

# Software Asset Management ([SWAM](#)) Capability Data Sheet

---

## Desired State:

- Only authorized [software products](#) and [executable files](#) are installed on in scope devices
- All [devices](#) are assigned or authorized for a set of [device attributes](#) and any authorizations are revalidated on a periodic basis
- All software installation and execution restriction mechanisms are deployed and configured correctly
- All blacklists are up-to-date
- All software not explicitly identified in a [whitelist](#) or blacklist is included in the [graylist](#)
- All graylist software is investigated and determined to be authorized or unauthorized within a specific period of time, and added to the appropriate whitelist or blacklist

## Desired State Data Requirements:

Data Item	Justification
<a href="#">Authorized Hardware Inventory</a> to include assigned and authorized device attributes	To identify what devices to check against what defect checks
The associated Value for attribute <sup>1</sup>	To prioritize defects associated with devices.
Sets of attributes that are designated mutually exclusive per the D/A's policy	For comparison with the set of assigned attributes for device

---

<sup>1</sup> This value will initially be defined by the D/A with some guidance from the CDM PMO. Once the necessary data becomes available, it will be calculated from the value assigned by the D/A to assets.

<p>A listing of all authorized software for the D/A to include:</p> <ul style="list-style-type: none"> <li>• Data necessary to accurately identify the software product and compare to actual state data collected             <ul style="list-style-type: none"> <li>○ Vendor</li> <li>○ Product</li> <li>○ Version/Release level/Patch level</li> <li>○ <a href="#">Software identification tag (SWID)</a></li> <li>○ <a href="#">CPE</a></li> </ul> </li> <li>• Authoritative listing of executable files associated with product</li> <li>• Software Manager</li> <li>• Expiration policy</li> <li>• <a href="#">Software authorization status</a></li> <li>• Date initially authorized</li> <li>• Date last authorized</li> <li>• Date revoked</li> </ul>	<p>To calculate expiration dates for authorized software</p> <p>To enable automated removal of differences that are not defects</p> <p>To be able to uniquely identify the software</p> <p>To be able to validate that the software on the device is truly the software authorized</p> <p>To know who to instruct to fix specific risk conditions found</p> <p>To assess each such person’s performance in risk management</p>
<p>Management responsibility for each software management function for each authorized software product</p> <p>Local enhancements<sup>2</sup> might include:</p> <ul style="list-style-type: none"> <li>• Approvers being assigned</li> <li>• Managers being approved</li> <li>• Managers acknowledging receipt</li> </ul>	<p>To identify management responsibilities for ensuring that licensing, patching, and configuration standards are up-to-date</p> <p>If not specified explicitly, this is assumed to be the device manager</p> <p>To know who to instruct to fix specific risk conditions found</p> <p>To assess each such person’s performance in risk management</p>

---

<sup>2</sup> Departments and Agencies can define data requirements and associated defects for their local environment. This is done in coordination with the CMaaS contractor and these local defects are not reported to the Federal Dashboard.

<p>A set of Software Profiles for the D/A to include:</p> <ul style="list-style-type: none"> <li>• Associated attributes<sup>3</sup></li> <li>• Allowed software</li> <li>• Mandatory software</li> <li>• <a href="#">Prohibited software products blacklist</a></li> <li>• <a href="#">Known-bad blacklist</a><sup>4</sup></li> <li>• Update frequency for known-bad blacklist</li> </ul>	<p>To compare with the software present on a device to determine defects</p> <p>To define authorized and unauthorized software on a per device basis</p> <p>To determine when software no longer authorized for the environment is being used for baselines</p> <p>To determine if known-bad blacklists are out of date</p>
<p>Sets of device attributes that require a unique software profile when assigned to the same device to include:</p> <ul style="list-style-type: none"> <li>• Software profile(s) replaced</li> <li>• Software profile(s) used</li> </ul>	<p>To enforce more restrictive policies on devices that are assigned sets of attributes (e.g., database server and database authentication server)</p>

**Actual State:**

- All enumerated software installed on all devices
- All known-bad blacklists deployed and date/time of last update
- Collection mechanisms and/or processes to detect and record/report the actual state information

**Actual State Data Requirements:**

While not explicitly stated below, all Actual State Data elements must have a date/time associated with each collection instance of that element<sup>5</sup>.

---

<sup>3</sup> Software profiles have a one-to-many relationship with device attributes. One profile can have more than one device attribute associated with it (e.g., both Internal Web Server and External Web Server can map to the same Web Server software profile), but every device attribute is associated with exactly one software profile.

<sup>4</sup> Known bad blacklists are quite large, very dynamic, and often maintained by an antivirus or antimalware vendor. It is not expected that the D/A will know what software is on the list, but that they will know what blacklist is to be used and how frequently it is to be updated.

<sup>5</sup> Collection often occurs in batches, where the sensors collect from a set of devices at once. As long as a date/time can be provided for the data resulting from that collection to a reasonable precision (i.e., ± 1 hour), that is acceptable.

Data Item	Justification
The software installed on every device. This data must be converted into a format that can be compared with the authorized software inventory. Examples include: <ul style="list-style-type: none"> <li>• SWID</li> <li>• CPE</li> </ul>	To identify when unauthorized software is installed on a device
Data necessary to determine how long unauthorized software has been present on a device. At a minimum: <ul style="list-style-type: none"> <li>• Date/time it was first discovered</li> <li>• Date/time it was last seen</li> </ul>	To determine how long unauthorized software has been on a device
Known-bad blacklist used to check device to include version number or date of last update	To determine if device was checked for unauthorized software To determine if the known-bad blacklist is up-to-date per policy

**Defects:**

A defect is defined as a discrepancy between the authorized software inventory and what software is present on a device. It can also be an inconsistency between policies or authorizations associated with: device roles, software products, blacklists, or software profiles. The following are defects for SWAM:

Defect Type	Why is this considered a risk condition?	Typical Mitigation <sup>6</sup> Option 1:	Typical Mitigation Option 2:
Software authorization is expired	The risk associated with authorization decisions increases with time. Decisions that were acceptable in the past may now be considered too risky	Reauthorize the software	Otherwise, revoke/suspend the software’s authorization
Software is on graylist and is marked treated as authorized <sup>7</sup>	Malicious software is allowed to be installed on devices	Authorize the software	Otherwise, move it to the graylist-treat as unauthorized

<sup>6</sup> Risk acceptance is always an option. In the case of Option 1 and Option 2, the risk conditions and scores do not go away. They remain visible to ensure that the organization understands the impact of their risk acceptance decisions over time and in aggregate.

<sup>7</sup> CDM will create a defect for any software placed on the graylist-treat as authorized. The expectation is that the D/A will decide how long organizations have to investigate and either add the software to appropriate white/blacklists. This grace period will be built into the scoring

Defect Type	Why is this considered a risk condition?	Typical Mitigation <sup>6</sup> Option 1:	Typical Mitigation Option 2:
An authorized device's assigned device roles are not collected or defined  (Non-reporting for device roles)	A device is allowed to have excessive, unauthorized, or outdated software.	Correct the processes developed to provide necessary data  If the device roles are known for the device, record them	Otherwise, remediate the collection issue if the process is working appropriately  Otherwise, determine the appropriate device roles and record them
The D/A's device role policy is violated  (e.g., One device is assigned device roles that are deemed incompatible by policy)	Device role policies are designed to prevent one device from being able to overly impact the organization due to excessive access or privilege.	Remove device from device role considered mutually exclusive	Otherwise, update device role policy;
Known-bad blacklist is not defined for device	Compromised devices continue to operate on the network	If known-bad blacklist to use for device is known, record it in the authorized software inventory	Otherwise, determine which known-bad blacklist to use and record it in the authorized software inventory
Known-bad blacklist is not deployed or implemented for device	Devices that have become compromised since their last check continue to operate on the network	Run known-bad blacklist against the device and deploy capability to perform timely automated checking	Otherwise, remediate the implementation issue with existing capability
Known-bad blacklist is out of date	Devices can be compromised by known attacks and continue to operate on the network until the blacklist is updated	Update the blacklist for the device and deploy capability to perform timely automated checking	Otherwise, remediate the implementation issue with existing capability

---

function in the dashboard instead of the defect identification process meaning that this will be a defect, but not scored until the grace period is expired.

Defect Type	Why is this considered a risk condition?	Typical Mitigation <sup>6</sup> Option 1:	Typical Mitigation Option 2:
Device has unauthorized, but <u>not</u> blacklisted or graylist-treat as unauthorized, software installed <sup>8</sup>	Unauthorized software is less trustworthy, more vulnerable, or increases the attack surface for an organization as compared to authorized software.	If the software should be on a device, authorize it and record it in the authorized software inventory	Otherwise, remove the software from the device
Device has blacklisted or graylist-treat as unauthorized software installed	Blacklisted software (e.g., malware) is most often either an artifact of compromise or an initial precondition for compromise	Remove the software from the device	If investigation shows that software should NOT be on the blacklist or graylist-treat as unauthorized, then authorize software for appropriate profiles
Software installation restriction mechanism is not deployed or configured correctly	Unauthorized or malicious software is allowed to be installed on the device	Deploy the mechanism or configure it per policy	Otherwise, remediate the implementation issue with existing mechanism
Software execution restriction mechanism is not deployed or configured correctly	Attacks that exploit vulnerable or flawed software can successfully compromise device	Deploy the mechanism or configure it per policy	Otherwise, remediate the implementation issue with existing mechanism
An important data element of the authorized software inventory is missing	A key piece of information used to score or assign risk is unknown	If the data element is known, record the information in the authorized software inventory	Otherwise, determine or define the data element and record this in the authorized software inventory
Installed software has not been reported within a set timeframe for a device  (Non-reporting for Software)	The Department or Agency's (D/A) ability to monitor vulnerable conditions is limited	Work with the sensor/collection managers or process owners to troubleshoot/resolve the problem.	Otherwise, revoke or suspend the device's authorization

<sup>8</sup> All software discovered on a device that is not currently on any white/gray/black list will be automatically placed on the graylist. The D/A will decide if the default is treat as authorized or treat as unauthorized.

## Appendix A - Definitions

<b>Term</b>	<b>Definition</b>
Authorized Hardware Inventory	List of authorized hardware assets for an organization or subnet.
Blacklist	List of unauthorized software for a D/A or device.
Common Platform Enumeration (CPE)	Common Platform Enumeration (CPE) is a structured naming scheme for information technology systems, software, and packages. Based upon the generic syntax for Uniform Resource Identifiers (URI), CPE includes a formal name format, a method for checking names against a system, and a description format for binding text and tests to a name. <sup>9</sup>
Device	IP-addressable asset on the network or a non-addressable component (e.g. removable media) able to interact with the D/A's data and resources.
Device Attribute	Device attributes are a way to describe a set of labels, values, and hierarchies associated with dimensions or characteristics of a device. The attributes assigned to a device are used to determine the applicability of a defect check, the result domain of a defect check, or create the appropriate desired state specification for a defect check associated with that device.
Device Attribute Authorization Status	Current state of a device role authorization. States can be 'pending', 'authorized', 'suspended', 'expired', or 'revoked'.
Device Attribute Hierarchy Name	D/As may have multiple hierarchies; the hierarchy that the device role is associated with must be identified to understand association and inheritance.
Device Attribute Hierarchy Status	This attribute will be Boolean and indicate to the capability whether the device role is part of a hierarchy or not. The attribute will dictate to the capability if inheritance based on the device role needs to be considered.
Executable file	For CDM, a specific file in persistent memory that can be loaded into active memory and executed by the CPU.
Expiration Policy	The policy that defines the requirements for when an asset's authorization expires. Examples include a piece of software is authorized for use in the enterprise for 2 years or device role authorizations are good for 3 years.
Graylist	List of software not authorized, but not explicitly prohibited.
Known Bad Blacklist	List of executables that are known to be malicious, fraudulent, or harmful.
Prohibited Software Products Blacklist	List of software products prohibited by policy. This enables a D/A to limit the versions of a software product that are authorized for use in the environment.

<sup>9</sup> <http://nvd.nist.gov/cpe.cfm>

<b>Term</b>	<b>Definition</b>
Scoring	The process of calculating the risk points for a defect. Identified defects will be “scored” based on the amount of perceived risk they create. The CDM program will be providing a scoring system that is generic across D/As. Each D/A may adapt this with additional D/A specific information to better prioritize defects for action.
Software	For CDM, software includes firmware, basic input/output systems (BIOS), operating systems, applications, services, and malware such as rootkits, trojans, viruses, and worms.
Software Asset Management (SWAM) capability	The CDM capability that ensures unauthorized and/or unmanaged software is 1) identified, 2) authorized, and 3) assigned for management, or 4) removed before it can be exploited compromising confidentiality, integrity, and availability.
Software identification tag (SWID)	Software ID tags provide authoritative identifying information for installed software or other licensable item. <sup>10</sup>
Software Authorization Status	Current state of a software product authorization. States can be ‘pending’, ‘authorized’, ‘suspended’, ‘expired’, or ‘revoked’.
Software product	The level of abstraction by which software is typically licensed, listed in registries during installation, and executed by users. Software products are roughly equivalent to the software identified by the NIST Common Product Enumeration (CPE) codes, and also by the ISO SWIDs.
Whitelist	List of authorized software for a D/A or device.

---

<sup>10</sup> ISO/IEC 19770-2: Software identification tag