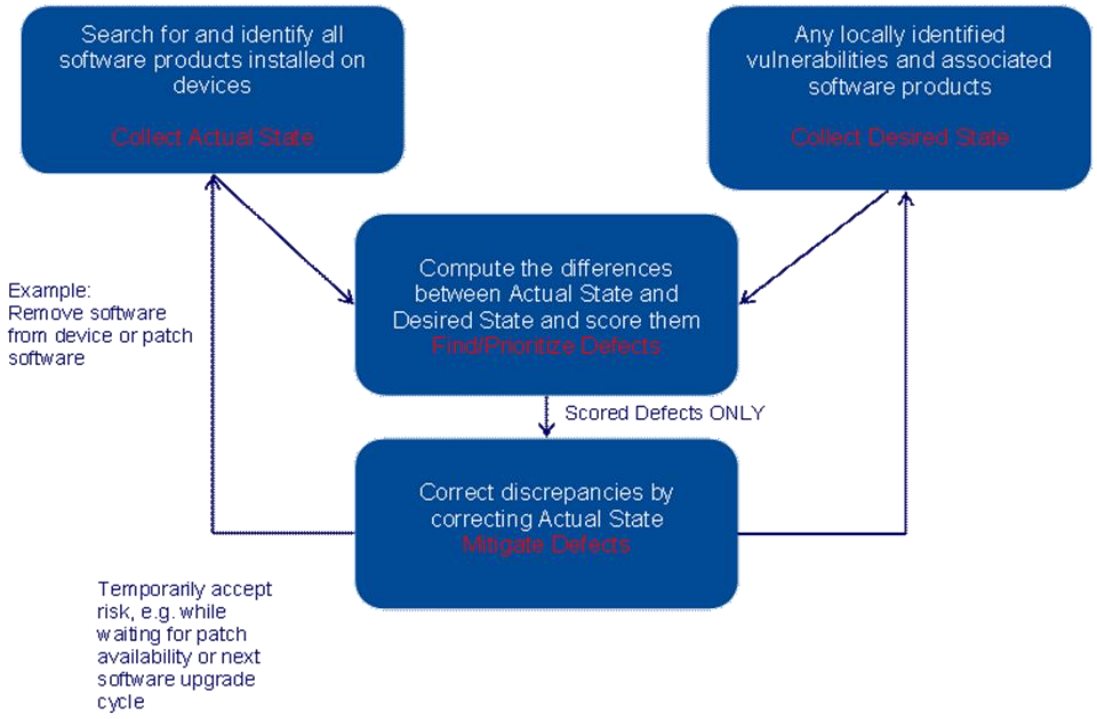
Manage Vulnerabilities (VULN) Capability Description

Purpose

Provides the Department or Agency (D/A) visibility into the known vulnerabilities present on their networks. Known vulnerabilities are those with a Common Vulnerability Enumeration (CVE) identifier or discovered by the local organization and associated with a specific set of software products.

How does it work?

The majority of known vulnerabilities are mitigated by either patching or upgrading software, and most patches/updates mitigate more than one vulnerability. While it is important to highlight the risk incurred by the presence of vulnerabilities, it is more important to accurately reflect the risk ***reduced*** by performing the recommended mitigation. For this reason, the VULN capability determines if devices have software products installed that contain at least one known vulnerability. This can be determined by collecting and comparing the enumerated software product data for each device with current National Vulnerability Database (NVD) information. If known vulnerabilities are present, the D/A will either patch/update the software product or remove it from the device.



How do known vulnerabilities impact the network?

Attackers continually scan devices for known vulnerabilities that can be exploited to gain a foothold into a network. Once a foothold is secured, attackers can exfilitrate sensitive data or launch additional attacks deeper into the network. Attackers also attempt to explolit known vulnerabilities using additional attack vectors such as malicious emails, web browser redirects, or executing embedded software code in the email itself.

Collect Actual State

Because a defect for VULN is the existence of a software product with at least one known vulnerability, the D/A can run tools or processes over the actual state software inventory collected by the Software Asset Management (SWAM) capability.

Tools to identify CVEs on devices (e.g., vulnerability scanners) exist and are widely deployed across most D/A environments. If these types of products are used, their output must be converted to report the vulnerable software, and not the CVEs, that are present. For example, if a particular executable has a CVE associated with it and that executable is associated with three distinct software products, then the actual state information that must be collected is the presence of any or all of the three distinct products not the presence of the one vulnerability.

Just like the other CDM capabilities, you will need to identify how much of the network is being monitored or checked for known vulnerabilities.

Collect Desired State

The main desired state specification required for VULN is a listing of software products (to version/patch level) with associated CVEs generated from existing NVD information[1]. If the D/A has a set of vulnerabilities that are not assigned CVEs but must be managed in their environment, then they must create a similar listing. This listing must contain the software product, the associated vulnerability ID, and a severity value for the vulnerability equivalent to a CVSS[2] score.

Diagnose (By Finding and Prioritizing **Defects**)

Comparing the list of software installed on a device with the list of software products containing known vulnerabilities will identify conditions that need to be addressed. See the Defect Type Table for a list of general VULN defects. After these conditions are detected, they will be automatically scored and prioritized (using federal and D/A defined criteria)[3].

---

[1] Initially, the CDM PMO will work with the CMaaS integration contractors to develop this desired state specification. Over time, it is expected that it will be made available through NVD.
[2] http://www.first.org/cvss
[3] Many defects will have a “grace period” built into the scoring function. For CDM, these grace periods are calculated from the time the defect is first identified, not when the desired state specification or actual state changed.

Mitigate Defects

The CDM dashboard will generally be organized to show worst problems first. Worst problems should be mitigated first. The following table shows the most important defect types and mitigation options. The full set of Defects and mitigations are documented in the *Managing Vulnerabilities Datasheet*.

Defect Type	Detection Rule	Mitigation Options
Device has known vulnerability	Actual State less secure than Desired State	• Update software OR • Uninstall vulnerable software OR • Remove device from network OR • Accept the risk score
Non-reporting	Actual State data unavailable	• Restore collection OR • Remove device from the network

Appendix A - Definitions

Term	Definition
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.[4]
Common Vulnerabilities and Exposures (CVE)	Common Vulnerabilities and Exposures (CVE) is a list of information security vulnerabilities and exposures that aims to provide common names for publicly known problems.[5]
Common Vulnerability Scoring System (CVSS)	CVSS measures the severity of a vulnerability compared to other vulnerabilities so remediation efforts can be prioritized.
Defect	A condition where the Desired State specification and the Actual State do not match in a manner that incurs risk to the organization.
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.
Manage Vulnerabilities (VULN) Capability	This capability is to ensure that vulnerabilities are identified and removed or remediated from devices to minimize exploitation.
National Vulnerability Database (NVD)	NVD is the U.S. government repository of standards based vulnerability management data. This data enables automation of vulnerability management, security measurement, and compliance. NVD includes databases of security checklists, security related software flaws, misconfigurations, product names, and impact metrics.[6]
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 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 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.

[4] http://nvd.nist.gov/cpe.cfm
[5] http://cve.mitre.org/about/faqs.html#a1
[6] http://nvd.nist.gov/

Term	**Definition**
Vulnerability	A flaw or weakness in system security procedures, design, implementation, or internal controls that could be exercised (accidentally triggered or intentionally exploited) and result in a security breach or a violation of the system's security policy.