Five Steps to Building a Successful Vulnerability Management Program

Bill Olson, Technical Director, Tenable

April 27, 2017

Table of Contents

I. Why is vulnerability management so difficult?

Vulnerability management poses a unique challenge for businesses. Despite proven technology solutions and the best efforts of multiple IT teams, unresolved vulnerabilities consistently serve as a source of friction and frustration. Regardless of how many vulnerabilities are fixed, there are always those that can't be easily remediated. Infighting between IT teams and business groups is common, each pointing fingers as to who is responsible for the potential risk these vulnerabilities pose to the environment. To deal with their frustration, companies often switch vulnerability management tools every few years, trying to use the latest technology to tighten control. While this may make sense if your security needs have changed, new tools won't help if it is your processes that are broken.

To build a successful vulnerability management program, you need to determine what matters to your organization, and then create a program designed to best address those specific issues. That sounds simple, but it's actually incredibly difficult. No two organizations are the same, so there is no standard playbook to follow that can tell you what to care about. Take the example of a European bank operating in the United States that fails several security audits because they did not patch operating system vulnerabilities. They could lose their ability to hold and handle currency in the U.S., costing them billions in revenue. To that bank, those vulnerabilities are mission critical. Alternatively, for a car manufacturer or grocery store chain, the exact same set of vulnerabilities might require no urgency at all. It's all about understanding how vulnerabilities relate to your business.

To start building a successful vulnerability management program, you need to focus on the right set of vulnerabilities in the right context. Tenable™ has identified five key steps that will help you create a personalized blueprint for addressing your unique vulnerability challenges. Following these steps will align your remediation and response actions with business priorities, help reduce friction and frustration between IT groups and provide more reliable information on the security state of your complete environment.

II. Step 1: Set the right goals

When most organizations start talking about vulnerability management program goals, they immediately state that their goal is "lowering risk." This isn't terribly surprising, since business leaders like CFOs are focused on managing business risk all the time – how much to invest in different areas for a quantifiable return. Unfortunately, security teams have a different vocabulary when it comes to risk. Managing market forces is not the same as managing the interdependency of Oracle, Java and ColdFusion on a web application that is accessed by 30,000 users a minute. There is no sliding "risk scale" of zero to 100% that a security team can track. The idea of "zero risk" in and of itself is a dangerous concept, because it's impossible for any organization to remove all risks.

When companies talk about lowering risk, what they generally want is to tie vulnerabilities to some monetary value. The amount, or risk, would be the potential financial impact that a vulnerability could have on the business. Unfortunately, most businesses simply don't know what is most important to protect in their environment. Which 20 or 30 systems and applications generate 80% of the company's revenue? If anyone has this information, it most likely isn't the security team. But even if they did, to build a comprehensive risk management program, you would not only have to account for those 20 or 30 critical systems, but everything else that touches those systems. This includes all of the applications on those systems, all of the connected systems or devices that access them, all the people who use those systems or devices, all of the buildings where those people and devices reside and so on. The interconnectedness and changing nature of technology systems make vulnerability risk nearly impossible to quantify. Imagine that kind of effort in a huge organization like the military – you would have to factor in every building, every truck, every guard station, every vendor that services the base and more. The inventory alone – being able to identify all these assets that factor into the risk equation – is unmanageable. If your goal is this broad, you can end up failing before you even get started.

Instead of focusing on the very broad goal of risk management, you can break your risk problem down into specific components that are measurable and meaningful. We recommend starting with the areas of attack surface hardening, asset inventory and patch auditing. While not trying to provide an overall picture of risk, these categories help you concentrate on where vulnerabilities can create exposure, then define what it means to successfully mitigate that exposure. How can I make that exposure as small as possible? Do I know what needs protecting? Are my systems up to date? You can define specific metrics to answer each of these questions – which become clear, meaningful goals to track your success.

Potential Vulnerability Management Program Goals

Category	Description	Goal	Example Metric
Attack Surface Hardening	How exposed is my organization?	Make attack surface as small as possible	% exploitable vulnerabilities on internet-facing systems
Asset inventory	Do I know what needs protecting?	Effectiveness at collecting accurate accounting of vulnerabilities – including for systems that require credentials	% of systems discovered vs scanned in last 30 days
Patch auditing	Are my systems up to date?	Effectiveness of patch process for security, feature/functionality and warranty needs	% of systems patched in last 30 days

In order to be successful, you not only need to set the right goals and metrics, you also have to take into account the staff you have to work toward those metrics. If you ask a CISO which of the above areas– attack surface hardening, asset inventory or patch auditing – that they want to focus on, they are going to undoubtedly answer “all three!” But trying to do everything – and doing it poorly – is another reason why vulnerability management programs fail. Without dedicated personnel assigned to track and improve each of these areas, some are likely to fall behind. It is a more effective strategy to decide which one or two priorities to focus on for a 6 to 12-month period. At the end of that period, you can adjust or expand goals based on results and changing business requirements.

III. Step 2: Make sure vulnerability data is accurate, actionable and timely

Bad data – false positives and false negatives – are one of the key stumbling blocks for any vulnerability management program. It’s also a source of frustration between security and operations teams, as well as between IT and the business. One bad number or one bad report and your credibility is damaged. Assessment tools not only need to be accurate, they need to be able to reach the entire environment – even systems that require credentials to scan them. Anything less and you’re providing only a limited view of your true vulnerability state.

Data also needs to have the appropriate context so that people can take action in an efficient manner. What does the data mean to the organization? What does it mean to the people who are consuming it to do their jobs? An abundance of data does not help you meet your goals if no one can actually use it. Take, for example, a technology manufacturing company that was troubled by their lack of progress in remediating vulnerabilities. Their team was tasked with resolving all critical and high severity vulnerabilities within 90 days of discovery, but were consistently missing this goal. When Tenable investigated how the vulnerability data was being shared with the team, we discovered that it came in the form of a 175,000-page report that included all severities of vulnerabilities – critical, high, low and informational. When questioned, the manager indicated that they were given this giant report because even though they are only responsible for critical and high severities, ideally they want them to fix everything if there was extra time. Digging through 175,000 pages is incredibly inefficient, so it was not surprising that the team was failing. After a recommendation to provide two separate reports – one for critical and high vulnerabilities and one for everything else – the team was much better equipped to meet its remediation goals.

In addition to being accurate and relevant, data also needs to be timely. Scanning frequently and reporting regularly is critical, because out-of-date data can be just as damaging as inaccurate data. However, you also need to make sure your reporting is aligned with your patch remediation cycle so that reporting and updates are relevant. If not in sync, you end up giving accountable owners the same information over and over again – forcing them to manually determine what they have to address and what they should ignore. This is not only inefficient, it can lead to negative behavior.

Take, for example, a regional bank who wanted to create highly accurate reports for their vulnerability management program so that they could track compliance with financial industry regulations. They created an elaborate system of collecting data, weeding out false positives and manually validating to ensure everything marked as fixed was actually remediated. The process was incredibly accurate, but it unfortunately took 6 months to complete. They would not receive the January report until June – at which point none of the information in the report is actionable, because the environment has completely changed. These reports gave the impression to both business groups and the operations team that the scanning and vulnerability management team delivers bad information, creating distrust and tension that wouldn't exist if they had focused on producing more timely reports.

IV. Step 3: Account for patching gaps

Unpatched systems account for a significant portion of vulnerabilities in an environment and become one of the key areas of friction between security and operations teams. There are generally three reasons that a system doesn't get patched:

1. The asset, OS or application owner can't be identified.

2. The system can't be patched because of uptime requirements, the potential something will break or because the system is out of support.

3. Something is broken with the update process or the technology itself.

You need to be able to identify if any of the above are the culprit for any unpatched systems so that there is a clear understanding of who is responsible – and whether it is already a known issue. Tenable met with a large healthcare company that would hold a weekly call to go through all unknown IP addresses that showed up in weekly reports one-by-one to determine who the owner was. This is incredibly time consuming, but extremely valuable to the business. If you leave assets unidentified, you are just adding to your bad data problem – it will continue to show up on the report week after week and your credibility and effectiveness will be questioned. Creating a regular process that ensures ownership of all unpatched systems is a best practice all organizations should follow. Even if you don't hold a weekly meeting to assign owners, you should ensure that unidentified assets are not included in reports or counted as part of team metrics, since any vulnerabilities on these systems are outside of the team's control.

If there is a known issue that prevents a system from being patched, like the patch would break a business process or application, then that information does need to be reflected accurately in reports. If that vulnerability – which you know you can't fix – shows up week after week as not fixed, it makes the owner responsible for patching look like they are not doing their job. This creates unnecessary distrust and friction between groups and individuals. If you are able to mark that system as a known exception, then it removes that source of conflict. The security team's success statistics are not impacted, and everyone is clear that there is a sound business reason for accepting the risk of that unpatched system.

Finally, sometimes systems fall out of management, their agents stop communicating or other technical challenges arise that cause patching to fail. Security teams need a solution to identify when those breaks happen, and quickly formulate a solution. Without visibility into the process, security teams can't understand why their numbers are off or are not improving. This ultimately undermines the credibility of the security team, where trust is their most important currency. Quickly identifying the sources of patching issues – technology, people or process – removes the problem and allows security to maintain a reliable process.

Quickly identifying the sources of patching issues – technology, people or process – and tracking them as exceptions allows security to maintain a reliable process and build trust. If all vulnerabilities for these unpatched systems appear in the same report, it presents a flawed view – particularly since most of these are issues security teams are powerless to address. Creating separate tracking mechanisms for each of these patching gaps will help drive resolution, increase transparency and improve confidence in the data.

V. Step 4: Address interdependencies and conflicts between people and processes

We have already discussed how bad data and unpatched systems can create tension between security and operations teams, or security and the business. It's very important to address these issues as part of any vulnerability management program. By focusing on the interdependencies between people and processes you can de-escalate conflict, build trust and improve effectiveness within and between teams.

Vulnerability reports are particular lightning rods for upholding or undermining personal accountability. Because the people on the receiving end of that data are directly affected, it should be a priority to reflect their efforts accurately. That way reports cannot be used to sabotage or undermine credibility of individual team members. Fortunately, technology makes it very easy to address these concerns. An excellent example where Tenable has seen this in action is at a large online commerce site who had just hired a new manager for their vulnerability management program. Before she started, the company would run a weekly vulnerability scan, then meet as a team for hours to walk through the giant report. The new manager decided that it could make the meetings much shorter and everyone more effective if they ran individual reports for each person that only showed the vulnerabilities that were relevant to their systems. This took very little effort, but everyone on the team had a very positive response. They received the information they needed, got several hours of their work week back and no one was unclear on who was responsible for which vulnerabilities. Contextual, timely, focused reports can build goodwill and reduce conflict if used correctly.

Another source of conflict is different groups working at odds because they have competing goals. This most commonly shows up between operations and security teams. One group needs to ensure uptime and availability, so they don't want to bring their servers down to be patched. The other group needs the servers to be patched to meet organizational and industry compliance/security requirements. An extreme example of this is a major payment processing company that had accumulated 1.2 million vulnerabilities across 40,000 systems – statistics that made their security team look terrible. The issue was that the business was committed to providing its payment processing partners with 99.9999% uptime. Systems were only allowed to be down for updates once every 4 months. Vulnerability reports were unhelpful, because the security team was never allowed to remediate anything. Resolving the issue between the operations and security teams had nothing to do with tools or better reporting – it required a business solution. They went back to partners and renegotiated more granular uptime guarantees that allowed them to take systems offline for patching on a weekly basis. They didn't receive backlash, as partners were happy that systems were more secure for a few less nines of uptime. Addressing interdependencies between security and other organizational requirements helps reduce friction and improve overall service to partners and customers, while still allowing you to meet goals.

Finally, understanding how processes affect individuals and teams within your IT organization is also critical to creating a successful vulnerability management program. A large manufacturing company decided they wanted to measure the effectiveness of their patching on a monthly basis, with the goal of patching 95% of systems within 30 days of a vulnerability being detected. The problem was, they were running reports on a weekly basis that included vulnerabilities that were detected in the last 7 days. The operations team looked at the report and viewed the security team as failing, because they weren't anywhere close to the 95% mark. The process was broken – reporting and patching schedules were out of synch. The operations team should have been only running reports for vulnerabilities that were unresolved for 31 days or more – since that is the window the security team was given to fix them. Making sure your process fits your vulnerability goals will determine how successful you are in rolling out your program.

VI. Step 5: Make sure you are measuring the right things

All the steps discussed to this point will help you put together a successful vulnerability management program. But to prove you're actually succeeding, you need a way to measure your success. Measuring the wrong things can be just as detrimental as bad data or having the wrong goal.

Many businesses think that if they measure how many systems are being patched or count vulnerabilities, they will have a clear picture of system compliance. But tracking remediation trends simply doesn't work for vulnerability management. There is always going to be a healthy ebb and flow of new vulnerabilities to address, new systems and applications being added and systems being retired. There is never a project end, never a point at which you can claim victory. This week you may take care of 1,000 vulnerabilities and not meet a goal of 80% of vulnerabilities resolved. Next week you may resolve 600 and exceed that goal.

Take this example report from a customer that wanted a monthly vulnerability scorecard. Part of their plan was to assign a weighted score to each vulnerability type and the asset's criticality. For example, Level 1 assets with a critical vulnerability get a score of 100, Level 2 assets with a critical vulnerability get a score of 65 and Level 3 assets with a critical vulnerability get a score of 25. The problem with this approach is that if one new vulnerability is detected and remains open each month, your report starts to look like this:

Unresolved Critical Vulnerabilities Report

Asset Category	January	February	March	April	May
Level 1	100	100	100	100	100
Level 2	65	65	65	65	65
Level 3	25	25	25	25	25

This report makes it look like the team has done no work from one month to the next. But that is completely misleading – they could actually have been remediating hundreds of vulnerabilities each month at each level.

Instead of trend lines, measurement for a good vulnerability management program focuses on the exceptions. If my patching process is set for 45 days, show what hasn't been patched on day 46. Then you can start to see areas of weakness. Why is Sally's group always behind Fred's group? Do I need to get rid of Sally? Do I need more Sallys because everybody is too busy cutting down trees that they can't sharpen their axes? Do I need to redefine my policy? Maybe 45 days is unrealistic and you need it to be 50 days. Maybe it's too easy and you need to make it 30 days. Regardless of what the specifics are, well defined measurement can help you adjust your vulnerability management program and put it on the track to success.

VII. Conclusion

Vulnerability management is not a project, because projects have a beginning, a middle and an end. Vulnerability management is something that will never end. You will never be done with it; it will never become unnecessary. Knowing this, the smart choice is to put together a program where everyone is an active, informed participant working towards common goals. If not, you end up with divisive, inefficient processes that have everyone pulling in different directions.

Using the five steps outlined above, organizations can build a solid, successful vulnerability management program that improves visibility into organizational risk and fosters better cooperation between security, operations and business management teams. You will be able to ensure alignment between business goals and your vulnerability management data collection, reporting and measurement processes. This makes your program more accurate and effective, and also fosters better relationships between your security team and other groups.

Tenable has a comprehensive vulnerability management solution to help you find and fix vulnerabilities faster. Using built-in dashboards, extensive report templates and automated integration with complementary third-party systems, our solutions gives you the accuracy, context and measurement you need to make your program a success by:

- Consolidating and validating vulnerability data across your organization.
- Prioritizing which vulnerabilities are most important.
- Using highly customizable dashboards, reports and Assurance Report Cards® (ARCs) to help you visualize, measure and analyze the effectiveness of your vulnerability management program.
- Providing multi-layered scanning for your complete environment, even on systems that require credentials, to ensure you have accurate, reliable data.

For more information on Tenable vulnerability management solutions, please visit tenable.com/products.

VIII. About Tenable Network Security

Tenable Network Security transforms security technology for the business needs of tomorrow through comprehensive solutions that provide continuous visibility and critical context, enabling decisive actions to protect your organization. Tenable eliminates blind spots, prioritizes threats, and reduces exposure and loss. With more than one million users and more than 20,000 enterprise customers worldwide, organizations trust Tenable for proven security innovation. Tenable's customers range from Fortune Global 500 companies, to the global public sector, to mid-sized enterprises in all sectors, including finance, government, healthcare, higher education, retail and energy. Transform security with Tenable, the creators of Nessus and leaders in continuous monitoring, by visiting tenable.com.