Hewlett Packard
Enterprise

Business white paper

Closing the vulnerability gap

Continuous Monitoring

Table of contents

Organizations face the daunting task of continually remediating all vulnerabilities in their IT environment. Central to this is evaluating the results of their monitor, remediation, and re-test paradigm for assets and determining if a vulnerability has been truly remediated.

In the last 10 years, on average 4,660 vulnerabilities were disclosed per year, with an all-time high of 6,462 vulnerabilities counted in 2006 followed by a continued decrease for the next five years down to 4,139 (64% of the all-time high) in 2011. However, in 2012 alone, the number of vulnerabilities increased again to a considerable 5,225 (80% of the all-time high), which is 12% above the 10-year average. This is the largest increase observed in the past six years and ends the trend of moderate declines since 2006.[1]

Understand the trend is real

According to Microsoft's Security Intelligence Report,[2] at the end of 2011, vulnerabilities were trending downward. This was very good news to most; however, in 2012, the industry noted the opposite—a considerable increase in vulnerabilities. With new innovative systems developed—mobility software, cloud environments, and the like, systems with security weaknesses and exposures remain. How industry handles these vulnerabilities will determine who wins the IT battle—hackers or IT security managers who protect their company's systems and data.

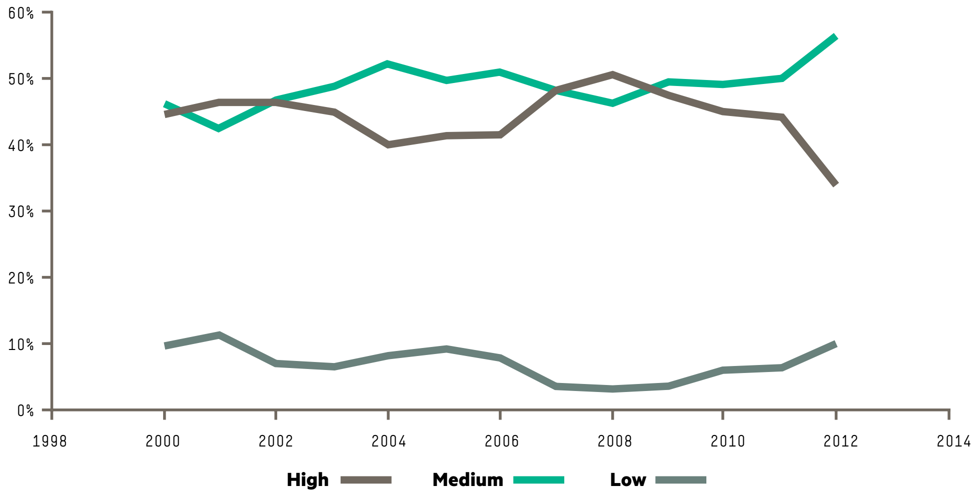

Figure 1
Industry-wide vulnerability disclosures and criticality distribution[3]

[1] Vulnerability Threat Trends, NSS Labs Stefan Frei, Ph.D. 2013

[2] Ibid

[3] Ibid

So who's the enemy? As of this writing, the Syrian Electronic Army (SEA) had hacked nine websites, including the *New York Times* and Twitter.[4] The SEA said it had previously hacked other media outlet websites as a warning shot to the United States (U.S.). They further stated that the U.S. could expect more attacks. In the spring of 2013, a global network of hackers broke into vulnerable bank databases, eliminated withdrawal limits on pre-paid debit cards, and created access codes. They then loaded that stolen data onto any plastic card with a magnetic strip and fanned them out over multiple cities to withdraw $45 million USD. This is what the IT industry is up against. Hackers are no longer lone college kids trying to steal an ID to change a grade. They are highly organized, extremely technical groups of individuals that have political agendas or a strong desire to steal money.

The National Institute of Standards and Technology (NIST) stated that 80% of all external attacks take advantage of known vulnerabilities in unpatched or misconfigured systems. Most industry experts would put this number at 90% or higher.

Learn what can be done

The compromised systems were left vulnerable by the IT personnel whose responsibility it was to protect them. Many were exploited through Denial of Service attacks (DoS), Domain Name System (DNS) credential hacking, or poorly protected databases. The State Department has reported that 80% of external attacks leverage known vulnerabilities and configuration management setting weaknesses.[5] Industry experts agree that best way to defend against these attacks is to ensure your systems are not vulnerable. Yet, how can this be done? And, more importantly, how can this be accomplished continuously?

In 2002, the United States Congress passed the Federal Information Security Management Act (FISMA)[6] with the goal of improving the security posture of IT systems in the federal government. This checklist approach required many man hours of manual audit tasks, including assessment, answering questions, and then reporting—all of which took a great deal of time and money. This rudimentary approach does not meet the needs of today's dynamic network of systems. When a user fails to patch a system, install a security update, or actually downloads a piece of malware, the system is suddenly vulnerable.

The FISMA guidelines were a good start. They provided a baseline for compliance. Government organizations began to put the approach into a repeatable process. With as many systems as were present on typical operational federal IT networks and their state of flux, it was obvious that a real-time automated approach was necessary. This is when NIST developed Special Publication 800-137, which included the concept of Continuous Monitoring (CM).

Understand Continuous Monitoring

NIST defines Information Security Continuous Monitoring (ISCM) as "maintaining ongoing awareness of information security, vulnerabilities, and threats to support organizational risk management decisions."[7] The key concept is to address vulnerabilities and understand how they negatively impact an organization's risk posture. ISCM requires a continuous view into an organization, with the ability to report key metrics in near real time. It also ensures continued effectiveness of security controls and maintains awareness of threats and vulnerabilities with respect to an organization's assets.

If systems are to continuously be kept from being vulnerable, there must be a robust definition of the concept. NIST defines a vulnerability as a weakness in an information system, system security procedures, internal controls, or implementation that could be exploited by a threat source. The vulnerabilities are broken into two predominant areas—a software flaw and misconfiguration. A software flaw is a piece of written code that (unintentionally) exposes an entry point to a hacker and primarily is corrected with a new update or software version. A misconfiguration is a piece of software that has external parameters that govern the software's operation, set incorrectly, to enable illicit entry into the system. Either of these situations makes the system vulnerable. Also, if there is a piece of software that isn't patched, an anti-virus agent is out of date, or a system is misconfigured out of policy, all of these issues will make the system vulnerable to be exploited.

[4] NY Times Caught in Syrian Hacker Attack, Mathew J. Schwarz, August 28, 2013, http://www.darkreading.com/attacks-and-breaches/ny-times-caught-in-syrian-hacker-attack/d/d-id/1111318

[5] http://www.state.gov/documents/organization/156896.pdf page 7

[6] Federal Information Security Management Act (FISMA) DHS, http://csrc.nist.gov/drivers/documents/FISMA-final.pdf

[7] NIST 800-137—Information Security Continuous Monitoring (ISCM) for Federal Information Systems and Organizations, http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-137.pdf

When a CM approach is used, three items continually happen in the environment to ensure security; vulnerabilities are identified, remediated, and re-tested for compliance according to the organization's policies. This work cannot be performed manually. CM systems must be able to share data and provide insight into important high-risk areas—efficiently and as quickly as possible. To facilitate this machine-to-machine (M2M) interoperability, NIST created the Security Content Automation Protocol (SCAP) pronounced "esKap."[8] It facilitates automation by standardizing communication formats and using common naming vulnerabilities. Software flaws are Common Vulnerabilities and Exposures (CVE) and misconfiguration items are Common Configuration Enumerations (CCE) under the NIST standards. At the time of this writing, SCAP version 1.2 is the latest and comprised the most recent updates to an assortment of standards.

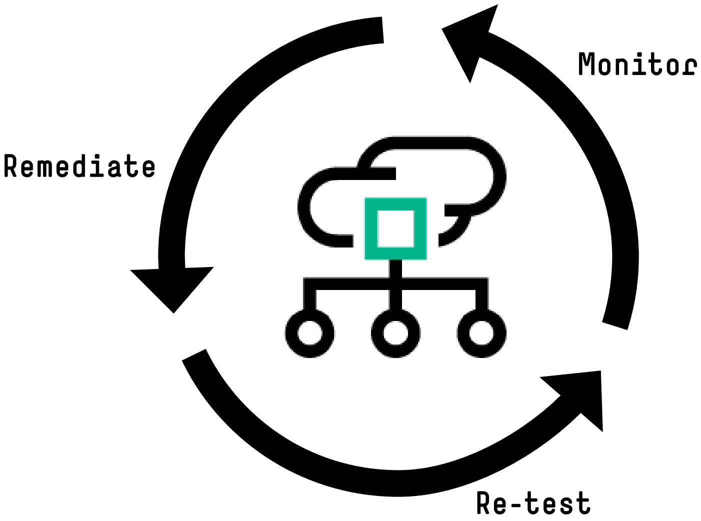

Figure 2
Continuous cycle of monitor (identify), remediate, and re-test

Use CM to keep systems from becoming vulnerable

First and foremost, systems must be continuously checked and re-checked (monitored) for exposures. Most companies today perform this task—some better than others, but most monitor their environments for vulnerabilities. They also do a fair job of remediating or mitigating these issues. Organizations use tools such as Tivoli Endpoint Manager or System Center Configuration Management (SCCM) to push patches to the asset, or manually log in to remediate a configuration. The problem arises with re-testing the asset for the vulnerability that was found. How does one really know when it has been fixed? How can this be evaluated quickly without a large impact on the network?

Most vulnerability scanners today report when a vulnerability exists, but do not report when a vulnerability has been fixed. Confirming the fix is a bit more subtle, as the scanner will no longer report the vulnerability for that asset—because it's not found. Consider the following fictitious scenario depicted in Table 1. Jordan is a system administrator for the Widget Company that recently ran a scan and identified CVE–1234 on a set of Windows systems.

Table 1. Vulnerability data example

Asset ID	Operating system	Vulnerability	POC
198.162.1.1	Windows 2008R2	CVE-1234	Jordan Smith
198.162.1.9	Windows 2008R2	CVE-1234	Jordan Smith
198.162.1.92	Windows 2003SP1	CVE-1234	Jordan Smith

Note: Example data was obtained from running a scan on Windows Assets. Data is a subset and only for example purposes.

[8] http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-126r2.pdf

This CVE was reported to the system administrator for a group of assets on the network; a patch was pushed and manually configured these systems to remediate this vulnerability. Jordan then runs a scan for a second time. The results omit any mention of CVE-1234 with those assets. Does this mean they are fixed? Did something go wrong? How does one know they have been fixed and fixed properly? The answers to these questions will only be known to Jordan, who fixed the problem on the devices in question.

How do dashboards handle this implied information? Most dashboards today incorporate a setting that removes the vulnerability after a period of time. Something similar to "remove vulnerability if not found in 'X' days." Many default to 14 days. Mobile environments must also be considered. With the number of smart mobile devices increasing at a very high rate and, the advent of bring your own device (BYOD), wireless and mobile devices are assets that also need to be monitored and protected. Mobile assets will continuously be moving in and out of various organization networks, and their footprint changes rapidly. Although scanning of these mobile devices is in its infancy, in November of 2012, there were 387 documented vulnerabilities found for the Apple IOS.[9] This includes iPhones and iPads. At the end of 2013, the HP Cyber Risk Report showed that malware had been downloaded tens of millions of times from Google Play by Android users.[10] These devices should be scanned and patched in a similar manner as nonmobile servers and client devices are done today.

Get the ultimate solution

Under SCAP 1.2, NIST developed the Assessment Report Format (ARF)[11] to help facilitate M2M interoperability, develop a common understanding of vulnerability and configuration information, and provide an avenue to solve this re-test issue. The ARF XML file primarily supports Asset Identification (AI standard) content areas, vulnerability, and configuration (CVE and CCE, respectively), and results (pass or fail). The focus must be on the results in order to provide some insight into resolving the re-test challenge.

The results use eXtensible Configuration Checklist Description Format (XCCDF)[12] results that include pass, fail, not applicable, not checked, not selected, informational, and fixed. These statuses are conveyed to the organization's IT team to identify problems and issues with an asset. The notion of results content indicates the importance of providing a status other than just the vulnerability ID (that is, CVE 1234). The addition of a "fixed" result also shows the option to indicate that a vulnerability has been definitively resolved.

Some solutions have suggested a need to report on all vulnerabilities for all assets all the time. This would mean that the scanner vendors would report a pass or fail result (or other status) for every vulnerability that may exist on that asset. For IT management working in these ecosystems, they know that transporting, handling, and storing unnecessary data adds no value to the current CM process.

Pushing logic up to the dashboard isn't the best option either and will prove to be a kludge. Indicating the status and vendor-specific status would help pressure scanner vendors into action. Dashboard vendors could place text in place of, or near the results, that states this value has failed and has not received a fixed notification from the vendor scanner. This solution again, is not optimum, but should drive more informative results.

[9] CMaaS Mobile VM Future Products, slide 5, Mike Kelly, Hewlett-Packard Development Company, L.P.

[10] HP 2013 Cyber Security Report, page 7, copyright 2014, Hewlett-Packard Development Company, L.P.

[11] The Asset Reporting Format (ARF) http://csrc.nist.gov/publications/drafts/nistir-7848/draft_nistir_7848.pdf

[12] The eXtensible Configuration Checklist Description Format (XCCDF), https://scap.nist.gov/specifications/xccdf/

Understand how to get there

Ultimately, with increased use of the ARF standard, the vulnerability gap will be eliminated. Adoption of it, however, has been slow by some. There have been several versions created prior to the published ARF 1.1 standard, and many of the scanner vendors have been caught trying to implement various versions. Some have claimed to support NIST 1.1 of ARF but have failed NIST Content Validation.[13] This will soon change. The ARF 1.1 standard is well-written and has unified support of NIST. Top government agencies are requiring its use, as it has versatility to provide results for CVE and CCE information. Community and government pressure on scanner vendors will also help as most scanner vendors understand the problem and have a strong desire to invest resources where there is customer and community concern and focus. Hewlett Packard Enterprise recognizes the importance of following NIST's lead and implements these standards into our CM security solution.

See the future

Vulnerabilities are not going away. Statistics show they continue to be more widespread than ever before. New mobile platforms are in use and introduce other vulnerability complexities. Systems must continue to be monitored, remediated, and re-tested—more effectively and efficiently. This can be achieved by using standards laid out by NIST and encouraging their adoption. Scanner vendors must be inspired to get smarter about how they scan and provide targeted results. Dashboard vendors must re-think how they use and display scanner data and enable customers with a more flexible approach for visualizing test and re-test. If scanner vendors won't adopt a knowledge of previous test results, dashboard vendors may have to add their own rule logic or display their own resultant information to facilitate closing the vulnerability gap.

Work with the best

Over the years, HPE's commitment to securing the enterprise has been proactively demonstrated with market-leading, innovative services and products. Consistent with this relentless dedication to advancing state-of-the-art security practices, HPE has invested millions of dollars in establishing a CM demonstration lab to better quantify the challenges of deploying a CM solution framework. This hands-on lab experience has given the HPE CyberSecurity Group unique insights into CM, and the intent of this paper is to share our knowledge with our valued go-to-market partners and customers

Learn more at
hpe.com/gov/security

[13] NIST Content Validation Tool,
http://scap.nist.gov/revision/1.2/index.html

About the author

Pete Schmidt brings more than 20 years of experience in the computer technology field for public and private sectors. Schmidt has a breadth of experience focusing on cybersecurity and application systems development. He has worked in complex federal environments bringing innovative solutions to provide comprehensive successful outcomes for clients.

Sign up for updates

The examples depicted in this document are conceptual in nature and may not be an actual representation of a production implementation.

4AA5-3762ENW, October 2016, Rev. 3