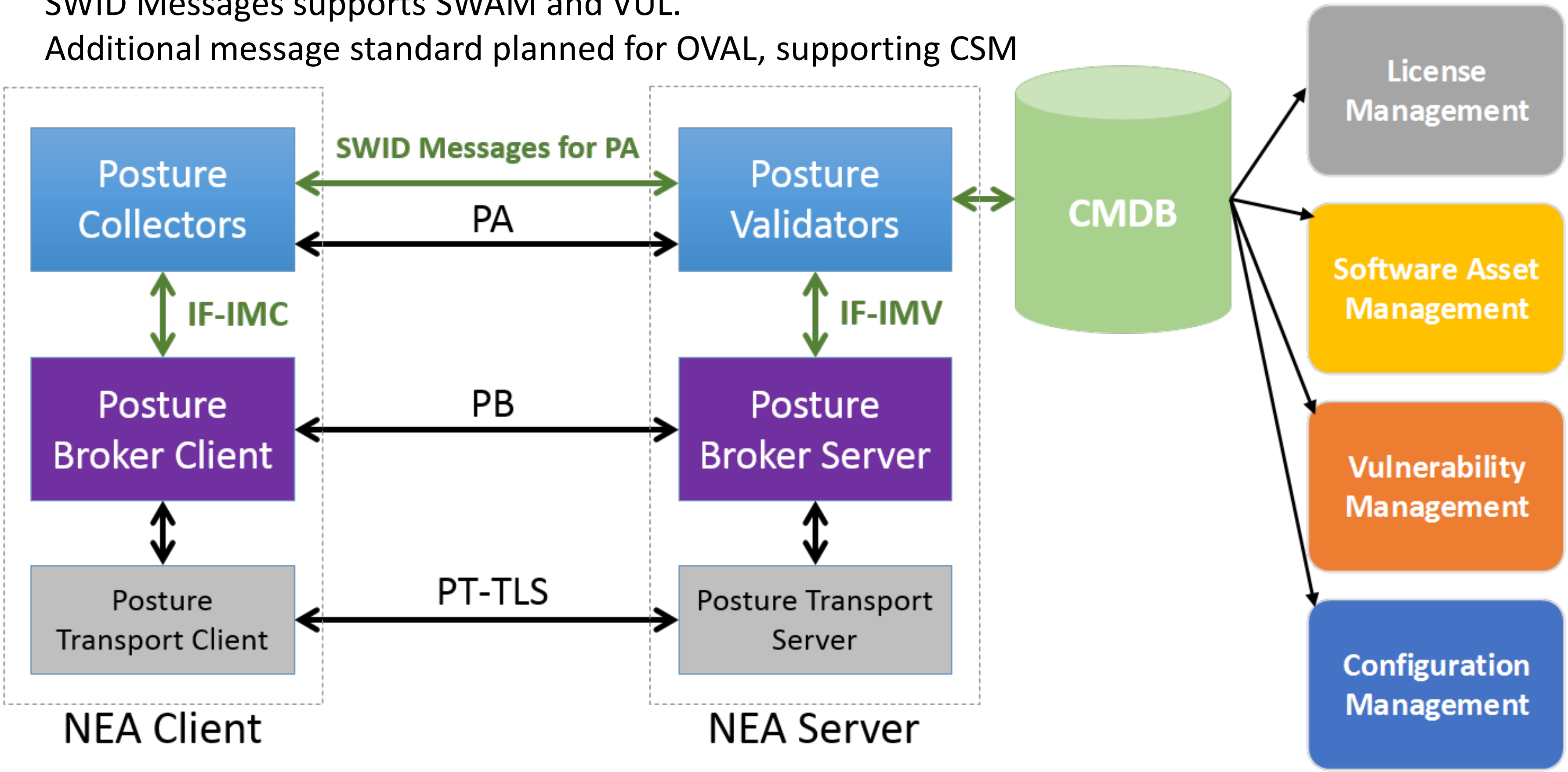
Future focus on Configuration Management and automating Courses of Action (CoA).

# Enhancing Trusted Computing Group (TCG) Specifications



# Use for SWAM, CSM, and VUL

SWID Messages supports SWAM and VUL.
Additional message standard planned for OVAL, supporting CSM



# Next Steps

- SCAP 1.3 Update (In progress) – FY2016
- IETF SWID Messages RFC – Early FY2017
- SCAP 2.0 Update – Planned for FY2018

**Get involved:**

IETF SACM Working Group: https://datatracker.ietf.org/wg/sacm/

Subscribe to the email list:

https://www.ietf.org/mailman/listinfo/sacm

# Summary

- Software identification and characterization information is needed to support patch, vulnerability, configuration, and license management as well as other operational activities relating to software
- SWID tags provide a standardized format for exchanging software identification and characterization information
- NISTIR 8060 provide guidance on the creation of SWID tags suitable to address cybersecurity use cases
- The IETF NEA protocols provide a foundation for extensible endpoint assessment
- SWID Messages and Attributes for PA-TNC can be used to exchange software inventory information supporting SWAM, and enabling VULN, and CSM

# Questions and Discussion

**David Waltermire**

david.waltermire@nist.gov

Computer Security Division

Information Technology Laboratory

National Institute of Standards and Technology

# CDM, SWAM, and SWIDs

## CDM, SWAM, and SWIDs

- How can your current tools and processes support a SWID tag environment?
- Ask your CMaaS integrator how the tools they are implementing support a SWID tag environment? If necessary, work together to request this from the vendor.
- Think about the CDM “desired state” – what you want on your network (whitelist).
- Think about how the sensor tools and a SWID tag environment can be utilized so that only authorized software is on your system.

# Questions and Answers

*What is the impact of …??*

*What about ….??*

How did the stakeholders adjust to ….??

*What would you do differently about ….??*

What should I do about ….??

*What would you recommend for ….??*

*How much time did it take to ….??*

*How are you maintaining 95% compliance with SWAM CAP….??*

*How did you handle … (latency, centralization, control issues, etc.) ??*

# Questions and Answers

**Doesn't the CPE's software version identify the patch level?**

They do, in a simplified way. Providing a single patch level is not sufficient to identify all possible patches that a software might have. Not every vendor expresses the patch level in their CPEs. A handful of patches could be applied to the same software, and with the current capability of CPE there is no way to differentiate which software is installed. This is critical for identifying vulnerabilities, and is not yet possible.

**Can you speak to whether commercial software vendors have embraced SWID tags?**

A few software providers have publically expressed their support for and adoption of SWID tags. With the 2015 version released in August 2015, Some vendors are working on implementation. Organizations who are embracing SWID tags are Microsoft, Red Hat, IBM, and others. We need your help as customers to request that vendors provide information such as SWID tags for software management.

# Questions and Answers cont’d

**Is the XML file placed in a location that cannot be easily edited?**

The SWID tags are placed on the device in the same location that the software gets installed. There is an expectation that the OS will apply the same integrity protection for the SWID tags as the software. SWID tags can be signed with XML digital signatures to detect changes in SWID tags. In the 2015 version, it is expected that SWID tags will not change and tampering would be detected. This allows for policy decision making.

**Are SWID Tags associated with the executable files or do they apply to supporting files, such as DLLs (Dynamic Link Libraries)?**

Support includes for executables as well as any associated libraries. Patches should update that information as part of the SWID tag. This allows for support of integrity checking and whitelisting applications.

# Questions and Answers cont’d

**What ensures that 2 products from the same publisher will spell the publisher name identically in their SWID tags?**

This is the reason to get authoritative information from the vendors. This is the problem we run into with CPEs. By receiving SWID tag information from the software publisher, the publisher can normalize the information to ensure the information they provide is accurate for their product. This is much more achievable and sustainable with the vendors providing the information as opposed to vulnerability researchers at the NVD.

**How does CPE change for the other domains (hardware, operating systems) if SWID tags replace ID function for software?**

It is possible that SWID tags from an authoritative source such as the software publisher could inform CPE names for software, version, etc.

# Questions and Answers cont’d

**Who are the members of the TGC? NIST? Private industry? Academia?**

The Trusted Computing Group (TGC) is an organization made up of technology industry giants, such as Microsoft, Intel, IBM, Cisco Systems, AMD, and many other companies. “The TCG was Formed to Develop, Define and Promote Open, Vendor-Neutral, Global Industry Standards, for Interoperable Trusted Computing Platforms.”

https://www.trustedcomputinggroup.org/

**Is the TCG the same group that developed the industry's TPM chip?**

Yes. The TPM (Trusted Platform Module) is a computer chip that can securely store artifacts used to authenticate your device.

https://www.trustedcomputinggroup.org/trusted-platform-module-tpm-summary/

# Questions and Answers cont'd

**Why would SWID tags be necessary if we have a strong whitelisting tool in place?**

The SWID tags would provide an authoritative source of information to support the whitelisting and software asset management tool capabilities.

**Would you provide more clarification on the patching problem. Currently all major SW vendors provide notification of release of latest version of patch and then we deploy that patch with a tool such a BigFix so what then is the problem.  Please help me understand.**

The SWID tags support the identification of software, whether its in a scanning, patching, configuration management, or other types of tools. The accurate identification of new (unmanaged) software on the network (agentless/unmonitored devices), existing software (in a tool like BigFix), and updated software is all made possible by authoritative software identification data sources like SWID tags.

# The CDM Learning Program

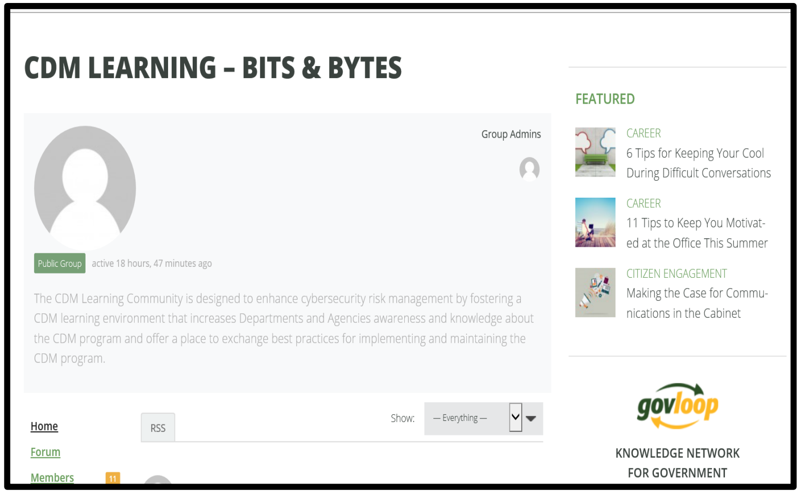
## CDM Learning Program

- Monthly Learning Community Event (CDM-LCE)
- Monthly Webinars
- Weekly CDM Bits & Bytes
- Online Vignettes

Resources Available:
https://www.us-cert.gov/cdm

Sign up to receive event information:
cdmlearning@hq.dhs.gov

Sign up for the CDM learning blog:
https://www.govloop.com/groups/cdm-learning-bits-bytes



CDM LEARNING – BITS & BYTES

Group Admins

Public Group active 18 hours, 47 minutes ago

The CDM Learning Community is designed to enhance cybersecurity risk management by fostering a CDM learning environment that increases Departments and Agencies awareness and knowledge about the CDM program and offer a place to exchange best practices for implementing and maintaining the CDM program.

FEATURED

CAREER
6 Tips for Keeping Your Cool During Difficult Conversations

CAREER
11 Tips to Keep You Motivated at the Office This Summer

CITIZEN ENGAGEMENT
Making the Case for Communications in the Cabinet

# Event Conclusion

Thank you for attending today's

CDM Webinar!

- A certificate of attendance will be available to download on the CDM Learning Program website at www.us-cert.gov/cdm/training, within one week of today's event
- Visit our website to learn more about the CDM Learning Program and upcoming events at www.us-cert.gov/cdm
- For any questions, comments, or suggestions for future topics, please email us at cdmlearning@hq.dhs.gov