How to Achieve a Saner, More Effective Vulnerability Management

David Kennedy, IANS Faculty

DAVE KENNEDY

OSCE, OSCP, CISSP, ISO 27001, GSEC, MCSE

Dave Kennedy is a computer hacker and the founder of TrustedSec and Binary Defense. Dave has helped with the Mr. Robot TV show as well as being directly mentioned on the show and others. Dave also served on the board of directors for the (ISC)² organization, is the co-author of the best-selling book "Metasploit: The Penetration Testers Guide", the creator of multiple widely downloaded open-source tools, frequent keynote speaker across the world and on the news, and an avid gamer.

Symposium Objectives

- Introduce one another and share and collaborate.
- Experiences help identify new ways to improve a threat and vulnerability management program.
- Discussion – not purely slide driven.
- If we don't collaborate, this will be a quick session!
- Be open, ask for help, share experiences – it's how we all get better!

- Vulnerability Management Discussion
- Process and Discussion
- Tools and Automation Workflows
- Baselines and Hardening
- Legacy: Patch, EOL, or Segmentation?
- Improve Discovery and Program
- Conclusion and Comments

IANS

A TVM Program

- TVM is one of the hardest programs around information security and one of the most important.
- Difficult due to size of organizations, technology adoption, and how fast this industry works.
- Most programs fail due to cycle of find + patch.
- Strategy plays equally or more important role than just fixing issues that are identified.

Addressing Issues, or not?

[redacted] 9:33 AM
I forget what customer it was but I did an internal early on at [redacted] Got access to a share with responder creds. Found someone old [redacted] report. Checked out the content determined they'd remediated - nothing - and pretty much recreated all the findings :-); plus more..

[redacted] 9:35 AM
i found the report i did the year prior for a client, in the clients IT share...they hadnt remediated anything either...

Patch Statistics

- ”Nearly 60% of organizations that suffered a data breach in the past two years cite as the culprit a known vulnerability for which they had not yet patched.” - DarkReading

- https://www.darkreading.com/vulnerabilities---threats/unpatched-vulnerabilities-the-source-of-most-data-breaches/d/d-id/1331465

- https://www.spamtitan.com/blog/poor-patch-management-policies-to-blame-for-equifax-data-breach/

IANS

Vulnerability Management Introduction

Building from the ground up.

Threat and Vulnerability Management Program

- Threat and Vulnerability management (TVM) is designed to identify, analyze, and remediate exposures within your network.
- Usually prioritized based on risk (larger discussion), and based on finite resources and time.
- Metrics often times difficult as an “increase” in vulnerabilities doesn’t necessarily show program is under hardship.

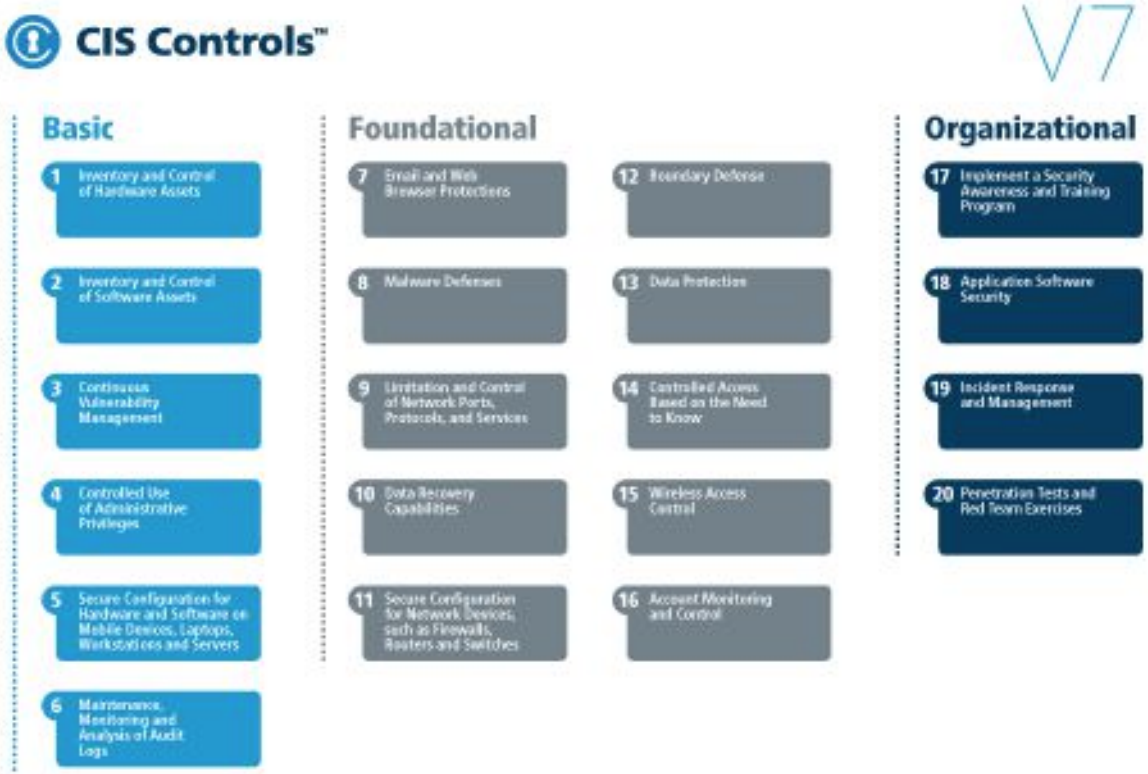
Ultimate Goal: VM

- To identify a program and process that allows the continual identification of exposures, address risk, and reduce them over time.
- Continue to identify new systems with a process being added to the network as well as identify network segments that you may not have visibility into.
- Prioritize based on risk and allocation of resources that address the issues identified.

VM Challenges

- Most vulnerability management programs fail due to not fixing root cause issues (hardening guidelines, patch cycles, CMBD).
- Third party applications tend to be the worst, and ownership of systems is also a program (owning OS vs application).
- “Where to start” when you have a substantial number of exposures.
- Little formalization of process, or process is flawed and difficult to change.
- Most organizations don’t have a formal asset inventory / CMDB.

20 Critical Security Controls



IANS

Critical Control 1

- Inventory of authorized and unauthorized devices
- Quick wins:
  - Develop asset inventory program
  - Deploy DHCP logging
  - Ensure automated “adds” of new equipment to the inventory
- Correlate with DHCP leases, known inventory, DNS entries
- Do you have a CMDB or inventory database?
  - If so, tie vulnerability reporting to it

System Inventory Visibility

- System inventory data that tends to be most useful:
  - MAC addresses
  - IP addresses/subnets
  - System DNS names
  - NetBIOS names
  - OS in use and version
  - Application headers
- Add administrative info
  - Purpose of each system
  - An asset owner responsible for each device
  - The department associated with each device

IANS

Hygiene: Critical Control 2

- Inventory of authorized and unauthorized software
- Knowing what services are running is critical for vulnerability management
- Quick wins:
  - Application whitelisting
  - Authorized software listing
  - Regular scanning and alerting
- Data important for services/processes and applications:
  - Name
  - Version and known vulnerabilities
  - Privileges in use (if available)
  - Credentials in use (when possible)
  - Banner / HTTP header
  - Ports in use
  - Encryption in use
  - Error messages/conditions

Metrics for System Inventory

- System counts:
  - Number of total systems found
  - Number of authorized systems
  - Number of unauthorized systems
  - % change in authorized vs. unauthorized
- Identifiers
  - OS and platform types/variants
  - IP ranges and network attributes
- Time on the network (over a period)

Metrics for Software Inventory

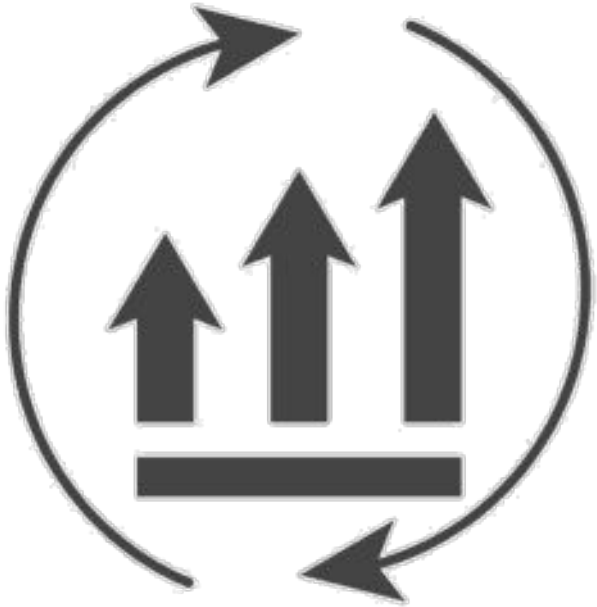
- Number of services available (external)
- Number of processes/services available (local)
- Number of unauthorized apps installed on standard builds
- Unauthorized app quantities over time
- Number of incidents involving systems with unauthorized apps vs. those without

Scope

- What is the scope of the vulnerability management program?
- What does that scope cover?
- The scope can be difficult to determine based on size and complexity of organizations.
- Multiple asset owners, legacy systems, all create chaos when it comes to a TVM program.

Blame the Scanner

- Most organizations blame the scanner as the main fault for the program.
- In reality, most scanners are comparable and little differences between them.
- It’s what best fits the process, the features are less of an extent and the scanner usually isn’t the problem.



Resource Allocation

- Vulnerability management requires resources in order to effectively analyze and understand large data-sets.
- Establish roles and responsibilities (often built into job descriptions) for performance metrics and priority.
- Needs to involve an asset management system of sorts to identify owners of systems.
- Integration into automation frameworks (discussed later).

Metrics Discussion

- Scanning Coverage and Visibility
- Number of high risk exposures in environment and broken out by asset.
- Risk associated with asset and based on priority of systems (external vs internal vs data).
- Average time to resolve high-risk exposure.
- Overall remediation of high-risk exposures percentage.
- More?

Program Process and Discussion

What works? What doesn't?

Process Discussion

- The “program” which is built on an integrated process is key.
- What’s the most effective process you’ve seen?
- How can the process be improved?
- What are some areas where you struggle in your process?

TVM Frameworks

- DREAD from Microsoft often used:
  - https://en.wikipedia.org/wiki/DREAD_(risk_assessment_model)
- NIST 800-40 and RMF
  - https://csrc.nist.gov/publications/detail/sp/800-40/version-20/archive/2005-11-16
- Factor Analysis of Information Risk (FAIR)
  - https://www.fairinstitute.org/
- Octave
  - https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8419

Case Study: VM

- Manufacturing Company Fortune 500: VM program in disarray, had several million critical and high exposures.
- VM was run by the security team but little to no adoption through the organization and especially IT.
- Asset fragmentation causing difficultly in identifying owners of systems.
- Find vulnerability, patch, vulnerability, patch in a cycle.

IANS

Case Study: Cont

- Process revamp, VM owned by security – patch management metrics owned by IT.
- Senior leadership meeting would hold each region accountable on reduction and go over any hurdles.
- Rewrite of job descriptions to include VM as a process.
- First step was to develop patch processes that were more frequent and included third party applications and security baselines.
- Second step was visibility and asset ownership of those systems.
- Last step was reporting metrics and monthly touch points.

Case Study: Cont – a year later

- Process continues to expand and mature to include new systems discovered (especially on shop floor locations).
- Exception process of Update, Legacy (phase out), Patch, or Remove.
- Substantial reduction of critical and highs over time and continue to focus on remediation cycles with team.
- New systems are automatically added to asset management system and incorporated into vulnerability management program.
- Security team focuses on risk prioritization and task list for the month with IT and application owners.

Case Study Conclusion

- Process first – most organizations want immediate results with a broken process.
- Can’t fix a vulnerability management program especially with the large data sets without fixing the root causes first.
- Having baselines and frequent patch cycles for systems first, and then focusing on deviations is much more achievable.
- Next level after that, the program expands to understanding business risk to individual assets.

Remediate, Mitigate, Accept

Three Choices – Choose Wisely

Goal: Risk Reduction

- The goal for the vulnerability management is to reduce your attack surface which ultimately reduces your risk.
- Categories typically fall into remediate, mitigate, accept.
- Often acceptance is the hardest one for most organizations to understand adequately.

IANS

Remediate

- Remediate the exposure through fixing the direct cause of the vulnerability.
- Patching as an example for remediation.
- Often the best and easiest solution (sometimes) for an organization.

Mitigate

- Mitigate through other methods such as network segmentation or removing an application service to address the security exposure.
- Often times falls in either addressing risk through other controls or what PCI would refer to as “compensating controls”.
- Mitigation is addressing the vulnerability (and ultimately the risk) through other controls that are equal or above the original remediation effort.

Accept

- Accepting risk is applicable when the organization understands the risk.
  - Emphasis on understanding the risk.
- Most business owners truly don’t “understand the risk” and acceptance used only when a full understand of exposures is understood.

Legacy: Patch, EOL, Or Segmentation?

How to handle exceptions.

Legacy, Patch, EOL, Segmentation?

- What's the appropriate answer?
- Do we have these processes discussed?
- What data constitutes one or the other?
- These questions are often the most important part of vulnerability management. What IF you can't do something and it causes the overall security of the system to drop?

Tools and Automation Workflow

When tools and automation workflow can help.

Tools

- There are plenty of tools out there that have varying levels of features based on the organizations needs.
- The tools often don't matter as much as the process of the program.
- Tools can make a difference depending on strengths. Let's discuss some of them.

Primary VM Solutions

- Rapid7 - NeXpose
- Qualys Qualysguard
- Tenable Nessus
- BeyondTrust Retina

- Primary vendors… but what about application security?

AppSec VM

- Source Code:
  - HP, IBM, Checkmarx, Veracode
- Scanners:
  - HP, IBM, Acunetix, Netsparker
  - More manual: Burp Pro
- How is application security integrated into your overall VM program?

Other Findings in TVM?

- What about Penetration Testing Findings?
- Are you tracking new CVEs mapped to an asset inventory? (BIOS??)
  - https://github.com/CVEProject/cvelist
  - https://cve.mitre.org/
- How are you tracking mainframes and databases?
- Other input feeds for the TVM program from other teams?
- Central place to track and maintain.
- Not always just about running a scanner.

Working Together

- VM program should adhere to both full stack operating system exposures (network, OS, etc.) as well as web applications.
- Teams should be working with third party vendors and identifying areas to address risk regardless of it being a web app or a OS patch.
- Root cause analysis (common theme) is so critical during vulnerability management.

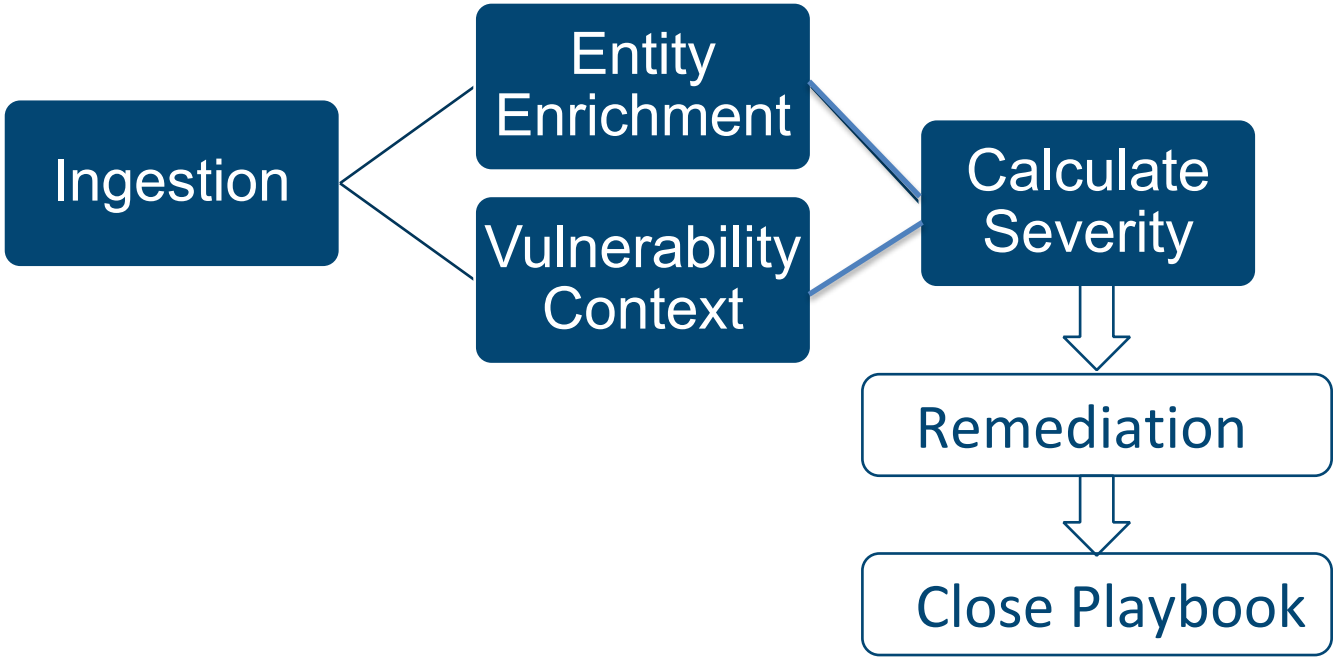
Automation Workflow

- As much as you can automate with vulnerability management the better.
- Custom risk scoring that is automatic is even better based on asset prioritization and weight.
- Integration into tools such as ServiceNow is already in place for most of these tools.
- Additional tools help ingest multiple data sources for more risk-based vulnerability management.
  - Example: Kenna Security, TripWire IP360, Demisto (now Palo Alto), Swimlane, and more.

Speaking of Automation…

- More effort has gone into automating various parts of security operations in recent years
  - Security orchestration, automation and response (SOAR) tools have driven this
- Many security teams are looking to automate some/all of the following:
  - Vulnerability data analysis and “pruning”
  - Adding context in the environment (asset priority, etc.)
  - Validating false positives through other data
  - Prioritizing vulnerabilities and risks

An Automation Example



VM that CAN be automated..

- Vulnerabilities prioritized based on customized risk.
- Automatic notification to asset owners and prioritization.
- Automatic discovery of new assets and automatic asset tagging (if applicable).
- Workflow automation once resolved, scanned and removed.

VM that CAN'T be automated..

- Subject matter experts to help decipher vulnerability data.
- Working with other groups in order to prioritize exposures.
- Root-cause analysis of system exposures and fixing.
- Developing better workflows in order to make the process easier.

Outsourced vs. Internal

- Has anyone successfully outsourced their TVM program?
  - What works?
  - What doesn't?
  - How does working with different owners work?
- Pro/Con of Outsourced vs. Internal
- Familiarity with systems and owners of systems.

Baselines and Hardening

A foundation for long-term programs.

Baselines and Hardening

- You can’t have a VM program without fixing root cause issues.
- Not having hardening guidelines or baselines (standards) will ensure failure to the VM program.
- Baselines should cover core systems and assets as well as new software/applications.

Baselines and Hardening (Cont)

- Think of baseline and hardening guidelines as ways to ensure systems remain protected.
- Secure by default, when deviations occur – adjust accordingly.
- Easiest is operating systems – hardest is new technology coming into the company.

Standards – Starting Off

- DISA STIGs can be a great reference point for starting off.
  - https://iase.disa.mil/stigs/Pages/a-z.aspx
- CIS Benchmarks another great source.
- Note that not all baselines will be applicable.
- Use as a framework, but adopt what's important to the company security and availability.

Deviations and Exceptions

- Exceptions should be documented and approved based on risk approach.
- When deviations occur, wen new exposures are identified – checked against exceptions policies.
- Not every system can be locked down 100%.

Root Cause Analysis

The most challenging aspect of TVM

Root Cause Analysis

- The root cause analysis of something is the research needed in order to determine why an exposure was present.

- Important to differentiate from standard patches of exposures and new exposures that do not fit your TVM program.

Patch Management Explained

- Most organizations do fairly well when it comes to OS patching.
- Third party patch management becomes a large concern especially for the application tier which is not owned by IT.
- Process needs to incorporate all aspects including business owners of assets.

Root Cause Analysis Analysts

- Have seen successful organizations where the TVM has:
  - A project manager or multiple project managers.
  - Root Cause Analysis TVM Analyst(s)
- Ability to work strategically vs. reactive can have long lasting impact to the TVM program.
- Understanding that there will be deviations and critical/high exposures that arise. Addressing them in the future is critical.

Challenging Area: Open Source / Third Party Libraries

- Application security (discussed earlier) is a challenge due to open source snippets and third party libraries.
- Usually application security is a different organization than the TVM team. Why? Discuss?
- Third party libraries that are integrated into our products are difficult to narrow down and provide root cause analysis with the development team.

Improve Discovery and Program

Moving away from the basics.

Improving Discovery and Program

- Improving discovery over time (new network segments or application stacks) is important.
- Nothing is perfect. A VM program can never be perfect.
- Finding new systems or processes to embed into will make the program successful.
- Visibility is king when it comes to VM.
  - New asset discovery (CMDB).
  - New network segments.
  - New business processes.

Incorporating Purple Teams

- TVM groups are a great place to help run purple team exercises.
- Already a program and practice for tracking exposures identified.
- Purple teams are a great way for collaboration between red and blue teams.
- Creating a method for simulating an attack, understanding gaps in weaknesses, and tracking those.
- Where is monitoring and detection weaknesses tracked? SOC? Is that effective?

Threat Intelligence Sources

- What sources of feeds do you use for TI?
- Big buzzword now, the “Tactics, Techniques, and Procedures” (TTPs) of attackers.
- Some of this might be out of scope from the TVM program however having an insight into new attack vectors, where does that come from?
- Do you work with TI teams or third parties in the identification of exposures?
- Who handles strategy roadmap for protection and is TVM incorporated into that?

Conclusion and Comments

Wrapping it up.

Visibility, Visibility, Visibility

- In order for this program to be successful, visibility has to be number one.
- Understanding that VM success relies on collaboration with groups, subject matter experts, and being able to identify new exposures.
- Metrics and KPIs can help with these and benchmarking visibility of assets and systems for the program.

IANS

Wrapping Things Up

- With VM, it's an overwhelming task however is something that is continual and gets better over time.
- There is no silver bullet for VM, it's a program that focuses on the reduction of exposures over time.
- You can only do as much as you have for resources and time. Prioritization based on threat models, and risk can help select what's important.

IANS

Resources

- https://www.sans.org/reading-room/whitepapers/threats/
- https://www.sans.org/reading-room/whitepapers/threats/framework-building-comprehensive-enterprise-security-patch-management-program-34450
- https://www.sans.org/reading-room/whitepapers/threats/applying-lessons-learned-generation-vulnerability-management-system-35997

Questions?

info@iansresearch.com