The Journey from Unmanaged to a Managed Risk Management Program

Kevin Thompson

East Carolina University

ICTN 6823 Information Security Management

Dr. Lunsford

July 21, 2020

Abstract

Over the five years my organization has pursued becoming more organized in their information security program. To develop an effective program, the organization must start with the basics. Identify and categorize all company assets to effectively apply desired controls later in this process. Next, you must identify reoccurring maintenance windows and communicate with the asset owners to ensure maintenance awareness. Identify and configure security applications that will apply remediations. Identify and decommission all stale assets to eliminate unnecessary risk in the environment. Once all the previously mentioned steps are complete the security management program is at the beginning stages to become an effective tool to lower risk in the environment.

The Journey from Unmanaged to a Managed Risk Management Program

Launching a vulnerability management program is one thing but launching a vulnerability management program and growing it into a mature and successful program is both laborious and strenuous. One major hurdle to overcome is the idea that patching is easy. Many believe you can follow the mantra "Just Patch!" and all vulnerabilities will be taken care of. Although, many wish this were true, there are many wrinkles that need to be ironed out to make a program successful. Unfortunately, if the organization does any business transactions over the internet or on a computer a vulnerability management is necessary to safeguard the business. The size of the organization and the current level of documentation will drastically affect the level of effort required to build the program into a success. Although, it doesn't matter if the organization is small or large, it only takes one asset to become compromised to allow the attackers in the network. Data breaches and cyber-attacks are in the news every week. Having a strong vulnerability management program is the best way to ensure that doesn't happen. An alarming 57% of cyberattack victims report that their breaches could have been prevented by installing an available patch, according to a new ServiceNow study conducted by the Ponemon Institute. And 34% of those respondents were already aware of the vulnerability before they were attacked (Kent, 2020). More recently, in 2009, the Conficker worm exploited a network service vulnerability in the Windows operating system and impacted a staggering 20 million computers worldwide, including those in the French Naval System and the British Ministry of Defense. Once again, interestingly, Microsoft had already released an out-of-cycle patch on an emergency basis to address this vulnerability as early as October 23, 2008. Still, nearly three months later, close to one-third of all systems remained unpatched in January 2009 when the worm wreaked the maximum amount of damage globally (Dey et al., 2015).

A vulnerability management program is imperative for several reasons:

- Documentation: Ensure every asset has a use and an owner. Ensure any unused systems get decommissioned. Keep the documentation up to date as changes happen throughout the environment.
- Security: Vulnerability management remediates vulnerabilities and misconfigurations in your environment that could lead to a security event. Removing these vulnerabilities and misconfigurations reduces the organizations risk.
- Software versions: Vulnerability management programs ensure only the latest and safest software version are used. This leads to safer applications, but also ensures applications have the newest and most stable features.

Those listed above are just general reasons to have a vulnerability management program. The organization must decide what is important to them. This can vary from organization to organization. Generally, vulnerabilities are given a score to determine the severity of the vulnerability. A recognized framework is the Common Vulnerability Scoring System (CVSS). The CVSS is well suited as a standard measurement system for industries, organizations, and governments that need accurate and consistent vulnerability severity scores (Vulnerability Metrics). Using a vulnerability scoring system will help prioritize the work that needs to be done to quickly lower risk in the environment. The higher the score, the more risk that asset is posing to the organization. Many of these frameworks will also rate the vulnerability on what level the attacker will need to be. The organization would want to remediate vulnerabilities that can be exploited by a novice or intermediate before they moved on to expert level vulnerabilities. There will always be some level of risk in the environment. The organization must prioritize risk and attack it in and order of priority by establishing clear objectives and by outlining important milestones.

Establishing Vulnerability Management

Vulnerability Management Tools

When launching a vulnerability management program, it is important to have an agreement on what tools will be used to secure the environment. Some popular tools are SolarWinds Patch Manager, Microsoft SCCM Patch Management, Ivanti Windows Patch, Windows Server Update Services (WSUS). There are many tools out there, so the organization will have to do their due diligence on what works best in their environment. Many tools incorporate third party applications that will need to be considered too. Many of the top vulnerabilities are from third party applications that many organizations find essential to run their business. Applications like Google Chrome, Mozilla Firefox, Adobe Acrobat, and Oracles Java will considerably increase the amount of vulnerabilities that will need to be remediated. A popular option is to perform a proof of concept to give a program a test drive before fully committing to using it fulltime. Each tool is going to bring different levels of robustness and complexity. It depends on the size of the organization and the degree of configuration and automation they would like to support. Programs that assist in the process of patching make it easy to scale. The time it takes to patch one asset can be scaled to be completed on one thousand assets in the same amount of time. This is invaluable when you need to patch and reconfigure numerous assets in a small window of time. These types of programs also allow administrators to ensure only certain patches get approved. It is important to be able to quickly evaluate which updates are critical, which ones are merely useful and which ones are unnecessary (Voldal). After determining what patches are absolutely needed to secure your organization, the administrator must also research if the patch has caused any issues in the community. If a patch has known issues, a decision must be made whether to hold off until a resolution is available. The organization must find a balance between security and reliability.

Thankfully with diligent research and a thorough backup strategy, finding a happy medium should not be a difficult proposition.

Role and Responsibilities

When starting a vulnerability management program, it is critical to define and assign roles and responsibilities. The responsibilities at a high level, would be vulnerability remediation administrator, security analyst, and project manager. The information security team should provide guidance on any out of band security remediations that need to be remediated out of their normal maintenance window. They will also normally be assigned with providing the reporting to the vulnerability remediation administrator. The vulnerability remediation administrator will oversee addressing vulnerabilities in the maintenance windows. The project manager would tie it all together and ensure that the vulnerabilities are heading in the right direction. If the application owners require testing after security patching, quality assurance analysts are needed to perform smoke tests to ensure the application is running as it should after patching. Roles and responsivities need to be clearly defined and must have upper managements support to be successful.

Find Ownership

The first step after realizing that a vulnerability management program is needed and engaging the right amount of resources to make it successful is to figure out who owns what. Finding owners for each asset is crucial in building a great program. Once the organization can find owners, establishing a good working relationship is also vital to making constructive and continuous progress. The organization must establish and keep a good systems inventory. If you don't have an accurate inventory of all the software and hardware elements connected to a network, it becomes incredibly difficult to ensure that all applications and devices are kept patched (MSP). Once every asset has been assigned an owner you

can remove systems that are not being used and collaborate with the owner to find a maintenance window for all productive assets

Decommission unused assets

By getting a good handle on asset inventory it will reveal any assets that may not be being used. These vampire servers sit idle and consume resources and increase your attack footprint. Identifying these and moving toward decommission safely can go a long way in reducing vulnerabilities and increasing resources. By ensuring a proper decommissioning plan and knowing who owns each asset, you can minimize your organization's chances of falling victim to the next breach (Parthena White, 2020). Reaching out to owners and getting signoff is the best way to move forward with a decommission. If no owner can be found, the asset can be powered down or the network cable disconnected to analyze negative impact. Typically, if nothing is noticed in two weeks, the asset is not providing any value and can be decommissioned. Coupling this process with a short-term backup plan can ensure that no server that is providing value is permanently decommissioned unintentionally.

Create Maintenance Windows

Once owners for each asset are established, they can provide the best time to do maintenance on each asset. In a perfect world, the fewer maintenance windows the better. It's recommended to perform Windows patching on a monthly basis, not quarterly. Prepare a schedule starting with development than user acceptance testing (UAT), production, and disaster recovery. Using this schedule, patching the assets within four weeks of time span is best practice (Arban, 2017) . As a guidance we recommend our customers to aim for 14 days SLA – in order to get ahead of most exploits. The Verizon DBIR from 2016 states that "Half of all exploitations happen between 10 and 100 days after the vulnerability is published", the same report from 2019 states that "Every time a vulnerability is

disclosed or a system update or patch is released, a hacker sees an opportunity. They research the disclosure or update notes to learn if they can exploit the vulnerability and where, searching for their best opportunity to monetize the vulnerability“(Livne, 2019). Automated updating is an important component of patch management, but automation brings its own set of issues for administrators. Updates during business hours can obviously introduce problems by creating performance loads on PCs when they might be needed most. However, scheduling all updates for 2:00 a.m. isn't a solution either because thousands of machines simultaneously downloading large patches could overload the organization's network connections. Distributing update times across nonbusiness hours seems like a simple solution, but not all applications have the same volume or size of updates: some might have large, frequently released patches, whereas others might require occasional updates. Allocating update times to minimize system load and reduce the risk of disrupting operations requires a careful review of patch frequency, plus knowledge about patch size averages and distributions for enterprise applications. This schedule should also factor in the need to reboot after patch deployment (Liu et al., 2009).

Documentation and Communication

To ensure everyone is on the same page, administrators should notify all stakeholders and/or application owners via email, before and after completion of the patching activity so that they can carry further application level testing to make sure applications are working as expected (Arban, 2017). Providing and documenting the details of the program can help eliminate any confusion by establishing a maintenance rhythm that administrators and application owners can become accustomed to. Creating a document that the organization can reference for ownership and maintenance windows would ensure everyone is on the same page. Just be sure to make it readily available, even during network outages.

Change Management

After the patch has been approved, tested in lower environments and is ready to be deployed to production, the proposed changes to systems and the results of the testing should be documented and approved by system owners. This will help safeguard against poor quality changes that could cause an unnecessary application outage while at the same time increase visibility. Only then should changes be made to production systems (Voldal). The approved and implemented change requests can be referenced later if needed. A good change management program will reduce the chance of an unauthorized change, reduce unplanned outages, and improve reputation of the teams that continuously close changes without negative impacts. When change control and vulnerability management program work together, security administrators can be more effective and productive.

Vulnerability Remediation

After completing all the important steps that lead up to vulnerability management, it is finally time to reduce the risk in the environment. All patches should have been approved prior to the maintenance window. Using the tools designated to patch, patches and security remediations will be pushed to all assets that are in the scope of the maintenance window. This included OS patches and all third-party software patches. Once they are successfully installed the device will need to be rebooted to complete the install process. When the asset(s) comes back online, they can be confirmed complete. Occasionally, multiple rounds of security remediations are needed. It is good to be mindful of the amount of effort needed to stay within the maintenance window. Once all assets have been confirmed to have the necessary remediations, a communication can be sent to advise pertinent groups to test their systems, if necessary. Assets that could not be completed need to be communicated too. These systems will need to be researched and completed as soon as possible while working with the asset owner to find a downtime to perform the work. At the very least, the issue should be researched and scheduled for special treatment in the next maintenance window.

Reporting

Once you have a vulnerability management program in place and you believe vulnerabilities are being reduced, you need a way to verify a reduction in vulnerabilities. A good reporting tool will identify assets that might fall through the cracks, it will showcase vulnerabilities that are not disappearing, and allow the organization to plan on how to reduce vulnerabilities in the future. In many cases the vulnerability scanning tools will provide the reporting needed to reduce the risk in the environment. Good reporting will allow the asset owners and security administrators to collaborate and prioritize vulnerabilities to work on. Without a good reporting structure, the organization can't get a grasp on if they are eliminating risk from the environment. It is easy to get into the routine of patching and believe the maintenance windows are making an impact. The truth is vulnerabilities are constantly being created and the vulnerability remediation program could just be maintaining a status quo risk level. One way to combat this is to create and maintain a vulnerability timeline. Use a risk score to check that work is being done to constantly maintain a downward trend. Maintaining a graph or pertinent statistics to demonstration the work that has led to a reduction in risk in the environment.

Constant Improvement

Vulnerability and risk management is an ongoing process, and it should continuously adapt to the evolving cybersecurity threat landscape. Therefore, the process should be reviewed on a regular basis, and staff should be kept up to date with the latest threats and trends. Continuous development for the people, processes and technology will ensure the success of the enterprise vulnerability and risk management program (Arampatzis, 2019). Investing time to research and understand how to remediate complex vulnerabilities is key to engineering solutions that will reduce risk without causing impact. Incorporating other methods to reduce risk in the environment is key. Adding a standard build so new vulnerabilities are not constantly being added to the environment will help. Learning from each

maintenance and documenting problems that arose and attempt to improve on the process the next time. Eventually the process will be so streamlined or automated that effort can be placed in other projects.

Challenges

When establishing a vulnerability management program, there will undoubtedly be challenges to overcome.

- Backups: The organization must ensure that there is a road to recovery if anything unintended happens during the maintenance. This could be snapshots in a virtual environment or backups in either a virtual or physical environment. Disaster recover if possible, needs to happen within the maintenance window.
- Incompatible software: Some applications will not run on the newest level of patches. This will need to be researched and dealt with once discovered. An exception will have to be granted if the organization is going to accept the risk.
- Staffing: There may not be enough resources to remediate vulnerabilities in a timely manner. In many cases the application owners are busy working on the application and don't have much time to focus on remediation.
- 24/7 Business: Many organizations provide services around the clock. Taking a server down to do maintenance means potentially losing money. The organization has to consider when downtime can happen to complete a maintenance window.
- Accept Risk: Accept the risk posed by that vulnerability and do nothing (Rapid7).
    - This is a last resort. This would only be accepted if all other avenues have been exhausted. The organization would need to document an exception and a senior level executive would have to sign off on this.

Listed above are some of the most common challenges organizations will run into while establishing their vulnerability program. The key is getting everyone together and working through each problem to come up with a solution that works for everyone. Or almost everyone. As long as it is a collaboration done in good faith the group can work together to immediately produce results.

Establishing a successful vulnerability management program is a vital, but challenging aspect in protecting an organization from data breaches and cyber-attacks. No organization can afford to ignore vulnerabilities in their network. Doing so would almost guarantee that a cyber-attack would occur. Measures should be taken by the organization to reduce the risk of a possible attack. Everyone in the organization must work together using a trustworthy reporting strategy to protect against emerging threats. Software bugs and vulnerabilities are consistently discovered in both operating systems and applications. It requires the whole organization from the top down to get on board for the program to be successful. By following the steps laid out above, the organization can put itself in a strong position to reduce risk to acceptable levels. Constantly improving on the process will allow the organization to reduce its risk rapidly and lower the effort needed to be effective. Patch management ensures that the appropriate patches are applied to the appropriate systems, in the appropriate maintenance windows. Vulnerability management verifies that assets are no longer vulnerable to known threats after remediation. Getting to a mature vulnerability management program and combining both to provide the organization a standardized patched and remediated environment, leading to a more stable and secure IT infrastructure that can be viewed as a proactive effort, instead of a reactive measure. The journey from unmanaged to managed risk management program is not an easy one, but a meaningful one.

References

Arban, M. (2017, October 23). *Windows Server Patching: Best Practices*. https://social.technet.microsoft.com/wiki/contents/articles/43406.windows-server-patching-best-practices.aspx.

Parthena White, H. (2020, March 18). *Best Practices in Designing a Data Decommissioning Policy*. Infosecurity Magazine. https://www.infosecurity-magazine.com/blogs/best-practices-data-decommissioning/.

Arampatzis, A. (2019, August 9). How to Build a Mature Vulnerability Management Program. https://www.tripwire.com/state-of-security/vulnerability-management/build-mature-vulnerability-management-program/.

* Cavusoglu, H., Cavusoglu, H., & Zhang, J. (2008). Security Patch Management: Share the Burden or Share the Damage? *Management Science*, *54*(4), 657–670. https://doi.org/10.1287/mnsc.1070.0794

* Dey, D., Lahiri, A., & Zhang, G. (2015). Optimal Policies for Security Patch Management. *INFORMS Journal on Computing*, *27*(3), 462–477.

Kent, C. (2020, April 23). *Enterprise Patch Management Best Practices*. IT TRANSFORMATION. https://workflow.servicenow.com/it-transformation/ponemon-vulnerability-response-study/.

* Liu, S., Kuhn, R., & Rossman, H. (2009). Surviving Insecure IT: Effective Patch Management. *IT Professional*, *11*(2), 49–51.

Livne, E. (2019, October 24). *Windows Patch Management: The Best Practices You Need to Start Today*. https://www.ivanti.com/blog/windows-patch-management-best-practices.

MSP, S. W. (2020, February 27). *Patch Management Best Practices*. Solarwinds MSP. https://www.solarwindsmsp.com/blog/patch-management-best-practices.

Rapid7. *What is Patch Management? Benefits & Best Practices*. Rapid7. https://www.rapid7.com/fundamentals/patch-management/.

Voldal, D. *A Practical Methodology for Implementing a Patch management Process*. SANS Institute: Reading Room - Best Practices. https://www.sans.org/reading-room/whitepapers/bestprac/paper/1206.

Vulnerability Metrics. NATIONAL VULNERABILITY DATABASE. https://nvd.nist.gov/vuln-metrics/cvss.