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

Ongoing Security Automation Standardization Efforts

David Waltermire

National Institute of Standards and Technology

Goals

- Define standard protocols and data formats for architectural components
- Provide ongoing awareness over the constantly changing state of endpoints
- Detect endpoint changes in cyber-relevant time
- Enable information sharing within organizations:
  - Support multiple operational and security processes
  - Inform courses of action – Patch, Configure, Block
  - Identify indicators of compromise – Find and prevent malicious software from executing
- Leverage existing standards where possible

Key Questions to Address

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

Working in the IETF

The Security Automation and Continuous Monitoring (SACM) Working Group

Current Focus: Enterprise Vulnerability Assessment

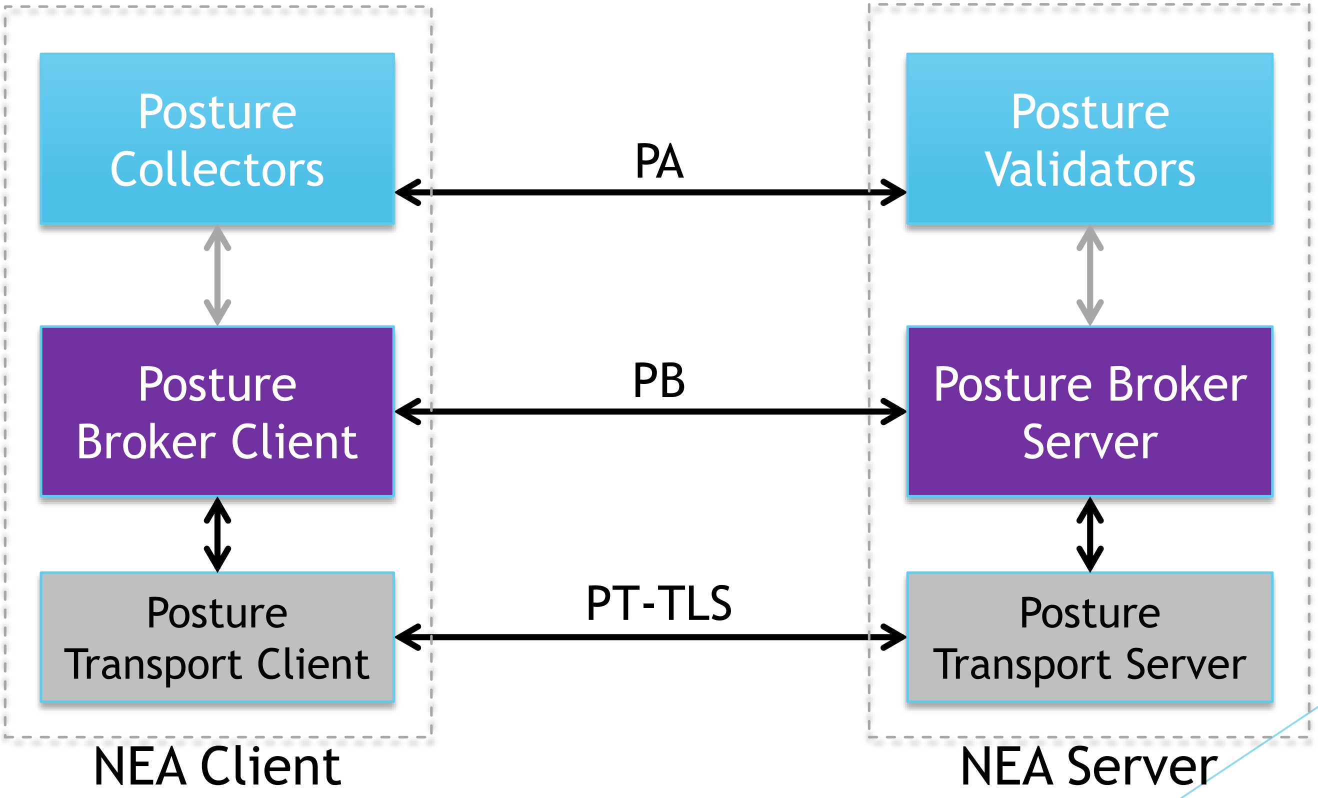
- Mechanisms to support online collection of endpoint software inventory
- Supports management of software patches and updates

Needed capabilities:

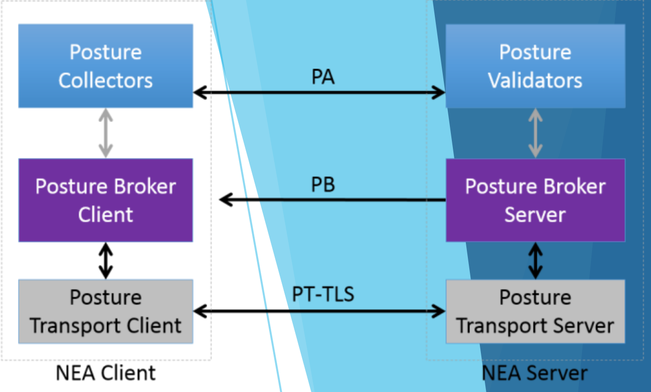
- Endpoint Identification
- Ongoing exchange of software inventory, open ports, enabled services
- Use of vulnerability alerts to determine vulnerable endpoints based on software load

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

Building on Existing Standards: The IETF NEA Architecture



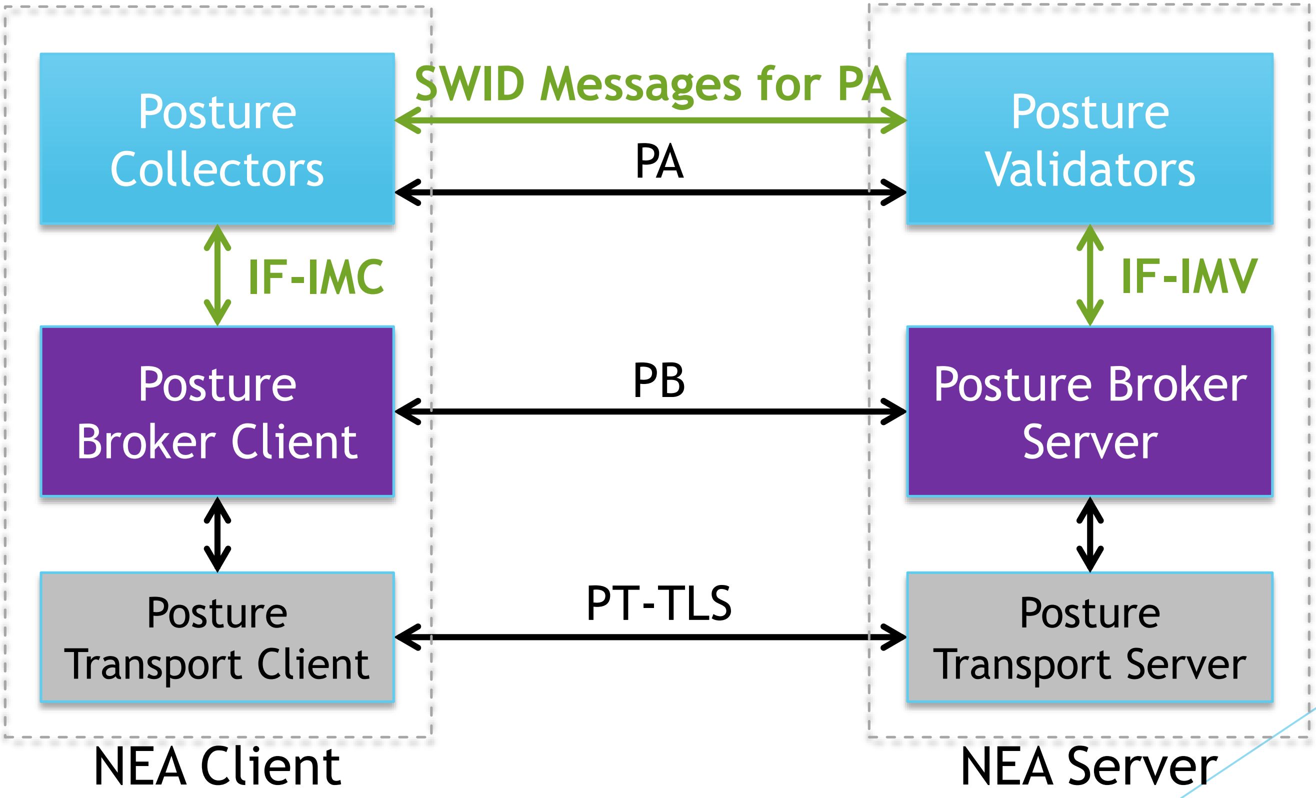
NEA Standards



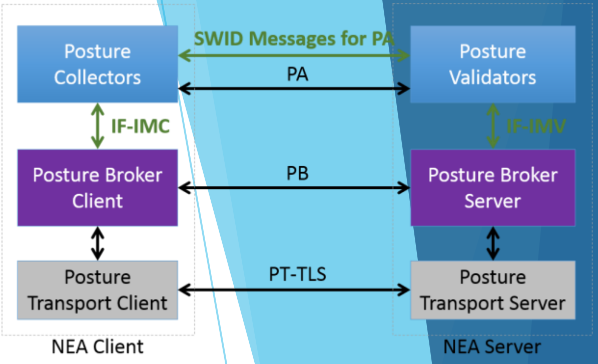
The Network Endpoint Assessment (NEA) stack includes:

- The Posture Broker (PB) protocol
  - A generalized client/server protocol to communicate endpoint posture
  - Leverages TLS for the underlying transport (PT-TLS)
- The Posture Attribute (PA) protocol
  - Supports information exchanges between collectors and validators
  - Allows extensible message types

Additional Trusted Computing Group (TCG) specifications



Use of IF-IMC & IF-IMV



IF-IMC: Standardizes how collectors are registered and communicated with

- PB Client can find and load new collectors
- PB Client can provide information to collectors so they can change their behavior

IF-IMV: Standardizes how verifiers are registered and communicated with

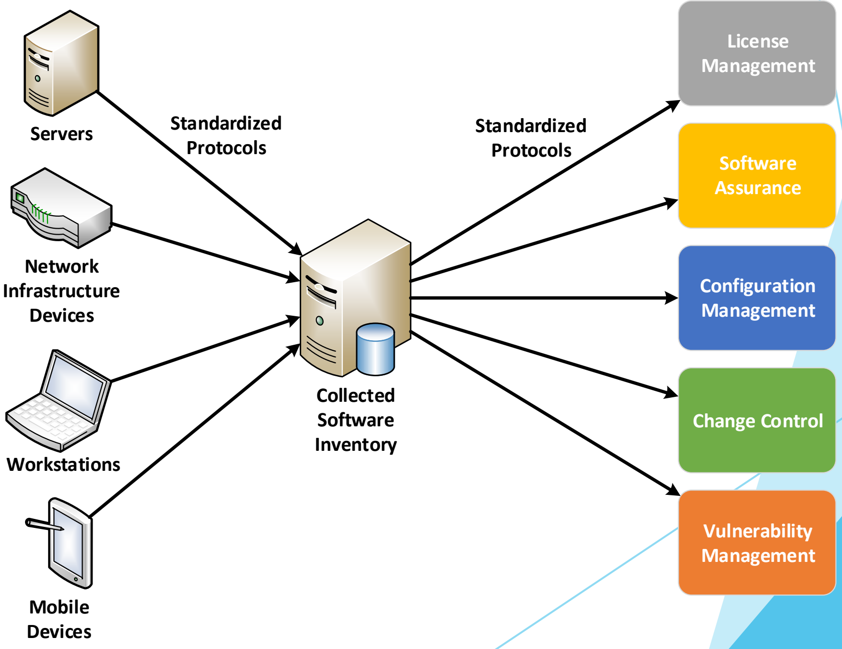
- PB Server can find and load new verifiers
- PB Server can provide information to verifiers so they can change their behavior

Use of SWID Tags

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

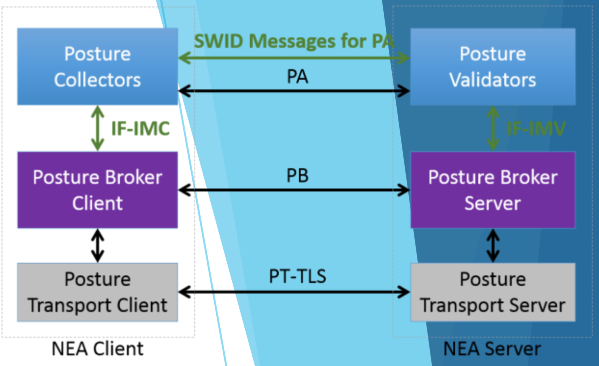


Development of NISTIR 8060 *Identification Tags*

NISTIR 8060: *Guidelines for the Creation of Interoperable Software Identification Tags*

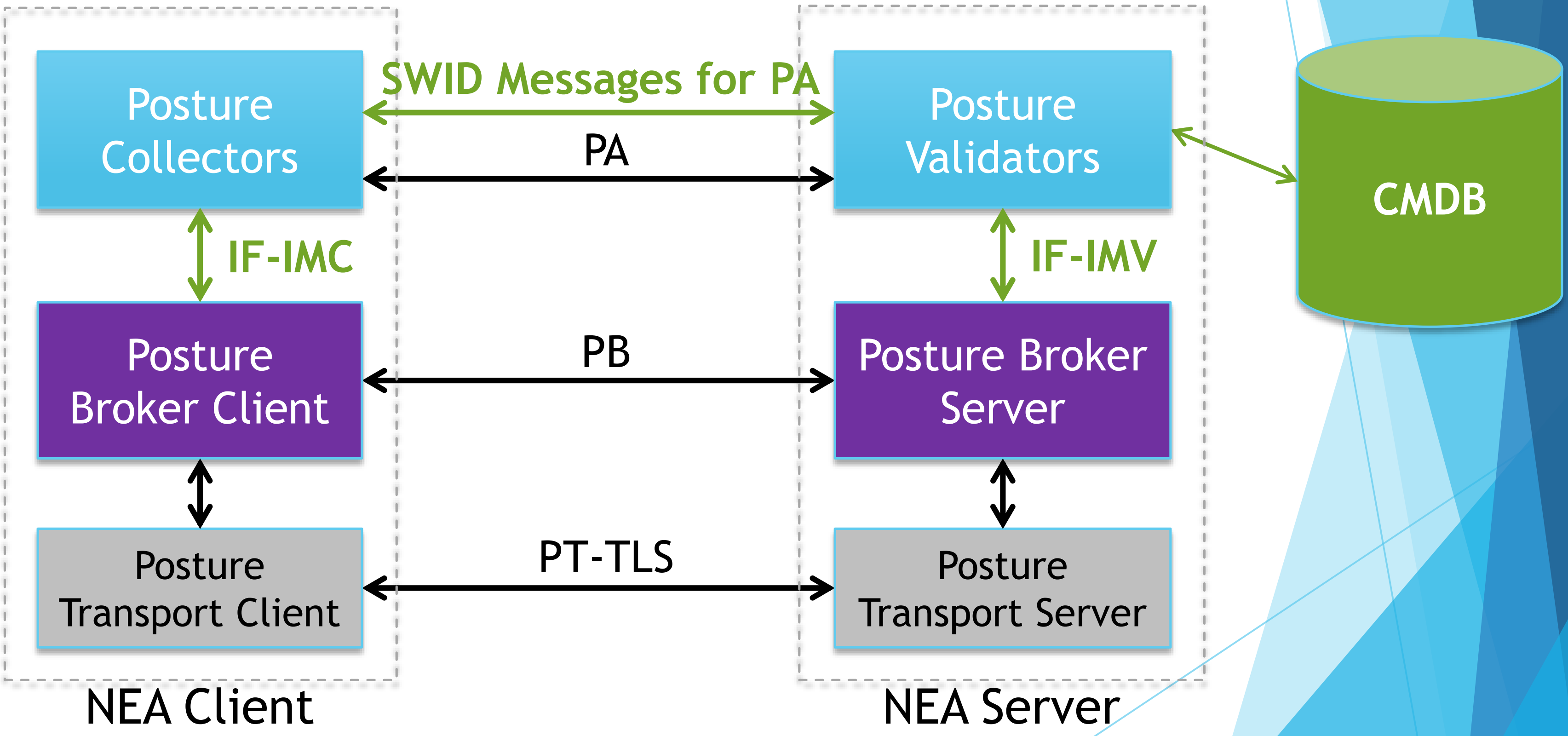
- Provides guidelines for creating SWID tags that support cybersecurity use cases
- Use case driven:
  - ✓ Continuously monitoring software inventory
  - ✓ Identifying vulnerable endpoints
  - ✓ Ensuring products are properly patched
  - ✓ Integrity measurement of installation packages and installed software
  - ✓ Preventing execution of tampered software

TCGs SWID Messages & Attributes for IF-M

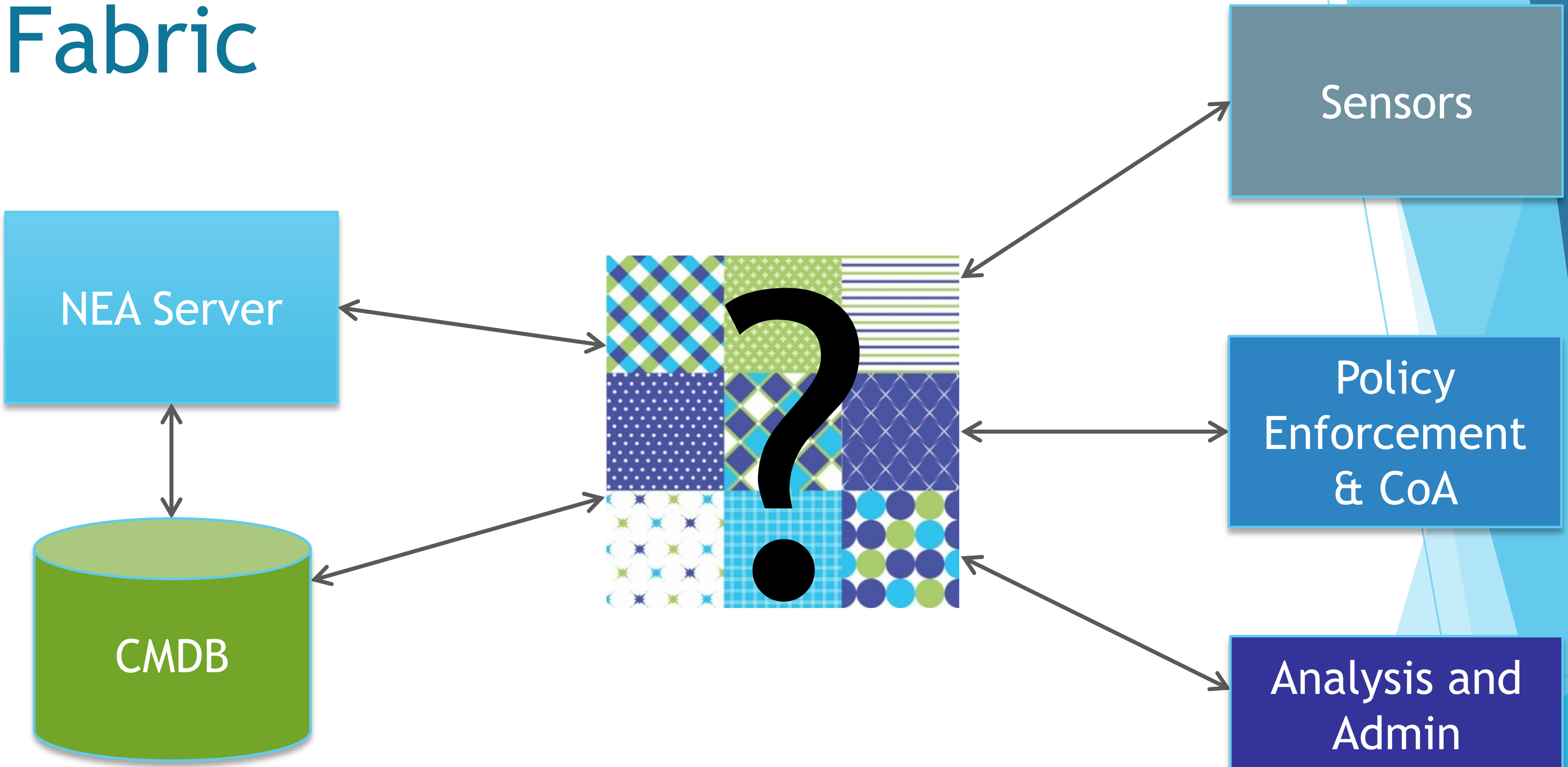


- Supports the maintenance of an enterprise repository of software inventory data
- Allows reporting full and delta software inventories using SWID tags
- Allows establishing subscriptions to monitor aspects of endpoints software inventory
- Detects updates to SWID tag repository on client machine, and update server
- Allows the server to query about SWID tag state

The TCG Endpoint Compliance Profile



The need for a Message Fabric



Questions and Discussion

- Counting endpoints and having basic knowledge of their state is common theme of compliance and control frameworks (e.g., FISMA, SOX, HIPPA). Do your customers see this the same way?
- How do you see a message fabric fitting in with this architecture?
- Do you envision other uses of endpoint software inventory and configuration information? How could a message fabric support these uses?

David Waltermire

Security Automation Team

Computer Security Division

Information Technology Laboratory

National Institute of Standards and Technology

david.waltermire@nist.gov