SWID M&A

draft-coffin-sacm-nea-swid-patnc-01

SACM WG Meeting – IETF 96
July 18, 2016
Agenda

• Data Model

• Architectural Role
What are we reporting?

• List of globally unique identifiers?
  • Who manages the identifier list?

• Descriptive information?
  • Seems to be agreement that what is used here should be mirrored in the SACM IM
  • Seems to be agreement that this needs to be extensible
  • How much is enough? (What does the Vulnerability Assessment Scenario need?)
    • How hierarchical? (Flat attribute list or something more structured?)

• Both?
Possible descriptive structures mentioned

• Based on ISO SWID 2015?

• Based on XORCISM (or lessons learned therefrom)?

• Other?

• Build our own?
Architectural Role

• At previous Virtual Interim questions were raised about SWID M&A (and NEA in general) as part of SACM
  • Questioned because of NEA’s role as a data consolidator rather than a general grid member
• Should have consensus as to if/how NEA lives in SACM
Status and Next Steps

• Once we have consensus on the SWID M&A data model, the spec will be mostly done
  • There have been no concerns over the SWID M&A commands and data flows – questions have been on “what” is transferred, but not “when” or “how”
BACKUP
SWID M&A in the NEA Architecture

```
Endpoint                     Server
+--------------------------+
|                          |
+--------------------------+    +--------------------------+
|                          |
| SWID                     | SWID M&A                |
| Posture                  |<------------------------>| Posture                  |
| Collector                | PA-TNC                  | Validator               |
+--------------------------+                          +--------------------------+
|                          |
|                          |
| IF-IMC*                  | IF-IMV*                 |
+--------------------------+                          +--------------------------+
|                          |
|                          |
| PB Client                |<------------------------>| PB Server               |
|                          |                          |                        +--------------------------+
|                          |                          |
|                          |                          |
|                          |                          +--------------------------+
|                          |
|                          |
| PT Client                |<------------------------>| PT Server               |
|                          |                          |                        +--------------------------+
|                          |                          |
|                          |                          |
|                          |                          +--------------------------+
|                          |
|                          |
+--------------------------+                          +--------------------------+

* Not currently part of NEA, but part of the compatible TNC architecture
SWID M&A Message Flows: Demand-Driven (Pull)

- 4 types of Response attributes depending on Request parameters
  - SWID Tag Inventory – Complete or targeted inventory expressed in SWID tags
  - SWID Tag Identifier Inventory – Complete or targeted inventory using tag IDs
  - SWID Tag Events – Changes since a given event number using in SWID tags
  - SWID Tag Identifier Events – Changes since a event number using tag IDs
Change Tracking in SWID M&A

• Posture Collectors MUST monitor their SWID tag collection for changes
  • Can be real-time or periodic monitoring
• Each change is assigned a unique, sequential “event number”
• All event numbers have an associated “event epoch”
• Within an epoch, event numbers fully order all change events
• All inventories are reported along with the event number and epoch of the last recorded event at time of inventory
  • Given this and a list of subsequent events, can track all changes just using deltas
  • Epoch changes represent discontinuities – no way to track across
SWID M&A Message Flows: Event-Driven (Push)