

SACM Information Model Based on TNC Standards

Lisa Lorenzin & Steve Venema

Agenda

Security Automation with TNC IF-MAP

SACM Information Model Based on TNC
Standards

Graph Model

Components

Operations

SACM Usage Scenario Example

Security Automation with TNC IF-MAP

Trusted Network Connect

Open Architecture for Network Security

- Completely vendor-neutral
- Strong security through trusted computing
- Original focus on NAC, now expanded to Network Security

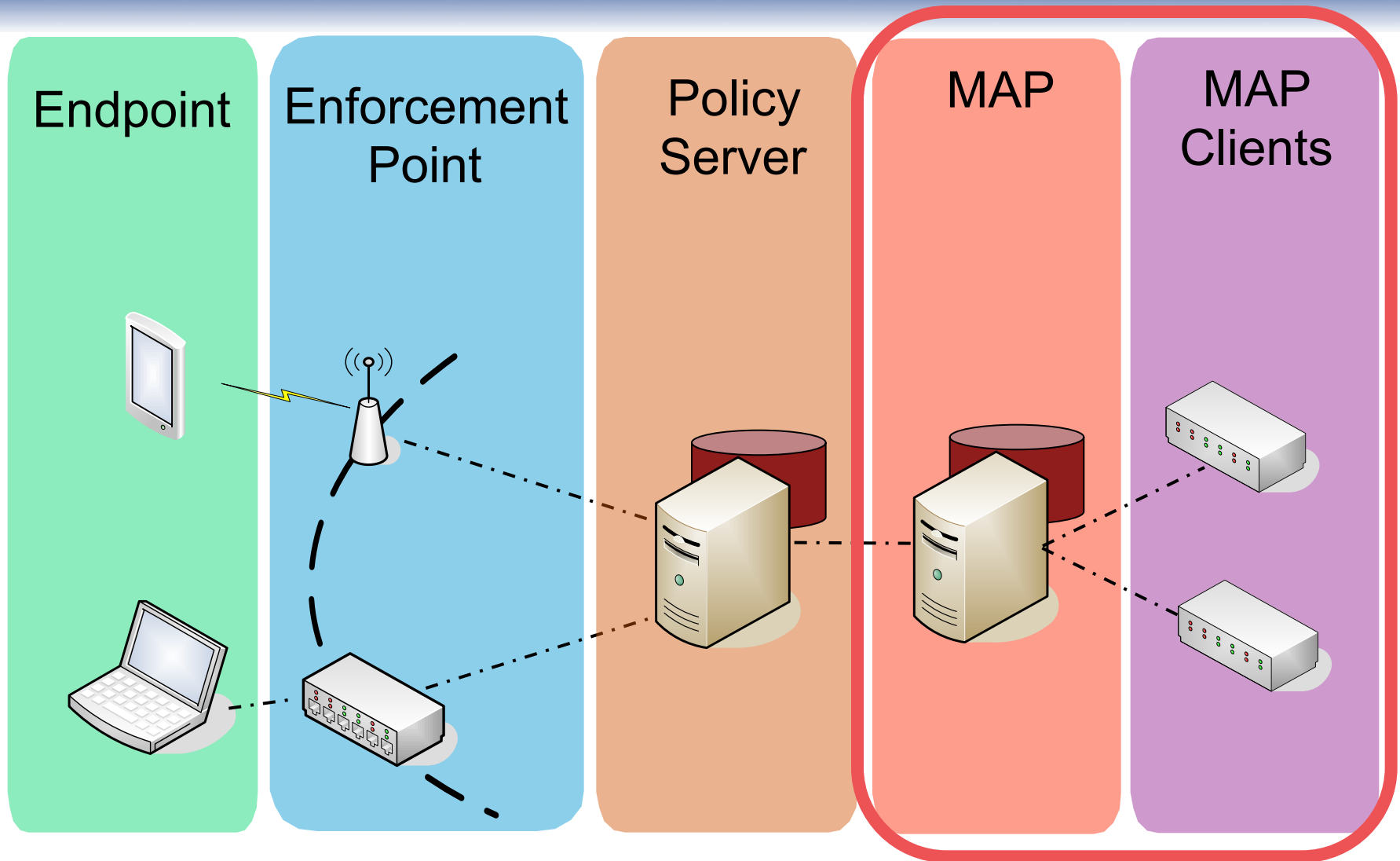
Open Standards for Network Security

- Full set of specifications available to all
- Products shipping since 2005

Developed by Trusted Computing Group (TCG)

- Industry standards group
- More than 100 member organizations
- Includes large vendors, small vendors, customers, etc.

TNC Architecture



Problems Solved by TNC

Network and Endpoint Visibility

- Who and what's on my network?
- Are devices on my network secure? Is user/device behavior appropriate?

Network Enforcement

- Block unauthorized users, devices, or behavior
- Grant appropriate levels of access to authorized users/devices

Network Access Control (NAC)

Device Remediation

- Quarantine and repair unhealthy or vulnerable devices

Security System Integration

- Share real-time information about users, devices, threats, etc.

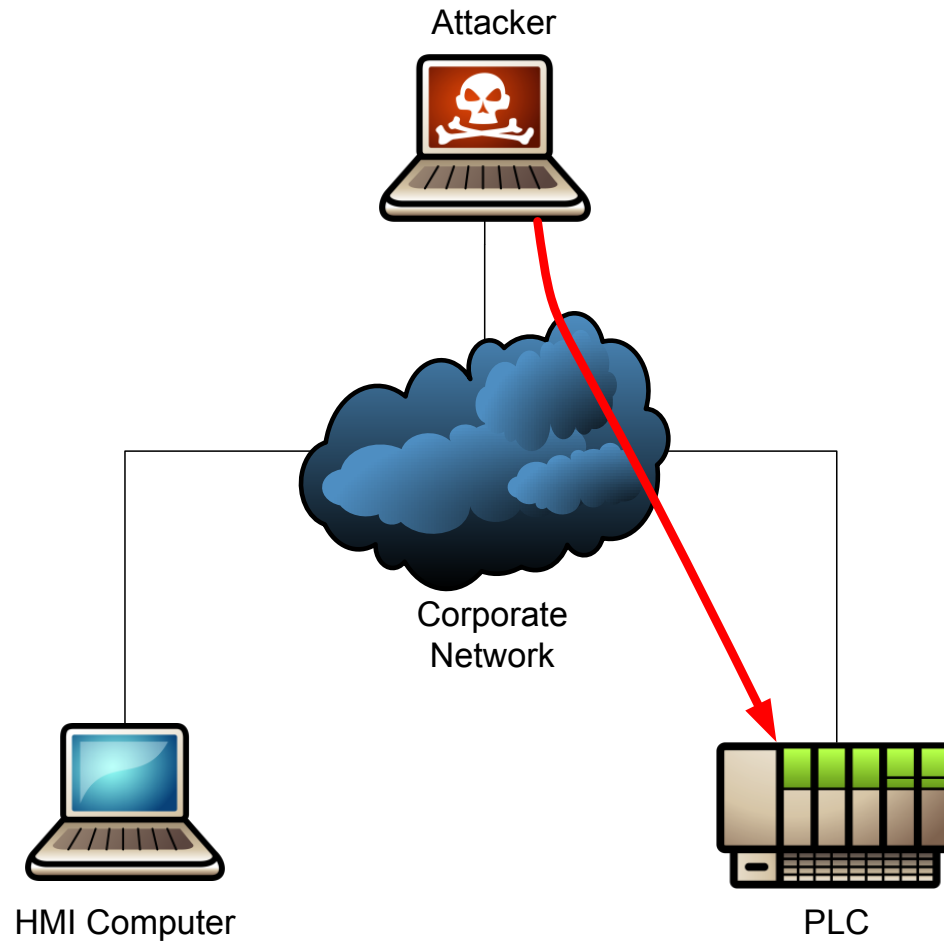
Security Automation

Coordination Challenge

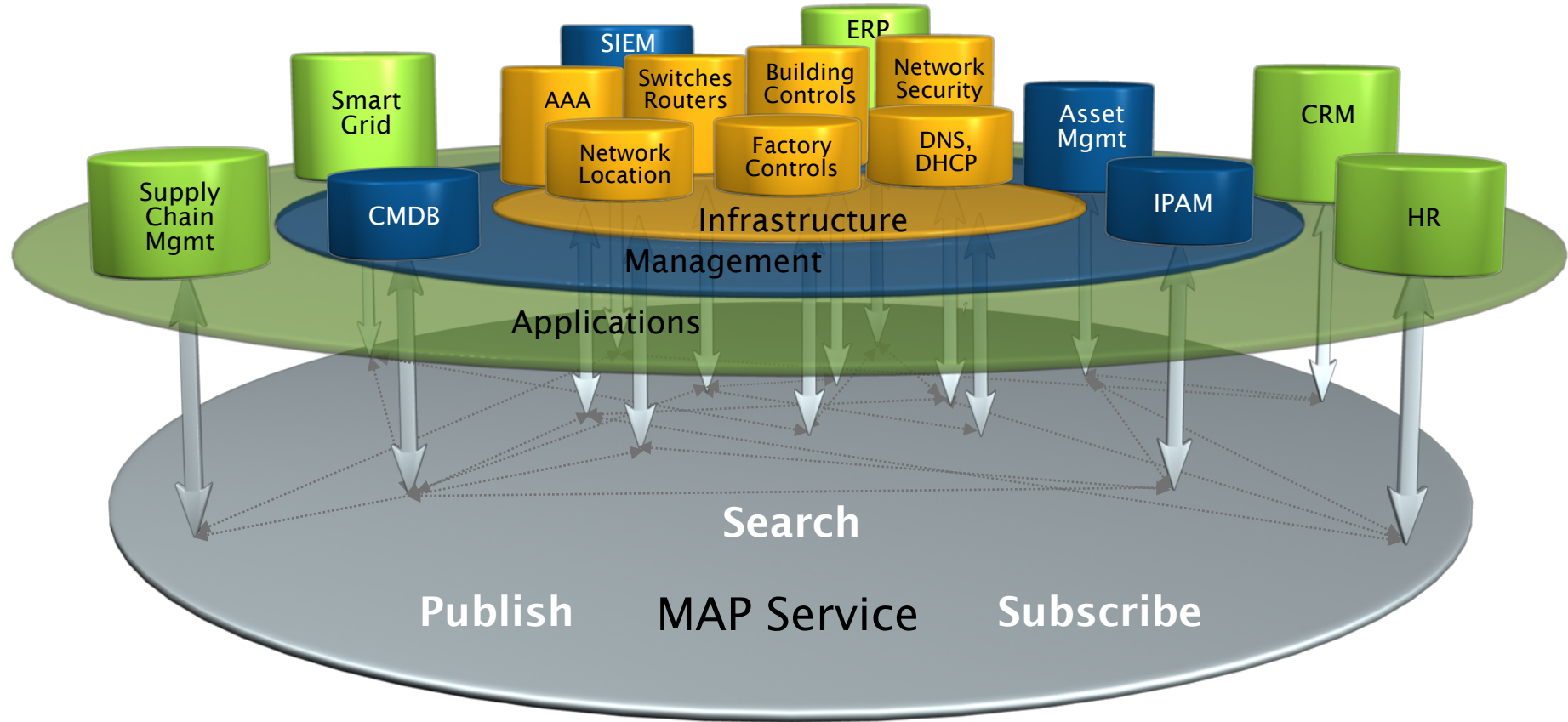
- Security infrastructure is complex, heterogeneous, and usually distributed
 - And it is only getting more so
- Large, real-time data flows between infrastructure components
 - Needed for coordination between Sensors, Flow Controllers, PDP's, etc.
 - Components often interested in different patterns & events
- Timely routing and reliability of delivery of this data is critical for coordination

Simple connectivity is insufficient for good coordination

ICS Security Challenge

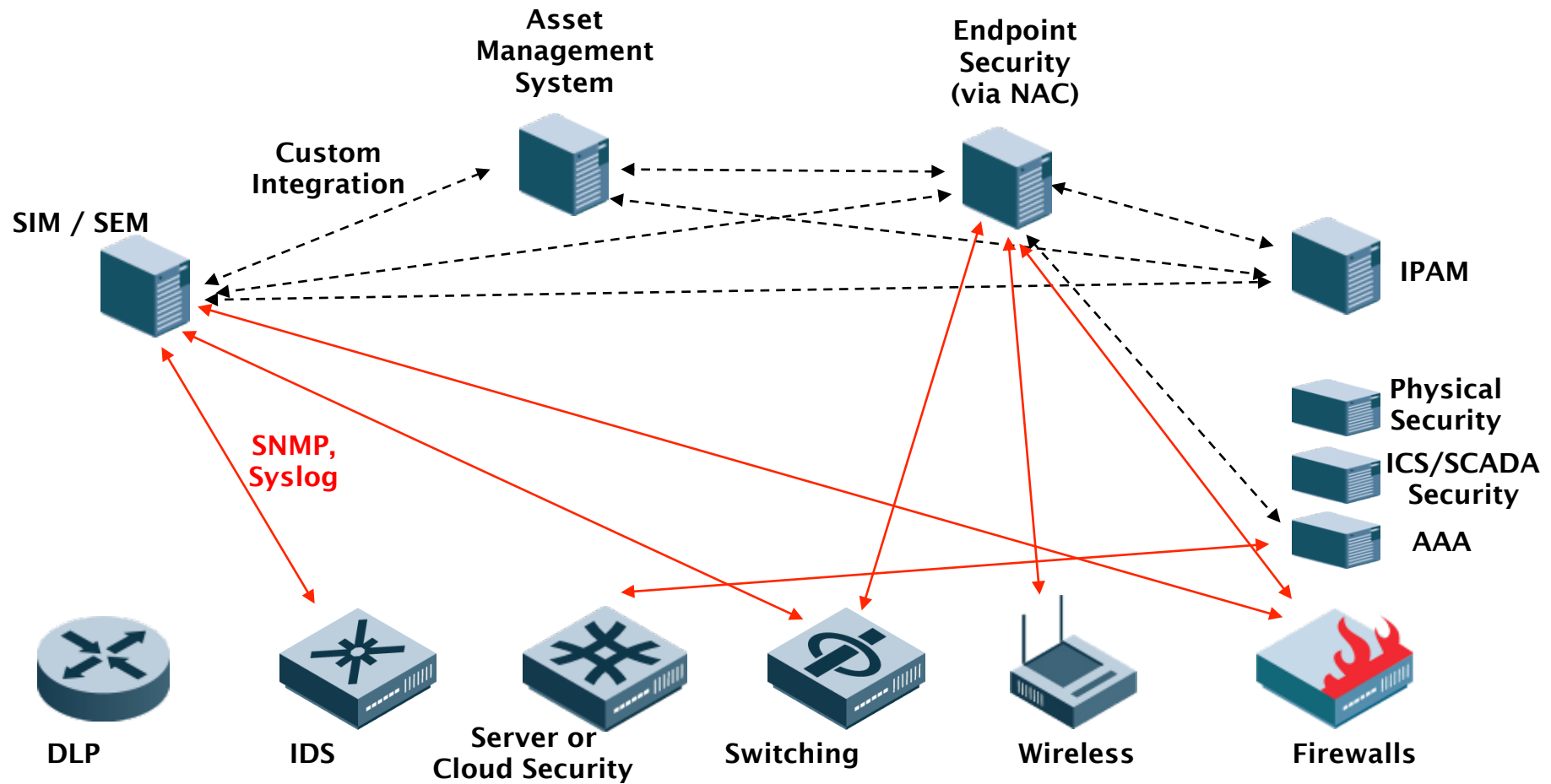


Security Automation with IF-MAP

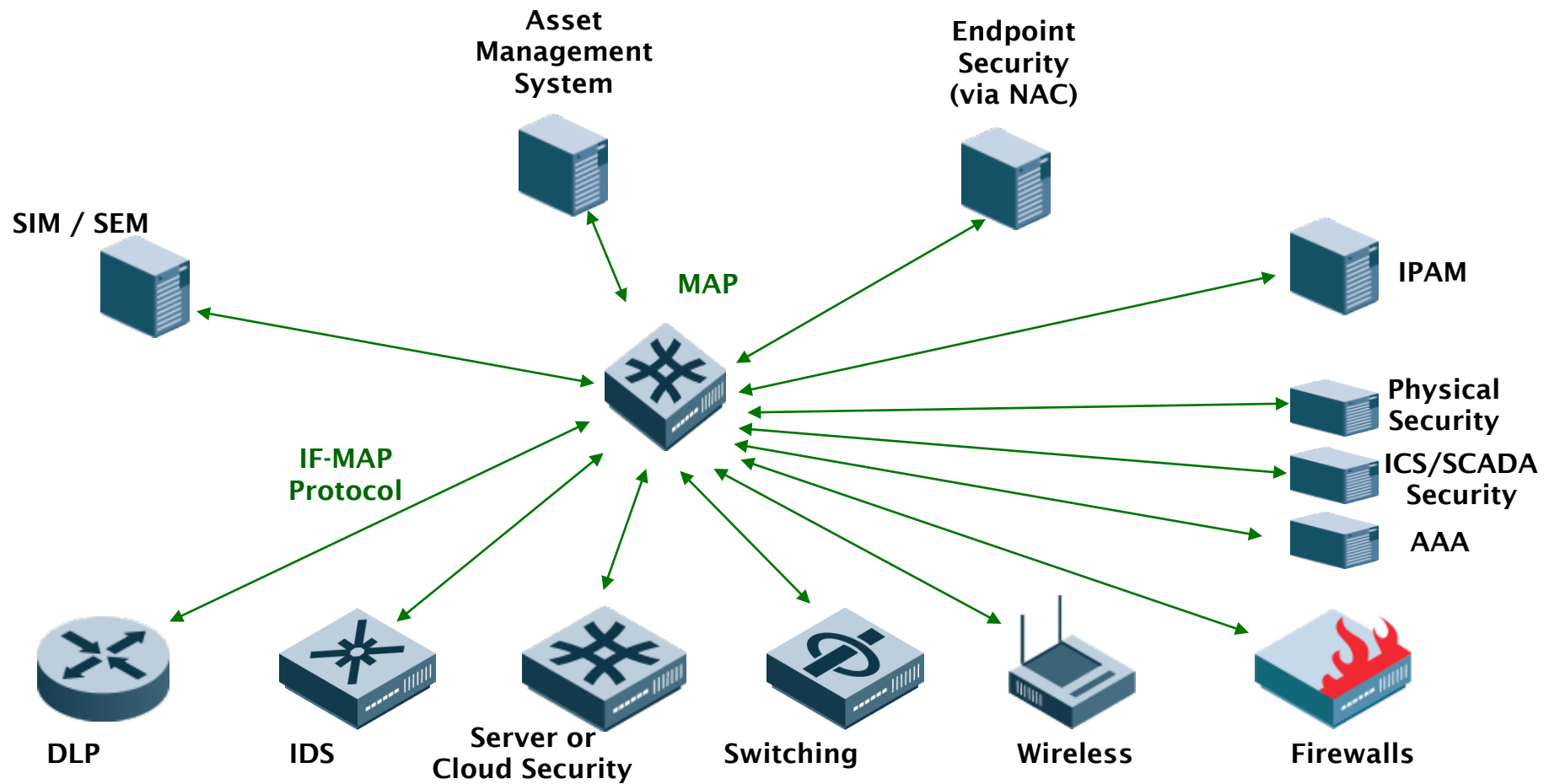


IF-MAP: XML > SOAP > HTTPS

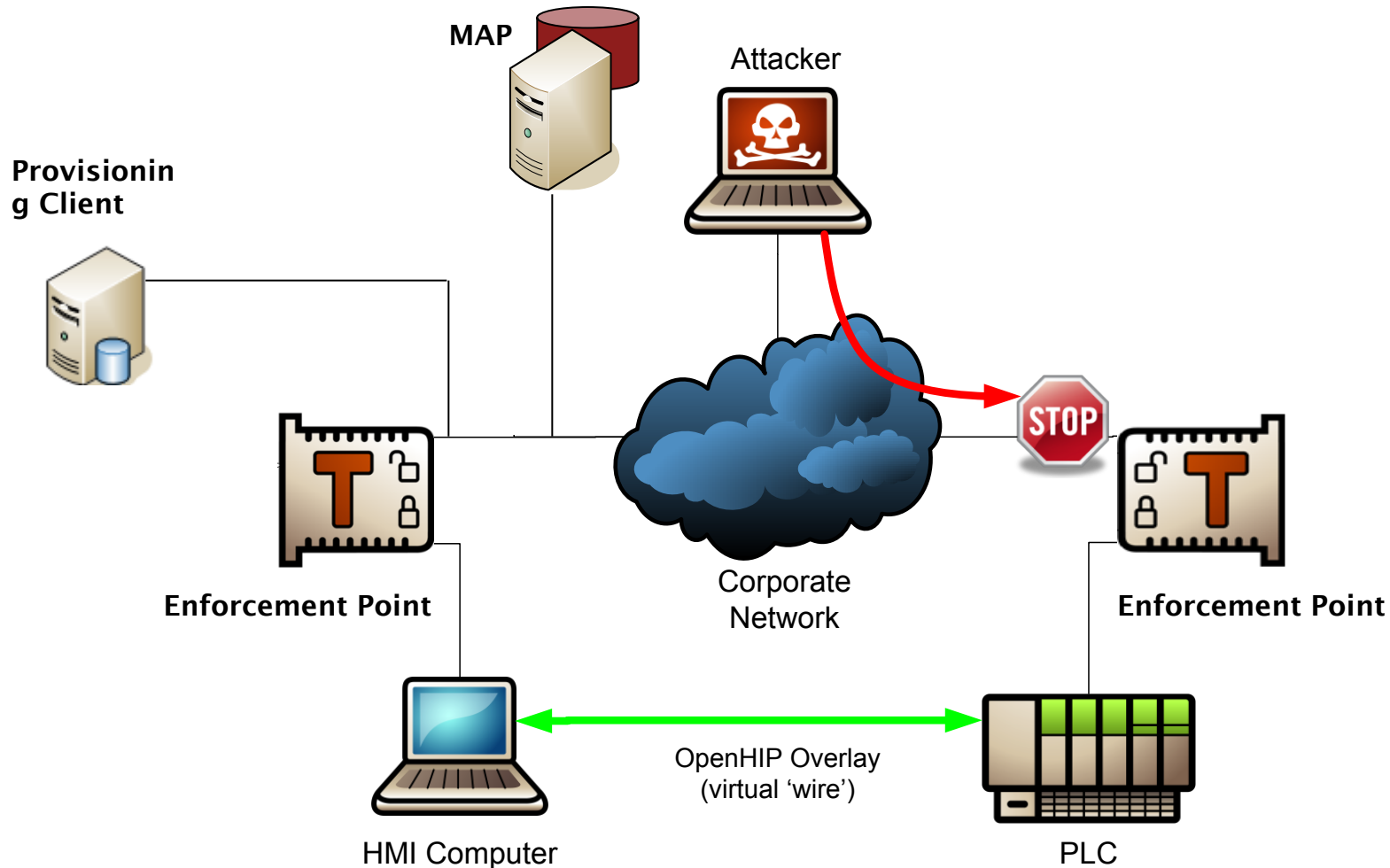
Communication Challenge



How IF-MAP Solves the Problem



IF-MAP Facilitates ICS Security



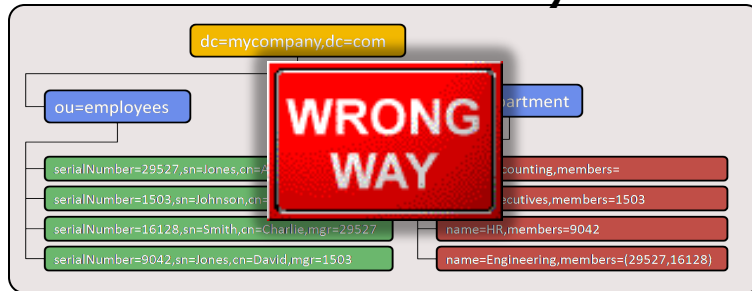


Properties of Security Coordination

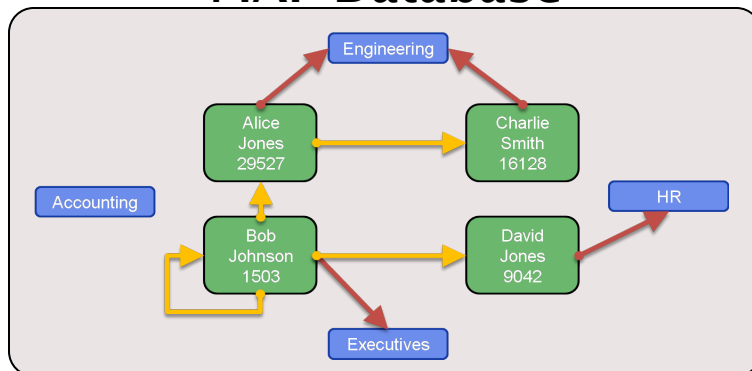
Relational Database



LDAP Directory



MAP Database



1. Lots of real-time data writes
2. Unstructured relationships
3. Diverse interest in changes to the current state as they occur
4. Distributed data producers & consumers

IF-MAP Components

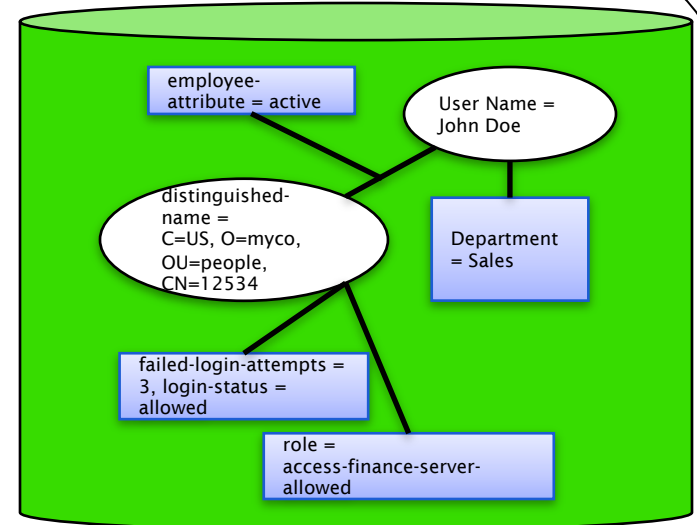
IF-MAP Client(s)



3 MAP Client Operations:

Publish
Subscribe
Search

IF-MAP Server



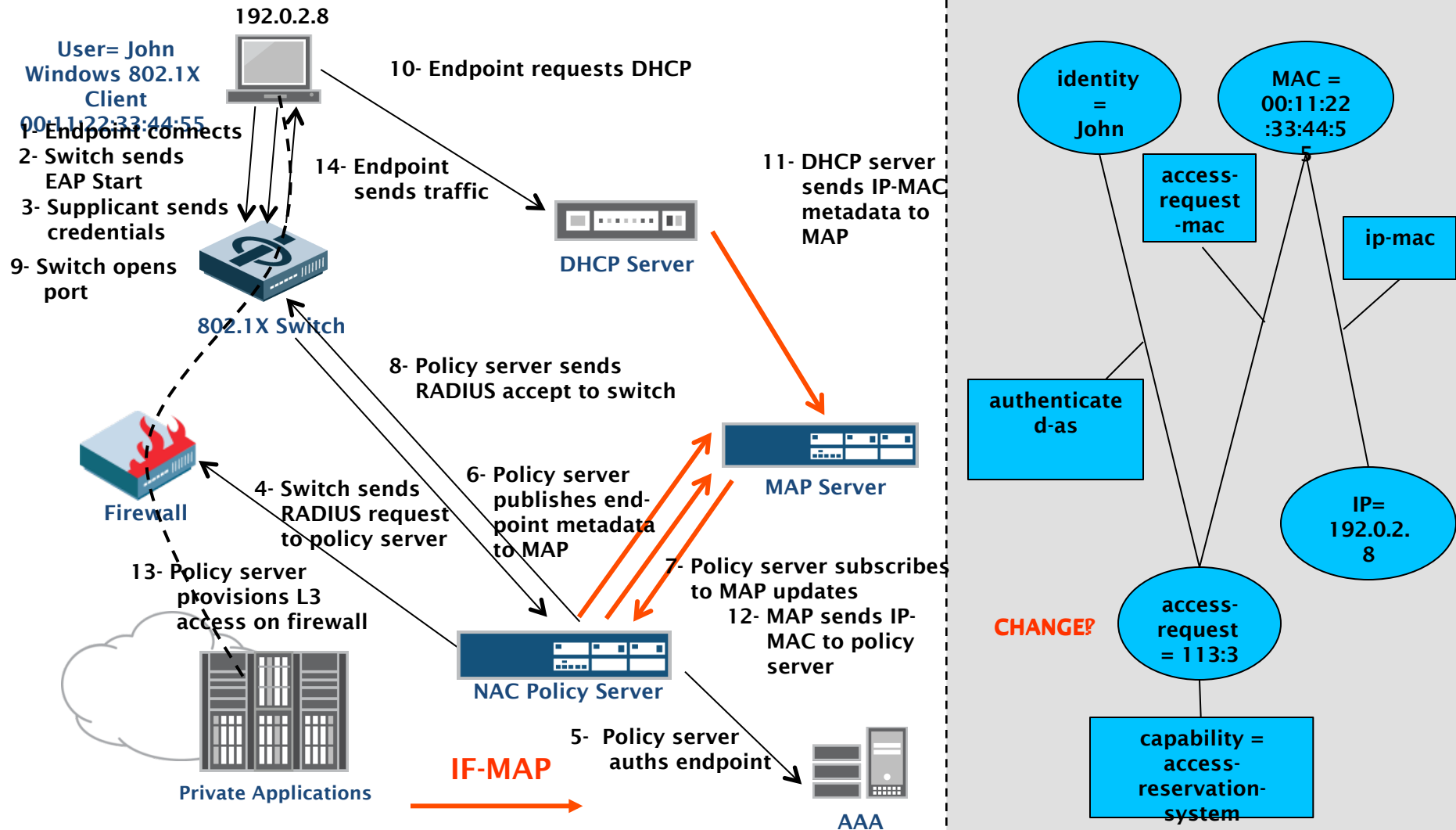
3 MAP Server Objects:

Identifiers
Links
Metadata

What Is Security Metadata?

- Metadata = Data about other data
 - A file's name and size are metadata about the file's data (the content)
 - “A picture of a car” is descriptive metadata about a file containing an image of a car
- Network security metadata describes attributes of network data flows and associated principals
 - Who is associated with what data flows?
 - What credentials were used?
 - What policy decisions have been made?
 - Recent unusual behaviors?

Network Security Metadata



Real-time Security Coordination

- IF-MAP is specifically designed to fit the security coordination use case
 - Optimized for loosely structured metadata
 - Publish/Subscribe capability for asynchronous searches
 - Highly scalable, extensible architecture
- Design is based on the assumption that you will never find the data relation schema to satisfy all needs
 - So you can move forward in spite of a lack of full relation specifications

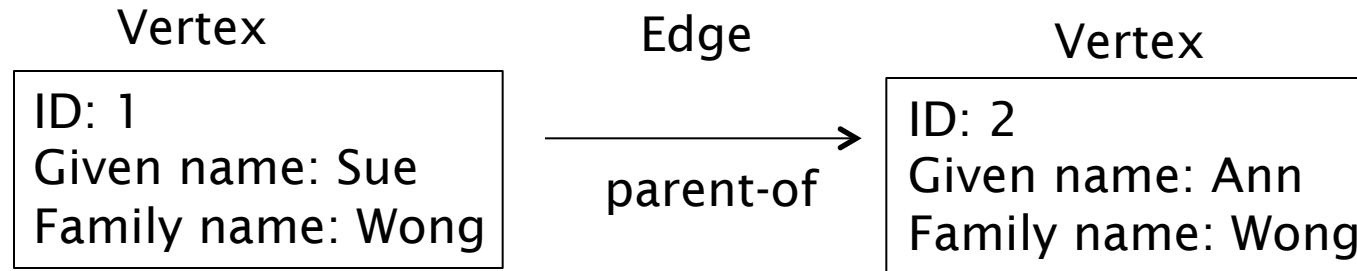
SACM Information Model Based on TNC Standards

Graph Models

A graph is composed of:




- A set of vertices
- A set of edges, each connecting two vertices
 - An edge is an ordered pair of vertexes
- A set of zero or more properties attached to each vertex and edge
 - Each property consists of a type and optionally a value
 - The type and value are typically strings

Graph Models & SACM



Graph Theory	Graph Databases	SACM Info Model
Vertex/Node	Node	-
Label	-	Identifier
Edge	Edge	Link
-	Property	Metadata

SACM Graph Model

	Identifiers	All objects are represented by unique identifiers
	Links	Connote relationships between pairs of identifiers
	Metadata	Attributes attached to Identifiers or Links

Typical Data Types:

- Identifiers: User, IP address, MAC address,
- Metadata: state (active/inactive), policy (allowed/denied), role (department/title), activity (failed authentication, violated policy,..)...

Elements

Components:

Actors...

- Posture Attribute Information Provider
- Posture Attribute Information Consumer
- Control Plane

Objects:

...and what is acted upon

- Collection tasks
- Posture
 - Posture attribute
 - Evaluation results
- Endpoint information
- History

Operations

Publish:

Tell others that...<metadata...>

- Providers share metadata for others to see
 - Example: Authentication server publishes when a user logs in (or out)

Query:

Tell me now if...*match*(metadata pattern)

- Consumers retrieve published metadata associated with a particular identifier and linked identifiers
 - Example: An application can request the current compliance state of an endpoint, filtered by who reported that state

Subscribe:

Tell me when...*match*(metadata pattern)

- Consumers request asynchronous results for searches that match when providers publish new metadata
 - Example: An application can request to be notified when any endpoint status changes from “compliant” to “not compliant”

Considerations – "Known Knowns"

Cardinality

- Single-valued vs. multi-valued metadata

Capability Negotiation

- Backwards compatibility
- Forwards expansion

Uniqueness

- Need “administrative domain” concept
- Harder than it first appears

Considerations – "Known Unknowns"

Provenance

- Whether producer is authoritative
- Freshness of metadata

Directionality of links

- Desirable to support a variety of use cases

Rootless searches

- Ability to query for / subscribe to information without knowing a specific starting point

Extensibility - "Unknown Unknowns"

- Metadata
- Identifiers
- Operations
- Search query construction
- Others?

Extend ALL the things!

SACM Usage Scenario

2.2.3. Detection of Posture Deviations

Example corporation has established secure configuration baselines for each different type of endpoint within their enterprise including: network infrastructure, mobile, client, and server computing platforms. These baselines define an approved list of hardware, software (i.e., operating system, applications, and patches), and associated required configurations. When an endpoint connects to the network, the appropriate baseline configuration is communicated to the endpoint based on its location in the network, the expected function of the device, and other asset management data. It is checked for compliance with the baseline indicating any deviations to the device's operators. Once the baseline has been established, the endpoint is monitored for any change events pertaining to the baseline on an ongoing basis. When a change occurs to posture defined in the baseline, updated posture information is exchanged allowing operators to be notified and/or automated action to be taken.

Components

- **Posture Attribute Information Provider**
 - An endpoint security service which monitors the compliance state of the endpoint and reports any deviations for the expected posture.
- **Posture Attribute Information Consumer**
 - An analytics engine which absorbs information from around the network and generates a "heat map" of which areas in the network are seeing unusually high rates of posture deviations.
- **Control Plane**
 - A security automation broker which receives subscription requests from the analytics engine and authorizes access to appropriate information from the endpoint security service.

Potential Identifiers

- Identity
- Software Asset
- Network Session
- Address
- Task
- Result
- Policy
- Others?

Potential Metadata

- Authorization
- Location
- Event
- Posture Attribute
- Others?

Workflow

1. The analytics engine (Consumer) establishes connectivity and authorization with the transport fabric (Control Plane), and subscribes to updates on posture deviations.
2. The endpoint security service (Provider) requests connection to the transport fabric.
3. The transport fabric authenticates and establishes authorized privileges (e.g. privilege to publish and/or subscribe to security data) for the requesting components.
4. The endpoint security service evaluates the endpoint, detects posture deviation, and publishes information on the posture deviation.
5. The transport fabric notifies the analytics engine, based on its subscription of the new posture deviation information.