VIRTUALIZATION
CLOUD
APPLICATION DEVELOPMENT
HEALTH IT
NETWORKING
STORAGE ARCHITECTURE
DATA CENTER MANAGEMENT
BI/APPLICATIONS
DISASTER RECOVERY/COMPLIANCE
SECURITY

TechGuide

Vulnerability Management Programs: A Handbook for Security Pros

To ensure that your vulnerability management program is effective and aligned with your broader risk-management goals, you must identify and prioritize vulnerabilities based on sound risk-management principles.

TechTarget

Sealing the Security Perimeter With a Vulnerability Management Program

With each new year, information security gets tougher—threats keep growing in both frequency and sophistication while corporate security budgets never seem to keep pace. So while the why of a corporate vulnerability management program is obvious; what's less clear is the how. This handbook tackles that dilemma by providing specific techniques and tools InfoSec pros can use to improve their programs.

A company that mounts a vulnerability management program often soon finds itself with an avalanche of information on network security vulnerabilities. A crucial next step, then, is to wade in and sort out the data, to identify what vulnerabilities get priority. To aid in that prioritization process, security expert Mike Chapple proposes a three-prong approach, including the calculation of "risk scores."

Once you've got your vulnerability priorities straight, you'll need a set of best practices to keep things running smoothly. Today that means being aware that security attacks are no longer unidirectional but might assault your system via multiple channels. Diana Kelley offers keys to a dynamic vulnerability management program, which include resource-maximizing tips such as how to reduce the "noise" of false positive threat alerts.

This handbook closes with CTO Dave Shackleford's innovative take on penetration testing, using the latest social engineering concepts. His four techniques—phishing, pretexting, media dropping and tailgating—take pen testing to new levels.

The threats to information security are not going away. But this handbook gives you the guidance you need to meet the challenge head-on and seal up your enterprise's security perimeter. ■

Brenda L. Horrigan
Security Media Group

How to Rank Security Vulnerabilities in Your System

Soon after initiating a vulnerability management program, enterprises often find themselves facing an intimidating avalanche of data about network security vulnerabilities. Scan results may show hundreds or even thousands of vulnerabilities distributed across a wide variety of systems and applications.

How should security professionals tackle this mountain of risk? In this chapter, we examine a three-prong prioritization program that incorporates external criticality assessments, data sensitivity and the existing control environment to help organizations successfully rank vulnerabilities and, in turn, prioritize remediation efforts.

This three-step process assumes that you have access to information about the network security vulnerabilities that exist in your environment, the sensitivity of information processed by systems and applications, and the state of existing security controls in the environment. These may come from a variety of sources within your vulnerability management program, including Web and network vulnerability scanners, data loss prevention systems and configuration management software.

STEP 1: DETERMINE VULNERABILITY SEVERITY

The first data element you need is an assessment of the severity of each vulnerability that exists in your environment. In many cases, this severity information is provided through data feeds from the vendors that provide your vulnerability management tools.

The severity assessment should be based upon the potential damage that a successful exploit might cause. For example, a vulnerability that allows an attacker to gain administrative access to a system is much more severe than one that causes a denial of service.

Severity information may also take into account the real-world existence of exploits; a theoretical vulnerability with no known exploits is less severe than one used by a virulent piece of malware.

For the purposes of our model, we will assume that you are using a product that uses a five-point vulnerability rating system, with vulnerabilities that have the highest risk of a damaging exploit receiving a 5 rating.

STEP 2: IDENTIFY DATA SENSITIVITY

The risk a vulnerability poses is magnified by the sensitivity of the information processed on systems containing that vulnerability. For example, systems containing Social Security numbers or credit card data should generally be handled with much more care and concern than systems containing only publicly available information.

This does not mean that only systems containing sensitive information should be well-managed; a compromise of your public-facing website could cause just as much reputational damage to the organization as a disclosure of sensitive information. However, the presence of sensitive information certainly magnifies the impact of a successful attack.

Gathering information on data sensitivity can be tricky, depending on the maturity of your organization's information-classification program. If you're just getting started, you may wish to use a fairly simple model that divides data into three levels:

- **Highly sensitive information** is either heavily regulated or would be extremely damaging to the organization if inadvertently released. This "crown jewel" of our information security programs contains data elements such as credit card numbers, protected health information and bank account details.

- **Internal information** is every piece of information that does not fit the "highly sensitive" definition but should not be publicly released. This category may seem overly broad; it is also the hardest to define. If you don't have a data classification program, lumping all this data into a single category is the most expedient way to get started. If business

needs dictate, consider subdividing this category at a later date.

- **Public information** is anything that your organization is willing to disclose to the general public, such as product literature, data shared on your public website and released financial statements.

When it comes time to assign data sensitivity ratings to systems, base your evaluation on the highest level of information stored or processed by a system. Systems processing highly sensitive information are assigned a data sensitivity rating of 5, while those processing internal information receive a 3 rating. All other systems are rated 1 on data sensitivity.

STEP 3: EVALUATE EXISTING CONTROLS

The final step of the process is to evaluate the existing controls that protect potentially vulnerable systems from compromise. The method you use to assign these ratings will vary depending upon the particular controls your organization requires. For example, if you have a highly secured network used for extremely sensitive systems, you might assign these systems a 5 rating on a five-point control scale. Similarly, a system with a public IP address that is accessible from the Internet hosting a Web application but not protected by a Web application firewall might be assigned a 1 or 2 rating. Choose a rating scale that accurately reflects the expected controls in your environment, and assign higher ratings to systems that have strong security controls.

PULLING IT ALL TOGETHER

Once you've gathered all of this information, you may use it to assess the vulnerabilities that show up on your reports. When you have it all consolidated in one place, perform this simple calculation for each vulnerability that exists on a system:

$$\textit{Risk Score} = \frac{\textit{Vulnerability Severity} \times \textit{Data Sensitivity}}{\textit{Existing Controls}}$$

If you chose five-point scales for each measure, this will result in a vulnerability rating

ranging from a minimum of 0.2 (for a low severity vulnerability in a well-controlled system containing only public information) to a maximum of 25 (for a high severity vulnerability in a system lacking security controls containing highly sensitive information).

An effective vulnerability management program based on risk-based prioritization decisions is a must for any organization looking to reduce IT security risk.

While this may seem like a lot of data to gather and math to perform, you can find ways to automate the process and feed your vulnerability prioritization efforts. For example, you might create a database that contains data sensitivity and control status information for all of your server assets. Similarly, scripts can parse vendor reports to automatically extract vulnerability severity information, pull relevant information from the database and calculate the risk score.

There are many ways to customize a vulnerability prioritization system for a particular organization. Regardless of the tweaks you make, an effective vulnerability management program based on risk-based prioritization decisions is a must for any organization looking to reduce IT security risk. Simplifying the process used to perform vulnerability risk analysis makes it much easier to begin and sustain such a program. —*Mike Chapple*

Five Tips to Improve a Threat and Vulnerability Management Program

Modern enterprise cybersecurity teams must be prepared to deal with a barrage of new and rapidly evolving threats. From script kiddies to sophisticated hackers working for criminal organizations, if an enterprise doesn't have plans in place to deal with such threats, it will pay the price in expensive, embarrassing data breaches.

An effective threat management program is undoubtedly a vital ingredient for any enterprise security team dealing with the modern threat landscape.

However, keeping such a program running smoothly takes time and ongoing planning. Resources must be allocated to put a program in place that can deal with a multitude of attacks.

In this tip, I offer five best practices that companies can implement to increase the effectiveness of their threat and vulnerability management programs.

1. MANAGE ALERTS

If a tree falls in the woods and no one is there to hear it, does it make a sound? This old philosophical question comes to mind when thinking about threat management. Like that tree, are alerts about suspicious activity and anomalous behavior that can signal an attack in progress that an administrator doesn't see or review really alerts? The most important thing a company can do to get a handle on threat management is to ensure that someone is there to review and respond to an alert that's been triggered. To meet this requirement, most organizations should assign a dedicated resource, or resources, with the remit to review log and alert consoles on a daily basis.

At the daily alert review level, it's not uncommon to see organizations assign different specialists to review different alert consoles. For example, a firewall operations expert may be in charge with reviewing firewall rule

changes and alert logs, while an applications engineer may be responsible for reviewing the logs and alerts from the Web application firewalls and Web app scanners.

2. TAKE A HOLISTIC VIEW

In the realm of detection evasion, attackers are growing increasingly sophisticated, as can be seen with their use of multichannel attacks and other techniques that are designed to fly below security radars. An example of a multichannel attack is the seemingly innocuous spear SMSish (SMS phish to a smartphone), which fools the user into clicking on a link that leads to a rogue site that has been designed to look legitimate. The user may then be tricked into entering sensitive data or clicking on a link that infects the targeted machine with a bot. Once the user's sensitive information has been collected, the attacker attempts to log in to a system and dig deeper into the corporate network for more valuable information.

Most organizations already monitor for threats and suspicious activity on most devices, including wired desktops, wireless tablets, smart devices, laptops, Web applications, databases and servers. To catch multichannel attackers, organizations should corral alerts from all of those systems into a single console where correlation rules can filter the seemingly innocuous activity that, when combined, creates a single, organized attack.

3. REDUCE FALSE POSITIVES

Excessive alerts and false positives ratchet up the "noise" ratio so high that it can be difficult (if not impossible) to sift through all the available data to find the truly malicious events. If an organization's administrators can't discern important alert signals through all the insignificant events, the alert system becomes useless. To reduce the number of false positives produced, an enterprise should first analyze the alert output of its threat-warning console, or consoles, and determine if the rules can be tuned to reduce the false positive noise, or filter alerts by level of confidence so that admins can see which ones are more likely to be relevant.

One way to lower those levels without losing

critical alerts is to set threshold levels that match normal activity on the network. For example, a company that forces all users to change passwords on the same 90-day cycle might find that failed logins increase significantly on the day after the end of a cycle. To account for this occurrence, a rule that normally

A company that forces all users to change passwords on the same 90-day cycle might find that failed logins increase significantly on the day after the end of a cycle.

signals an alert after three failed logins could be increased to five failed logins on days following the password change. The logins could also be linked to other threat indicators, such as attempts to log in using the same ID from different IP addresses, to increase accuracy.

Keep in mind that overtuning or setting thresholds too low will result in false negatives, so test thresholds carefully before implementation.

4. INTEGRATE WITH THE SOC

As mentioned earlier, aggregating threat information into a single console gives organizations threat visibility across the whole enterprise. To gain even deeper visibility, a company can integrate that single or multiconsole view with its security operations center (SOC). At most companies, the SOC's main purpose is to monitor security activity and respond to attacks quickly, which makes integrating the threat management program with the SOC something of a no-brainer.

To integrate threat information with the SOC, filter alert information into a SIEM system and log data into either the SIEM or whatever is being used for log centralization. Next, create rules in the SIEM, log aggregation tool or both to parse through alert information and flag legitimate attack activity for further investigation or response. To integrate effectively, make sure that engineers and administrators in the SOC have access to the standard operating procedures for incident response, so that the team knows the correct escalation paths, communication protocols and approved response activities.

5. VALIDATE REMEDIATION ACTIVITIES

In the heated atmosphere of an incident response, organizations can easily overlook validating the remediation activities. Even during routine activities like patch management, many companies fail to close the remediation loop with validation. Did the patch get loaded properly? Did it close the intended vulnerability? Without testing, an organization can't be certain that the remediation was successful and the threat exposure was closed.

Complete the threat management cycle with steps for validation. These can include rescanning systems to validate patches and performing application and network penetration testing to confirm that fixes or controls are blocking vulnerabilities as expected.

CONCLUSION

The modern threat landscape is complex and attacks come in from multiple channels and sources. Organizations need to have a multichannel approach to managing and responding to threat activity. Rolling up activity data into the SIEM, or other management console, and having trained professionals review the alert data will increase situational awareness and improve response time and efficacy. And when patches or controls are in place for remediation, don't forget to validate that they are installed and working. Stopping all attack activity is impossible, but by taking steps to improve a threat and vulnerability management program, businesses can avoid an incident becoming catastrophic. —*Diana Kelley*

Social Engineering Penetration Testing: Four Effective Techniques

SOCIAL ENGINEERING HAS become one of the more prevalent attack methods in use today, and has been featured heavily in some high-profile breaches. The 2011 RSA breach, for example, involved a targeted spear phishing campaign and an exploit-laden Excel file. Thus, for organizations to adequately model the real threats they face, social engineering penetration testing should be a mandatory tactic in every pen testing toolkit.

Social engineering relies heavily on psychology. There are several types of incentives and motivators to which people are highly susceptible, allowing social engineers to persuade people to take an action. For example, Dr. Robert Cialdini in his classic book *Influence: The Psychology of Persuasion* (first published in 1984) described six key motivators:

- **Reciprocation:** Feeling indebted to someone for doing something for you.
- **Social proof:** Looking to others for guidance on how to act.
- **Commitment/Consistency:** Developing patterns of behavior and maintaining them out of habit.
- **Liking:** Wanting to "fit in" and being more easily persuaded by someone you like.
- **Authority:** Acquiescing to requests or demands from perceived authority figures.
- **Scarcity:** Feeling higher motivation to pursue something if it is limited or exclusive.

Pen testers can leverage these motivators when performing social engineering assessments.

There are four social engineering techniques that pen testers can use to test an

organization's security: phishing, pretexting, media dropping and tailgating.

PHISHING

Phishing involves sending an email to a user to persuade the user to perform an action. The goal of most phishing emails in a pen testing project is simply to entice the user to click something and then record that activity, or to actually install a program as part of a larger penetration testing effort. In the latter case, exploits can be tailored to client-side software known to have problems, such as browsers and dynamic content/media plug-ins and software.

The key to a successful phishing campaign is personalization. Tailoring the email to the targeted user, such as by sending it from a trusted (or perceived-to-be-trusted) source, makes it more likely the user will read the email or follow some direction in it. A good pen tester always remembers to check spelling and grammar; a well-written email, even a short one, is much more believable.

Probably the best-known tool for creating phishing attacks is the open source Social Engineering Toolkit (SET). With its menu-driven email and attack-creation system, it's one of the simplest ways to get started with phishing. Commercial tools like PhishMe Inc.'s PhishMe and Wombat Security's PhishGuru can also be useful.

PRETEXTING

Pretexting involves telephoning the target and trying to solicit information from him or her, usually by pretending to be someone who needs assistance. This technique can work well in a penetration testing project by targeting non-technical users who can provide useful information.

Pen testers can use phishing, pretexting, media dropping and tailgating to test an organization's security vulnerabilities.

The best strategy is to start with small requests and drop names of real people in the organization who may be waiting for something. In the pretexting conversation, the pen tester explains they need the target's help. (Most people are willing to do small tasks that aren't perceived as suspicious requests.) Once rapport has been established, the pen tester can ask for something more substantial with more success.

Reconnaissance before the pretexting exercise, using Google and tools like Paterva's Maltego, can provide needed background information. Phone-masking and proxying tools like SpoofCard (a subsidiary of TelTech Systems) and SpoofApp from SpoofApp.com LLC, as well as Asterisk PBX add-ons from Digium Inc., can disguise the pen tester's phone number, even making it appear to come from the organization's own number block.

MEDIA DROPPING

Media drops usually involve a USB flash drive left somewhere conspicuous, like a parking lot or building entrance area. The social engineer places an interesting-sounding file on the flash drive that launches some sort of client-side attack when opened.

One free tool for creating these files is Metasploit, with its built-in malicious payload generators. The "Infectious Media Generator" option in SET also uses Metasploit, but helps automate the process. SET can create a "legitimate" executable that runs automatically when Autorun is enabled on a target's PC. Using automatic execution techniques and interesting-sounding files together can increase the chances of success.

A more sophisticated approach to performing a media drop as part of a pen testing project is to develop custom attacks and programs on a USB drive, or to purchase USB drives that are

Phone-masking and proxying tools can disguise a pen tester's phone number, making it appear to come from the organization's own number block.

pre-built for this purpose. To increase the success of USB attacks, add both automated exploits and attack-laden files to the device (PDF, Word and Excel formats are best). Labeling the device with an interesting sticker, like "HR Data" or "Employment," can help, too.

TAILGATING

Tailgating involves getting into a physical facility by coercing or fooling staff there, or just walking in. Usually the focus of these tests is to demonstrate that the pen tester can bypass physical security.

Pen testers should plan to procure sensitive data or install a device quickly to prove they were successful, as they may have only a short window of time before needing to leave the facility. The pen tester can take pictures of exposed documents left on printers or desks, or install a pen testing drop box device to provide Wi-Fi or 3G network access back to the environment later.

> ***A pen tester can uncover vulnerabilities and then recommend security controls and education techniques that will reduce the odds of malicious attacks.***

By using these four social engineering techniques, the pen tester can uncover an organization's vulnerabilities and then recommend security controls and education techniques that will reduce the odds of an organization falling prey to malicious social engineering attacks. —*Dave Shackleford*

MIKE CHAPPLE, *Ph.D., CISA, CISSP, is an IT security manager with the University of Notre Dame. He previously served as an information security researcher with the National Security Agency and the U.S. Air Force. Chapple is a frequent contributor to SearchSecurity and serves as its resident expert on enterprise compliance, frameworks and standards for its Ask the Experts panel. He previously served as site expert on network security, is a technical editor for* Information Security *magazine and the author of several information security titles, including the* CISSP Prep Guide *and* Information Security Illuminated.

DIANA KELLEY *is a partner with Amherst, N.H.-based consulting firm SecurityCurve. She formerly served as vice president and service director with research firm Burton Group. She has extensive experience creating secure network architectures and business solutions for large corporations and delivering strategic, competitive knowledge to security software vendors.*

DAVE SHACKLEFORD *is principal consultant at Voodoo Security, senior vice president of research and CTO at IANS and a SANS analyst, instructor and course author. He previously worked as CSO for Configuresoft, CTO for the Center for Internet Security and as a security architect, analyst and manager for several Fortune 500 companies. He is co-author of a SANS Institute book on virtual security and currently serves on the board of directors at the SANS Technology Institute.*

TechTarget

Vulnerability Management Programs: A Handbook for Security Pros is a SearchSecurity.com e-publication.

Robert Richardson | *Editorial Director*

Eric Parizo | *Executive Editor*

Kathleen Richards | *Features Editor*

Kara Gattine | *Senior Managing Editor*

Brenda L. Horrigan | *Associate Managing Editor*

Brandan Blevins | *Associate Editor*

Sharon Shea | *Assistant Editor*

Linda Koury | *Director of Online Design*

Neva Maniscalco | *Graphic Designer*

Doug Olender | *Vice President/Group Publisher*
dolender@techtarget.com

TechTarget
275 Grove Street, Newton, MA 02466
www.techtarget.com