RSAConference2022
San Francisco & Digital | June 6 – 9

TRANSFORM

SESSION ID: **TECH-T01**

Building a Vulnerability Management Program: How to Eat an Elephant

Megan Benoit

Senior Network Security Engineer
NFM

Disclaimer

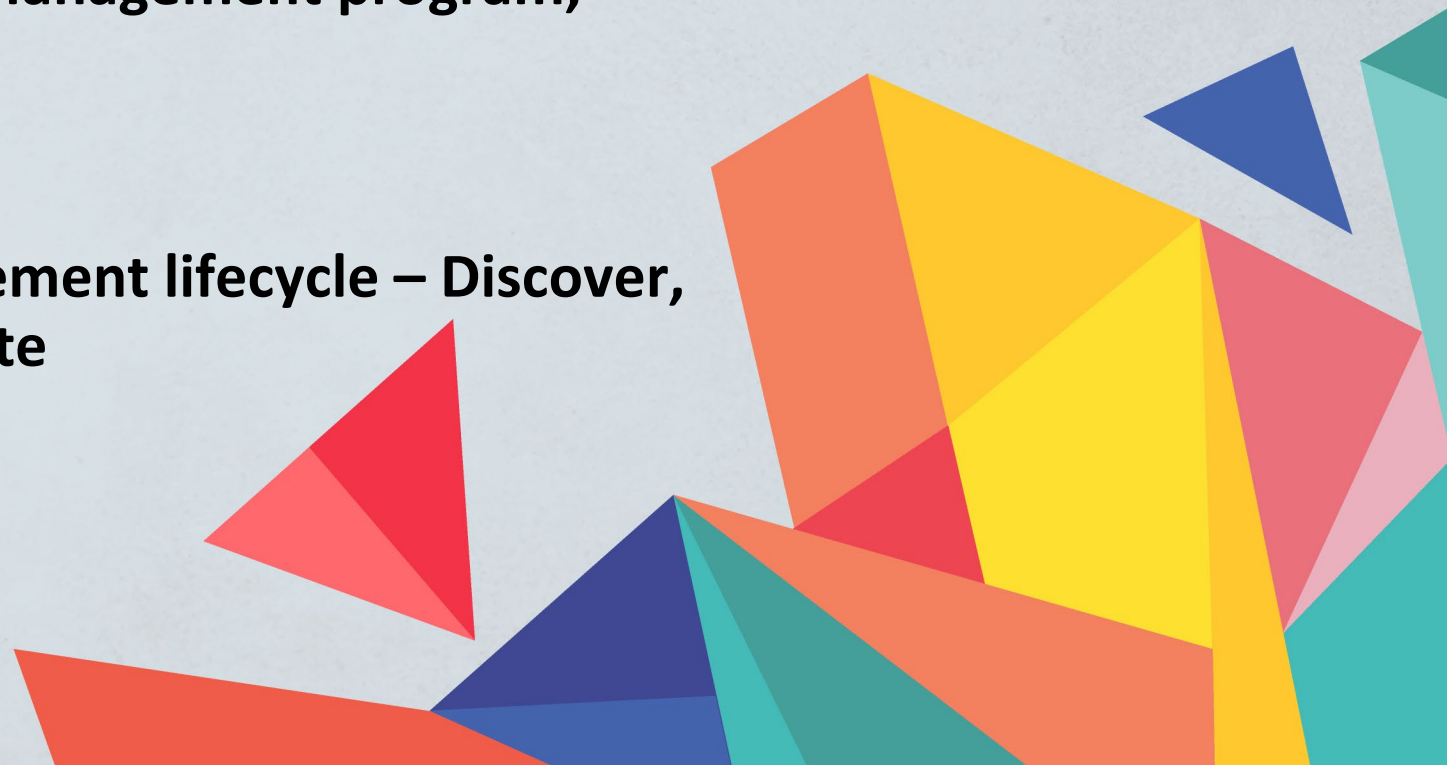
Presentations are intended for educational purposes only and do not replace independent professional judgment. Statements of fact and opinions expressed are those of the presenters individually and, unless expressly stated to the contrary, are not the opinion or position of RSA Conference LLC or any other co-sponsors. RSA Conference does not endorse or approve, and assumes no responsibility for, the content, accuracy or completeness of the information presented.

Attendees should note that sessions may be audio- or video-recorded and may be published in various media, including print, audio and video formats without further notice. The presentation template and any media capture are subject to copyright protection.

RSAConference2022

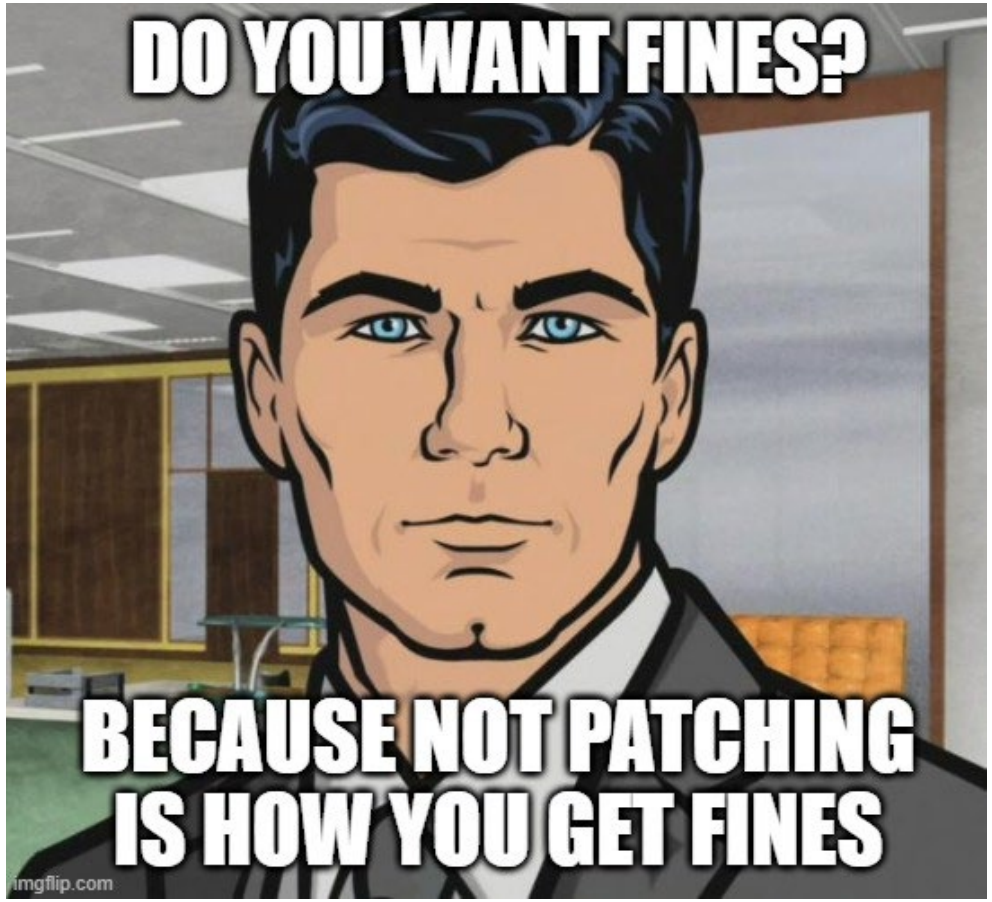
Eating the Elephant

- **Why do you need a Vulnerability Management program, anyway?**
- **Plan and build your program**
- **Navigate the Vulnerability Management lifecycle – Discover, Scan, Prioritize, Remediate, Validate**



Why do I need a vulnerability management program?

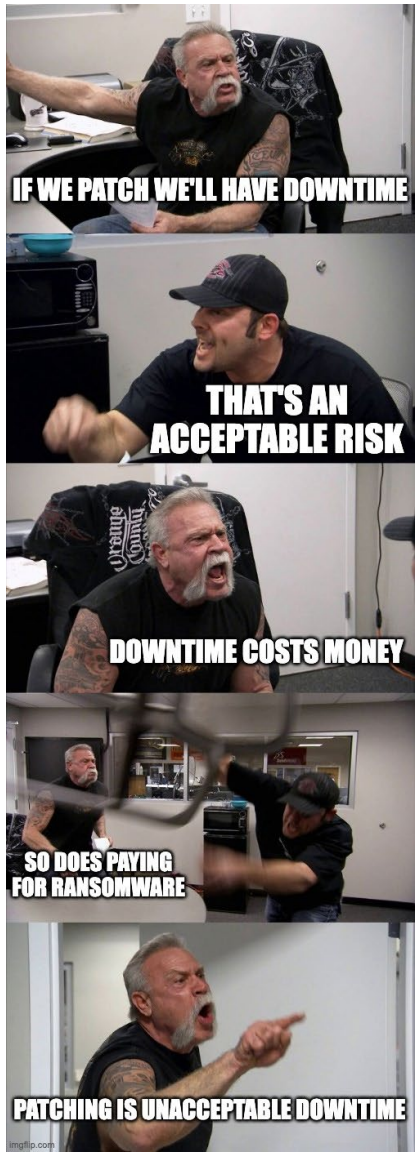
- Vulnerability management is risk management
- PCI/HIPAA/DISA/etc says so
- Hackers don’t want you to patch
- No one wants to be on the front page of the news



How do you build a program from scratch?

- Get management buy-in
- Write (or fix) your vulnerability management and/or patch management policies
- Get the right tools and develop procedures for scanning and reporting
- CYA (document, document, document)

Red Flags



"We'd rather pay the fines than patch."

"We'll just accept the risk"

"We patch annually/quarterly/never because we can't afford to have downtime."

"We don't have the tools to patch so we don't."

"What's the point in patching? There'll just be more patches next month."

Choose your weapon

- You don’t need an expensive tool (but it helps)
- Plan your attack
  - Document requirements
  - Pick the vendors you want to assess
  - Try to stay objective
- Assess your vendors
  - Ask a lot of questions and don’t be afraid to push for answers
  - Assess all the requirements on your list
  - Do a POC and validate everything

Example Requirements Matrix

Requirement	Weight (1-5, 5 is most critical)	Vendor A (1-5, 5 fills requirement completely)	Vendor B (1-5, 5 fills requirement completely)
Scans must complete in a timely manner	4	1 (individual assets can take hours to scan)	4 (individual assets take 5-10 minutes to scan)
Scanner must offer an API for automation	2	5	5
Reporting interface must be simple and easy to use	5	1 (interface is slow, basic reports take hours to run)	3 (interface is easy to navigate but reports can take hours to run)
Vendor must offer an on-premise central console	4	5	1 (vendor only has a cloud offering at this time)
	Weighted Score	**39**	**45**

Plan Your Deployment

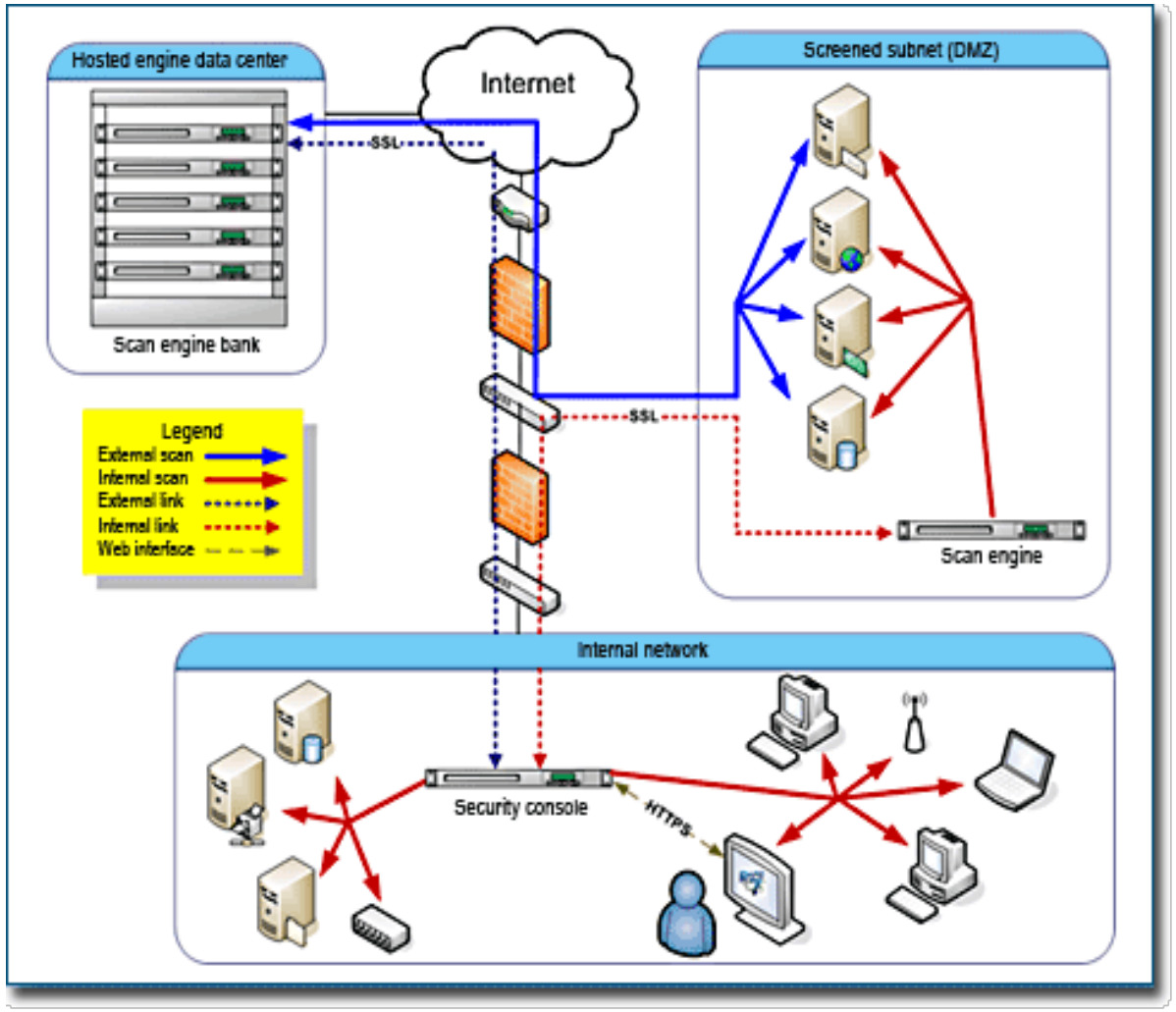
- What does your network look like? Do you have any impediments like low latency links or firewalls in between sites?
- Do the math - how many assets do you need to scan, and in what amount of time?
- Be prepared to add more resources

Credential Management

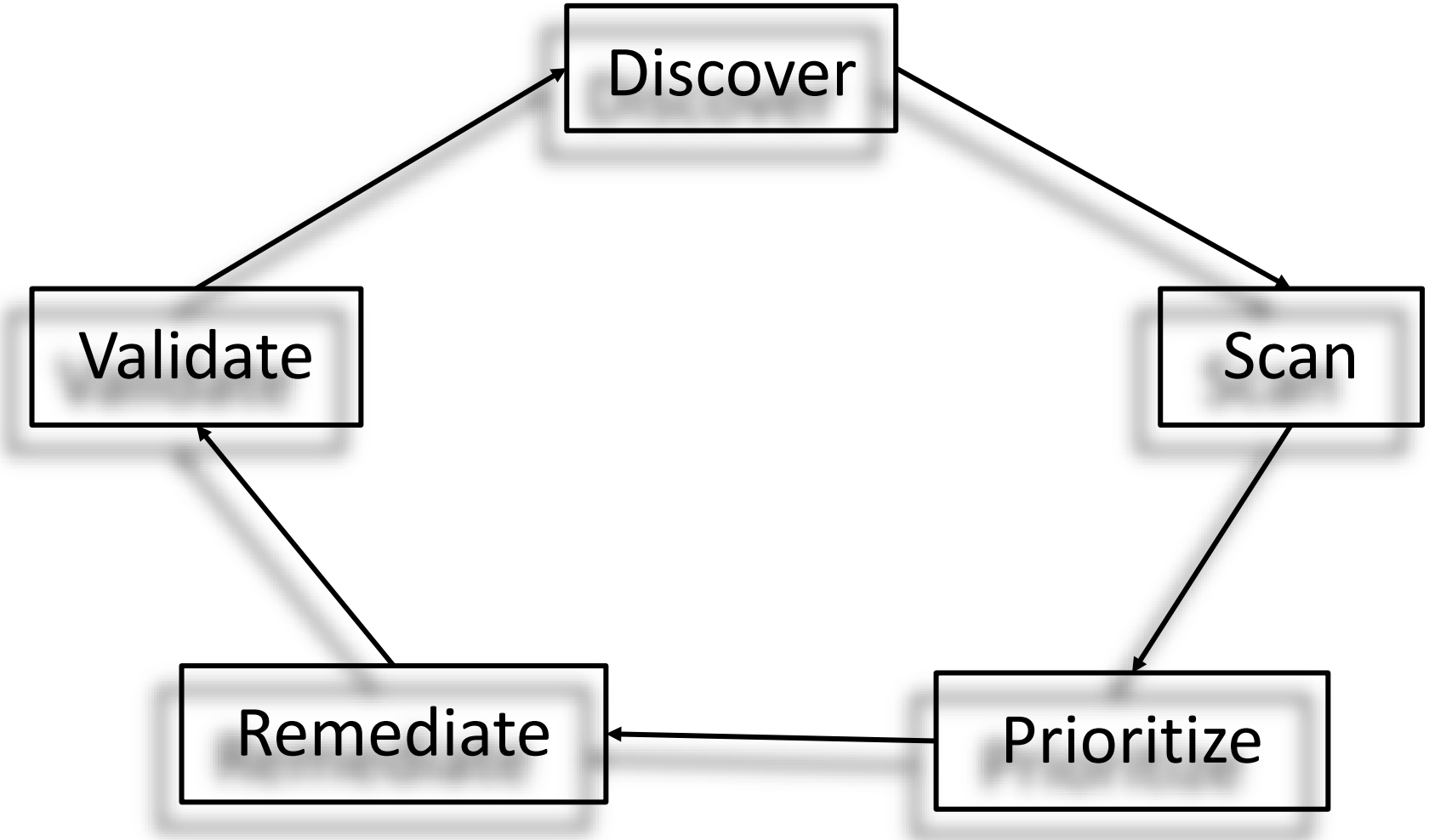
- Work with infrastructure on credentialed scans – every system is different
- Do you have a password manager? How will you reduce risk from cached credentials?
- Test extensively and track authentication failures

Plan your deployment

- Avoid opening firewalls more than necessary - deploy your scanners close to the systems they're scanning
- Be mindful of shared resource limitations on virtual machines
- If your scans are agent-based, who is responsible for deploying and maintaining the agent?

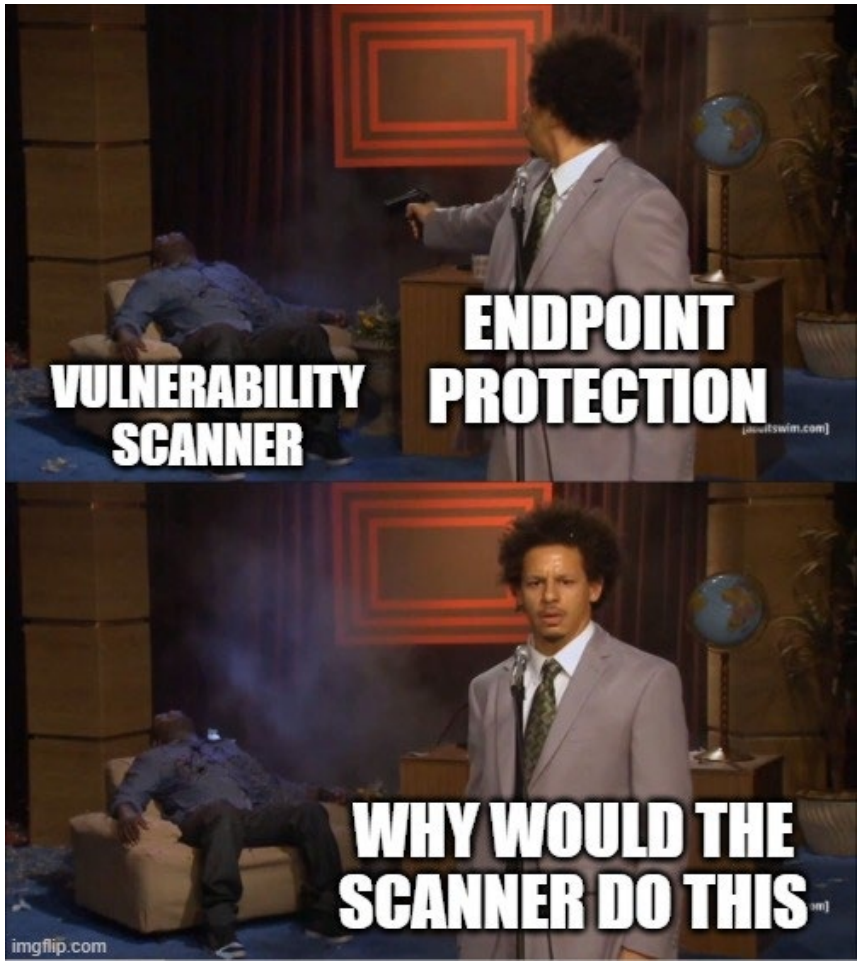


Enter the Lifecycle…



Discover

- Run your discovery scans
  - Use the OS detection
  - The endpoint firewall will shank you – get your scanners whitelisted
  - Cull the herd - watch out for ghosts, VIPs, dead DNS names
- Build your vulnerability scan lists
  - How do you want to group assets?
  - Make sure your credentials are assigned and configured correctly by system, port, protocol, privilege escalation, etc



Scan

- Avoid resume generating events
  - CYA - work with the business on scheduling, follow the change control processes, notify everyone every time you do something
  - Proactively test things that are likely to break
  - You will break things! Be available and ready to kill scans if needed
- Avoid letting the business dictate your requirements
  - No open ended questions - offer options for them to choose from
  - Don’t take “you can’t scan it, it’ll break” as the final answer
  - No one wants you to scan during business hours in case there’s an outage, but if you cause an outage during non-business hours, no one is around to fix it.
- Pick a schedule and stick to it

Double check your scan results!

- "The scans came back clean!" Or your credentials were wrong... or the local firewall blocked your scanner… or you were pointed to the wrong subnet…
- The patch management system says there's no patches required but the scanner says otherwise…
- Talk to your system admins before you hand them hundreds of tickets
- Cross check with endpoint-based agents

Prioritize – Where do you take the first bite?

- Two types of findings - the patch all the systems are missing, and the system missing all of the patches
- Not every finding is “patchable” – can you implement a group policy or other remediation globally to address one issue?
- It can be overwhelming - find one thing that seems manageable and start there

Top Priorities

- Windows OS and Browsers
- The unholy trinity - Acrobat Reader, Chrome, and Java
- Unsupported/Out of date operating systems and software
- Edge devices (routers, firewalls, VPNs)
- Systems containing sensitive info (domain controllers, database servers, HR/Payroll)

Prioritizing Based on Cost and Risk

Cost = User impact + Operational cost + Monetary cost

- Operational cost includes packaging and deploying patches, manual patching, and potential downtime resulting from a patch that breaks business processes
- Monetary cost might be implementing third party patching solution, professional services, etc

Risk = Vulnerability + Threat + Exposure

- Vulnerability - CVSS score, vendor's recommendations, criticality
- Threat - what are your threats? Depending on your industry the threat level can be very high or very low
- Exposure - where do the assets with the vulnerabilities sit? How protected are they?
- See if you can reduce the risk - what mitigations are available to reduce exposure if you can't patch right away?

Prioritizing Based on Cost and Risk

Cost \ Risk	Low	Medium	High
Low	Medium	High	Highest
Medium	Low	Medium	High
High	Lowest	Low	Medium

Organize your findings by cost and risk – lowest cost and highest risk -> highest priority to remediate

Prioritizing Based on the Threat

- CVSS isn't the "full picture" of the threat a vulnerability poses
- Exploit Prediction Scoring System - EPSS – https://www.first.org/epss
- What is the industry worried about? CVE Trends – https://cvetrends.com
- Pay attention to your open source intelligence if you're having trouble deciding what to tackle next

Remediate

- Manage business uptime requirements - have a regular cadence so they know when the reboots are coming
- Leverage any centralized patch management
- How do you handle laptops? Legacy systems? Special equipment?
- Have a plan to document exceptions
- Anticipate problems

Validate

- Remediate, Rescan, Repeat
- APIs and self-service - give the admins access to do validation scans themselves (even if they probably won't)
- Track the time it takes to remediate
- Can you ever really hit 100% compliance? (spoiler alert: probably not)

Follow Through

- Track your progress - use your report data to build graphs showing progress and risk reduction
- Give credit where credit is due - a little gratitude can go a long way
- Don’t give up - collaborate, try different angles of attack, find the smart people and empower them
- Be proactive - make vulnerability scans part of the system build process to make sure they’re up to date before they’re brought online

In a nutshell…

TL;DR -- Vulnerability management is 80% people, 20% patching

- Every organization is different, there is no one size fits all
- Be prepared to change your approach
- Don’t let perfect be the enemy of good
- Keep fighting

Applications

- Next week:
  - Review your existing vulnerability management policies and management buy-in (and if they don't exist, start working on them…)
  - Identify any existing tools for scanning and patching
- 30-day plan:
  - Identify application and infrastructure stakeholders and inform them of your plan to perform scans
  - Perform baseline discovery and vulnerability scanning
- 90-day plan:
  - Prioritize findings and identify key remediation targets
  - Engage with stakeholders to plan remediations

Thank you!

Questions? Comments? Let me know!

@cybertricoteuse