

# Description of Actual State Sensor Types for the Software Asset Management (SWAM) Capability

---

7 Jul 2014

---

# 1 Purpose

This document is intended to provide insight on the types of tools and technologies that can be utilized to support the collection of asset information required to perform the SWAM capability (as part of Continuous Diagnostics and Mitigation (CDM)). The 'Description of Generic Sensor Types for the Continuous Diagnostic and Mitigation (CDM) Collection System' document described the actual state sensor types for CDM to include information about potential for operational impacts and general data accuracy issues associated with each particular sensor type.

The SWAM capability provides an organization visibility into the software installed and operating on all in-scope devices so they can appropriately manage authorized software and remove unauthorized software. The SWAM capability relies on many different sensors on a network collecting software related data. These sensors directly and indirectly collect software data from each managed device connected to the network. Sensors often have primary roles that do not necessarily include just reporting device data; however, during the process of performing its primary function (e.g., asset management), the actual state sensor collects detailed device data that can be extrapolated and used to support SWAM. Following are examples of how common tools and technologies can be employed as actual state sensors to support the SWAM capability. The level of control of the type of data being collected and the collection capability of the sensor on the network determines the actual state sensor category in which the technology, device, or tool is performing. For example, a host-based agent that is collecting event logs and periodically sending them to an audit management system that CDM can access is considered an Asset Management Repository and not an Endpoint-Based Agent. This is because the configuration and deployment of the host-based agent is performed by the audit management system for the purposes of the audit management system, and the only influence/interaction for CDM is receiving reports from that system. Host-based agents that are collecting information directly for CDM consumption are considered to be endpoint-based agents, even if CDM receives that data from the management server and not the endpoint directly.

---

## 2 SWAM Actual State Sensor Types

### 2.1 Active Network Sensor

*An Active Network Sensor actively probes the network or a device over the network.*

An active network sensor either probes or queries (e.g., software and application scanners) devices on the network for current or existing asset information. For SWAM, these sensors are configured to collect and report installed software information from devices (e.g., patch level). Software and application scanners can be utilized to identify approved and rogue software installed on network devices that may be found through scans (to include outdated and malicious programs). Depending on the network size however software and application scanners may take a considerable amount of time collecting results from each device residing on the network. Larger networks with hundreds of software products installed on thousands of devices may take hours to collect results. This should be considered before deploying this type of sensor. These sensors can be great at identifying software data but they can only assess a “snapshot” of time in terms of a network device’s status. So scanning with these tools needs to be performed regularly as device changes may occur frequently.

Configuration management tools provide situational awareness of assets operating on the network and can track and control software related changes for managed devices operating on the network. These tools provide a capability to establish an operational software baseline. In the event that a change was made to that baseline, possibly through the removal or addition of software, the configuration management suite would be able to identify when the change was made, what change was made, and who changed it. In most network infrastructures, a central manager is used to collect the data received from endpoint devices. This can be done through a credentialed means (e.g., SSH authentication, SNMPv3) to produce more accurate and granular results during polling periods, however constant or excessive polling can introduce more bandwidth overhead during collection and deployment cycles. By supplying credentials administrators are not only able to scan for results but also deploy security policies and software configuration settings and push any required software related updates to network devices.

Lastly, products or tools (e.g., vulnerability scanners) can be used to perform passive TCP/IP fingerprinting of remote devices by collecting identification information during standard network communications. These tools establish connections (e.g., telnet) by sending morphed or empty TCP packets to a remote target device and the response from the device is then analyzed for identification purposes. The identifying information may include OS types and versions. These tools however are ineffective against defense mechanisms in place that block or limit the type and amount of traffic a device responds to.

### 2.2 Passive Network Sensor

*A Passive Network Sensor is designed to capture and/or collect network traffic that passes across a monitored network link.*

Passive Network Sensors only collect software data that they are configured to identify—all other network traffic outside the configuration scope of the passive network sensor will not be collected. Packet analyzers and certain network scanners/sniffers are examples of passive network sensors that can support the SWAM capability indirectly.

Packet or protocol analyzers act as passive network sensors by capturing data associated with all devices communicating on a network segment. By capturing the communications between the devices, the data can then be analyzed to identify any software related products used during the communication. Network scanners or sniffers can use enumeration techniques like banner identification (i.e., banner grabbing) to glean information about devices on the network. These tools can provide comprehensive and timely insight into what software products and applications are communicating on the network, but have varying degrees of accuracy for determining what version of the product is installed. These tools cannot determine anything regarding software that does not communicate across the network.

### 2.3 Asset Management Repository

*An Asset Management Repository is a collection of data created and updated as part of a process or activity that manages that asset for an organization.*

An asset management repository aggregates managed device data (to include software asset information) as part of a tool or process that manages that asset for an organization. The data it contains may include software license details, software versioning, and software authorization information. These sensors in most cases may not be configured to just report software asset information but due to their ability to maintain a plethora of device asset information can be utilized to support the SWAM capability.

Asset management tools and inventory collection managers serve as great sources of software asset information because they collect data that can be used by the SWAM capability and eliminate the need to deploy additional sensors on the network. Patch management tools, which can be integrated with other software suites, offer a means for all managed devices to stay compliant and up-to-date with the latest software updates. These tools maintain an inventory of every device that reports asset information to a central manager or console and can be used to provide granular software related details (e.g., patch levels, software versions). Although deploying software updates with patch management tools can be automatically configured for all the devices on the network it can take a considerable amount of time for each of the devices to receive the required updates if there are a great deal of updates to push or install across the network. Also these tools do not provide an automatic “clean up” mechanism for the uninstallation of software patches that are no longer supported.

Software license management tools provide an automated capability to record various types of software licenses and product keys on the network and its respective owners. For mobile devices on the network, mobile device management suites can be utilized for identifying, maintaining, and updating managed mobile devices. Once this device data has been collected the database can be utilized to provide software related information in regards to what type of software or application is operating on the mobile devices along with the current software status level.

Another location for software related asset information is the enterprise “software depot”. These tools maintain copies of software approved for use in the organization and keep logs of which devices have downloaded which products. Lastly, inventory auditing/collection tools that are used to collect asset information that includes all the software installed on a managed device or any log entries related to install/uninstall operations can also support SWAM.

All of these different repositories can be queried to create comprehensive lists of installed software to include software patches/hotfixes.

## 2.4 Network Event Sensor

*A Network Event Sensor is designed to detect and report events of interest to a defined location in a timely manner.*

Network event sensors provide situational awareness of unauthorized events that take place on the network. These sensors are able to do this by monitoring and alerting on predefined audit security and compliance relevant information received from network devices. Managed devices on the network are configured through security policy to forward audit log data via a specified protocol (e.g., syslog, WMI, SNMP) to a network event sensor and once the event has been received it can be analyzed through real-time correlation and historic analysis. Configuring a device’s event audit log to alert when software is installed/uninstalled on a device or being able to identify when malicious software has been executed or downloaded onto a device are examples of events that should be defined and can be identified with network event sensors.

Another example is in the use of application whitelisting (AWL). By utilizing AWL tools, organizations can specify beforehand which applications are allowed to run on a device and deny all other applications and programs from executing. There are times when AWL products can be configured to provide alerts instead of blocking when new or unauthorized software is installed.

Examples of network event sensors include event logging tools and Security information and event management (SIEM) suites. A SIEM is an example of a network event sensor that aggregates logs from various other sensors to provide the ability to consolidate and correlate device data. This is beneficial in regards to SWAM because policies can be configured on the network to notify administrators when a change has occurred on a network device. Once that change has happened or been attempted (e.g., unauthorized installation of software) the local IDS will send an alert to the SIEM for correlation and further analysis. In most cases if the managed endpoint device has the event log or auditing function activated data can be forwarded to a SIEM tool however in some instances a SIEM tool may require the use of agents or credentialed means of access to obtain event log data.

## 2.5 Endpoint-Based Agent

*An Endpoint-Based Agent is a software client installed on, or natively embedded within, the operating system of a device.*

Endpoint-based agents are often configured to collect software inventory related information and monitor for unauthorized software events. While they usually report findings to an asset management repository, they are listed in this section because they are specifically configured to collect SWAM related information for use in managing software products and defending against malware. These agents can be configured on the managed device to detect or prevent unauthorized events from happening based upon heuristics or certain signature-based detection methods. Security endpoint agents are examples of endpoint-based agents that could be used to support SWAM. These sensors may include file integrity monitoring agents, antivirus or anti-malware agents, and trusted network connect (TNC) installed technology (either natively embedded in the OS or client side agents). File integrity monitoring agents send alerts to a central server when files may have been altered or changed on a managed device according to a security policy. Antivirus and malware agents can be installed on a managed

device to provide a means to detect known malicious and unauthorized software and prevent malicious software from being downloaded or executed however they do not defend against any new malicious software for which no known signature exists (e.g., zero-day threats). Using TNC endpoint technology, organizations are able to keep track of what software a device is running and ensure devices comply with enterprise security policies. TNC provides the standards-based mechanisms to support the secure exchange of software identification (SWID) tag information, which enables accurate software inventory information to be made available to an organization.