Software Vulnerability Assessment

Setup Guide

Contents:

About Software Vulnerability Assessment
Setting Up and Running a Vulnerability Scan
Manage Ongoing Vulnerability Scans
  - Perform Regularly Scheduled Scans
  - Adjust the Vulnerability Threshold
Addressing SVA Findings
  - View SVA Scan Findings
  - Act on SVA Issues, Findings, and Events

About Software Vulnerability Assessment

The exploitation of software vulnerabilities is a leading means of attack against networked servers, whether in or out of the cloud. For both compliance and general security reasons, organizations with networked software must ensure that all system and application components are protected from exploits that use known vulnerabilities. Patching those vulnerabilities can help you to avoid malicious exploits, remote buffer overflow attacks, denial-of-service attacks, and other security compromises.

Defending against exploits. Defense against these attacks usually means installing the latest vendor-supplied security patches and upgrades. Scoring and ranking vulnerabilities for the risks they pose to your business is important to prioritizing your remediation efforts. Constant monitoring is also required, to ensure that new threats and vulnerabilities are identified and addressed, and that software changes and upgrades are examined for vulnerabilities in a timely fashion.

Halo Software Vulnerability Assessment is an important component of CloudPassage Halo's security exposure management capability. Halo regularly scans all of your protected servers to detect known vulnerable packages.

These scans identify software vulnerabilities in your servers by comparing the versions of your servers' software packages (operating system, drivers, daemons, and applications) against a number of published vulnerability sources, including the National Institute of Standards and Technology (NIST) database of Common Vulnerabilities and Exposures (CVE). Each resultant vulnerability has a score (assigned by NIST) according to the Common Vulnerability Scoring System (CVSS) and compared against a threshold value that you can set in the Halo portal. Software packages with scores above your specified threshold are considered critical vulnerabilities and are flagged as such in reports.
Here's how a vulnerability scan works:

1. A Halo user logs into the Halo portal to manually initiate a scan or define a schedule for automatic scanning.

2. The Halo analytics engine initiates a scan of each Halo-protected server, on which the Halo agent gathers a list of all installed software packages on the server.

3. The agent sends the list to the engine, which compares each package name to the NIST CVE data and/or data from other sources—including the Microsoft patch database, third-party vulnerability feeds and databases, and proprietary information from CloudPassage's own security research team—and records all detected vulnerable packages.

4. The engine sends the resultant list of vulnerable packages back to the portal, which displays them as scored security events.

5. The Halo user or security admin remediates the vulnerabilities or schedules them for remediation through the organization's patch management program.

**Remediating vulnerabilities.** To remediate detected vulnerabilities, you can take any of these steps:

- Apply the latest patches that address the reported vulnerabilities.
- Remove unnecessary packages that contain vulnerabilities.
- Create exceptions for vulnerabilities that you will address in the near future or that do not pose an actual threat of exploit to your server installation.

**Long-term goal.** Your ultimate goal in performing regular ongoing vulnerability scans should be to lower the number of security issues (reported vulnerabilities) over time. The lower the number of issues that occur—especially critical issues—the more confident you can be that your servers are well protected against software exploits.

---

**Setting Up and Running a Vulnerability Scan**

Perform the following steps to run software vulnerability scans.

1. **Create a Special Events policy.**
   Halo records each software vulnerabilities it detects as a "Vulnerable software package found" Special Event. For information, in the *Halo Operations Guide*, see [Create a Special Events Policy](#).

2. **Perform an automatic or manual scan.**
   For information, in the *Halo Operations Guide* see [Working with Scans](#).
3. **View and act on scan results.**
   - To view SVA scan findings, click the status value of a SVA scan on the Scans screen of the Environment page. In the *Halo Operations Guide*, see View Scan Findings.
     
The scan findings page opens. See Addressing SVA Findings.
   - To view SVA issues, start with the Issues view on the Environment screen, then view the Issue Details Sidebar. In the *Halo Operations Guide*, see View Issue Details.
     
To view details about the Common Vulnerabilities and Exposures (CVE) underlying an issue, in the CVEs area of the Issue Details sidebar, click a CVE entry. For information see View CVE Details.

---

**Manage Ongoing Vulnerability Scans**

Conducting one scan shows you the basics of how to detect and remediate vulnerabilities. Next, consult the strategies presented here to learn how to (1) extract just the valid vulnerabilities of real concern from your scan results, and (2) maintain the most secure stance for your servers going forward.

**Perform Regularly Scheduled Scans**

To maintain a high level of security, it is important to establish a schedule of regular scans of your servers. As the software on your servers changes through updates and new installations, and as new vulnerabilities are discovered and added to the NIST database, frequent automatic scanning will allow you to capture any recently added vulnerabilities.

You can conduct vulnerability scans manually or automatically. For automatic scans, decide whether and how frequently you want to conduct them. Then, from the Site Administrator menu in the Halo portal, select Site Administration and click the Scanner Settings tab.

Under **Scanner Scheduling**, in the line for "Software Vulnerability Assessment", select **Enable Automatic Scanning**, then choose a scan frequency from once per hour to once per week. Leave **Execute scan on daemon start** selected if you want to run an initial scan on each server as soon as it starts up.

Under **Scanner Options** and **Software Vulnerability Assessment**, optionally enter a new value for the...
CVSS score threshold, depending on what score values you think should indicate critical vulnerabilities. See Adjust the Vulnerability Threshold for more information.

The next scheduled scan will occur in as little as one hour or as much as 24 hours later, depending on the frequency you have specified. All active servers, regardless of what server groups they may be in, are scanned at each automatic scan.

Adjust the Vulnerability Threshold

The Software Vulnerability Assessment Settings page lists the CVSS score threshold (default = 5.00). Any vulnerability with a CVSS score at or higher than the threshold will be marked as critical on a server's Software Scan Details page. In general, you should remediate critical events as soon as possible, whereas you might schedule remediation of non-critical events as your workload permits.

You can adjust the threshold downward if you feel that too many events of high severity (in terms of your organization's security requirements) are not being categorized as critical; likewise, you can adjust it upward if you feel that too many events of lesser severity are being unnecessarily classified as critical.

Addressing SVA Findings

To accurately assess the level of risk associated with a given SVA finding, issue, or event, you may need to examine the details of one or more SVA scan findings.

As described in Setting Up and Running a Configuration Scan, the easiest way to examine SVA findings in detail is to click a Scan's Status column to display a summary of the scan's results. From there, you can view the individual findings within a given result.

To accurately assess the level of risk associated with a given SVA event, you need to examine the event's details.

View SVA Scan Findings

By default, the SVA scan results page lists the results of all the all of the vulnerable software packages detected in an SVA scan. To see a full inventory of all installed packages on the server in addition to the vulnerable packages, click View Full Report (with Passes). Halo adds all non-vulnerable packages detected during the scan in addition to the list of vulnerable packages.

By default, critically vulnerable packages are listed first, followed by those with a lower vulnerability score.

The Server Scan Details page differs slightly between Linux and Windows.

SVA Scan Details Page for Linux
Scan details for Windows include the following information not applicable to Linux:

- **CPE.** This is the Windows "Common Platform Enumeration" name for the listed package. It is a standardized NIST specification of the name and version of the software in question. If a NIST specification does not exist for the package, the value in the CPE column is "No NIST CPE".

- **Security updates.** Clicking the View Security Updates link at the top of the table displays the following page relating to the software installed on this server:

This table is an inventory of all the security updates (patches) that have been applied to the server. Each update is identified by its "kb" number, the unique ID of the Microsoft knowledge base article that describes the update. The installation date for the update is displayed, if it is known. The CVE(s) that the update addresses are also listed. This information is useful for verifying whether a specific update has been applied to the server, or for quickly determining which CVEs a given update addressed.

**Note:** The set of CVEs associated with a given update in the table includes not only CVEs that were explicitly addressed in that update, but also all CVEs that were addressed in earlier updates that have been superseded by that update.
View CVE Details

To view the Common Vulnerabilities and Exposures (CVE) reference number for a vulnerability (Windows or Linux), in the list of Scan Details, in the **CVE Reference** column click a CVE entry.

![CVE-2010-0624](image)

**Vulnerability Details**

Heap-based buffer overflow in the `mrt_read` function in lib/raspipe.c in the `mrt_client` functionality in GNU tar before 1.23 and GNU zip before 2.11 allows remote `mrt` servers to execute arbitrary code by sending more data than was requested, related to archive filenames that contain a `:` (colon) character.

*Published*: 2010-03-15 13:28:25  
*Last Modified*: 2012-02-02 03:02:20

**CVSS Base Metrics**

- **NIST CVSS Base Score**: 5.0  
- **Access Vector**: NETWORK  
- **Access Complexity**: MEDIUM  
- **Authentication Required**: NONE  
- **Confidentiality Impact**: PARTIAL  
- **Integrity Impact**: PARTIAL  
- **Availability Impact**: PARTIAL  
- **Source**: http://nvd.nist.gov

You can also use the Issue Details sidebar to view CVE details. For information, in the **Halo Operations Guide**, see **View Issue Details**.

The information on the CVE details page comes directly from the NIST database.

- **Vulnerability Details**: Where and in which versions of what software modules this vulnerability exists, what kind of exploit it allows, and possibly additional details on how an exploit could work.

- **Published** and **Last Modified**: How long this vulnerability has been known, and whether/when NIST has updated this information.

- **CVSS Base Metrics**: Important information on how severe this vulnerability is:
  - **NIST CVSS Base Score**: The numerical severity score for this vulnerability. By default, Halo classifies any score over 5.0 as a critical security event, although you can change that threshold on the Software Vulnerability Assessment Settings page in the portal.
  - **Access Vector**: How remote an attacker can be. A value of **Network** means that the attacker does not need physical access, local-account access, or local-network access to the target.
  - **Access Complexity**: How common the conditions are in which this vulnerability can be exploited. A value of **Low** means that it is very common.
  - **Authentication**: Whether and how many times an attacker must authenticate to the target system to perform the exploit. A value of **NONE** means that no authentication is required.
  - **Confidentiality Impact**: The impact that a successful exploit could have on the confidentiality of the target. A value of **Complete** means total information disclosure.
  - **Integrity Impact**: The impact that a successful exploit could have on the integrity of the target. A value of **Complete** means total compromise of system integrity.
  - **Availability Impact**: The impact that a successful exploit could have on the availability of the target. A value of **Complete** means a total shutdown of the affected resource.

All of the above information is taken into account in assigning the numeric score to the vulnerability. For more details, click the **NIST FAQ on Base Metrics** link.
**Affected Packages.** The name(s) of the package or packages that exhibit this vulnerability. This is the CPE (Common Platform Enumeration) of the affected Linux or Windows package.

- **External References.** A list of links to additional information about this vulnerability.

Take both the numeric score and the values for the individual CVSS base metrics into account in assessing your organization’s priority for remediating each vulnerability. The next section discusses remediation strategies.

---

**Act on SVA Issues, Findings, and Events**

Following a scan that reveals potential software vulnerabilities in your servers, your first task is usually to apply all known patches, bringing your servers up to date. Then you can take further steps to identify and remove any apparent issues that do not constitute a real threat.

**Apply the Latest Patches and Re-Scan**

If you are using golden images, you should update those images with every major patch upgrade — no matter what.

To address the reported software-vulnerability issues in your servers, first upgrade your operating systems and critical applications to include all the latest patches. The recommended practice is to patch your gold master servers then re-instantiate as many servers as possible from the masters: Reinstantiation works well for larger, fluctuating pools of high-volume servers such as frontend web servers, but may be impractical (not to mention expensive) for other types of servers.

Alternatively, you could use automation tools such as Chef, Puppet, or RightScale to re-provision all your servers with the latest software.

After patching, re-scan your servers to learn what vulnerabilities remain. Then analyze them to separate the true unpatchable vulnerabilities from issues that are false positives or that may not be of concern to your particular server installation.

**Delete Unnecessary Packages**

Some of the software vulnerabilities detected by Halo may occur in packages that your organization does not use. For example, common administrative tools supplied with many distributions may have vulnerabilities; if you do not use those tools on the server, you should remove them.

One way to address such a vulnerability might be to define a vulnerability exception for it, so that it does not appear in future scans. But a simpler and more secure solution is to simply remove the file from your servers.

If your servers never execute such a file and have no need for it, delete it from all affected servers and from any gold-master server templates that you use for creating new server instances.

**Specify Exceptions**

Halo may detect software vulnerabilities in certain software packages that you may not need to remediate at the present time. For example:

- You may have scheduled an upgrade or patch in the near future, but you do not wish to or cannot perform that upgrade immediately.

- You have a compensating control that addresses the vulnerability in some other way. For example, if a vulnerability were reported in `httpd`, but your firewalls already disallow any HTTP traffic to or from the outside, the vulnerability may be less of a concern.

- The vulnerability may appear to be valid based on the package version number, but in reality it already has been patched with an updated package. (This situation can occur because vendors sometimes update a package without actually changing its version number.)

- You may have recently upgraded your kernel or some other package for which the installer does not remove the
older, vulnerable version when it installs the newer one. Subsequent scans will continue to detect that vulnerable package, even if it is not running. You may not wish to delete the older package, but you'd prefer that it not continue to appear in your scan results.

You can keep those events from showing up in your future scans by defining a software exception for that package. The exception temporarily or permanently suppresses the reporting of that event, possibly until a time at which you expect to have remediated the issue by patching or upgrading the package.

To create an exception, click Add Exception in the line describing the event on the Software Scan Details page for a server. The Add Software Scanning Exception dialog opens:

Specify when, if ever, the exception should expire. Decide whether it should apply to just the server on which it was reported, or to all servers in its server group, or to all of your Halo-protected servers. Optionally include a reason for the exception, so you'll have a record of why the exception is justified.

All defined software exceptions are listed on the Software Exceptions page in the Halo portal, at Policies > Software Exceptions. If you want to delete an exception before it expires on its own, click the Remove icon (x) for that exception on the page.

Address Remaining Unpatched Vulnerabilities

The purpose of software vulnerability assessment is to allow you to find and fix software vulnerabilities in your cloud servers. Halo will show you where the vulnerabilities are; once you have applied all known patches, removed vulnerable packages that you do not need, and created exceptions to hide issues that do not concern you, what
remains—if anything—is some number of vulnerabilities for which no patch is currently available.

You should handle those vulnerabilities by following the recommendations in your organization’s security policies. You may decide to stop using the vulnerable packages and remove them from your servers, or you may create compensating controls to protect the packages in other ways. You might even wait for a patch to become available, if you expect one soon. In any case, it is important to take steps to minimize your organization’s exposure to vulnerability-related risk, both at the present time and going forward.