Study Guide

Executive Vulnerability Management

Created By: Ravi Raj, Teaching Assistant

Module 1: Vulnerability Management

Lesson 1.1: Introduction

Skills Learned From This Lesson: Course prerequisites, course structure, Course Overview, Course objective.

- Prerequisites:
  - General understanding of IT and security principles.
  - Should be part of executive management or be an IT security manager.
  - Basic understanding of vulnerability management and associated practices.
  - NIST Risk management framework, patch management lifecycle.
- Supplementary materials:
  - Case studies: Bluekeep, Spectre/Meltdown, Wannacry
  - VM quiz
  - CVSS scoring guide
  - Patch management guidelines.
  - NIST guides SP 800-53A, 800-37, 800-137
  - FIPS 199 and 200.
- Target Audience
  - Someone looking from an executive leadership perspective.
  - Trying to improve on knowledge of vulnerability management.
  - Someone looking to understand challenges in patch management and overcome them.
- Learning Objectives:
  - What is vulnerability management and how executive leadership and improve practices.
  - What is patch management lifecycle and how important it is to daily business practices.

- How executive leadership can support IT security and streamline vulnerability management.

Lesson 1.2: What is Vulnerability Management?
Skills Learned From This Lesson: Managing vulnerabilities, prioritization of vulnerabilities, Risk

- Learning Objectives:
  - What is vulnerability management: Patch management is part of vulnerability management.
  - Why prioritization is important in vulnerability management?
  - How is risk response important to vulnerability management concepts?
- What is Vulnerability Management?
  - Identification, classification, remedy, mitigation of vulnerabilities: Vulnerabilities are not just limited to patches and vary from misconfigurations to cameras not pointing in the right direction for monitoring.
  - Must understand risks to provide patch management or solution: Each vulnerability has an associated risk which must be understood.
  - Continuous information security risk management: Vulnerability management is a continuous process as day on day basis new vulnerabilities are discovered.
  - Executive leadership or security management help to drive processes: Vulnerability management is successful with executive leadership involved.
- Prioritization:
  - Must have an accurate vulnerability discovery process: We can only remediate the vulnerabilities if we know about them. Scans must be authenticated to ensure we are discovering all the possible vulnerabilities.
  - To remediate the vulnerabilities we need to understand the scoring techniques like CVSS which gives an idea of the criticality. Each environment is unique and we need to prioritize the remediation based on its uniqueness. A low score vulnerability may have a more devastating impact if exploited in an environment due to the number of servers it makes vulnerable and overall impact so.
  - What is the “low hanging fruit”: A patch which when deployed may result in remediation on a large scale. A misconfiguration (may be just related to checking a box) which when acted upon remediates a large number of vulnerabilities.

    - How many systems vs impact of vulnerability: An external facing server are more critical. We may prioritize remediation of a web server related to injection attacks compared to an internal server.
- Continuous Vulnerability Management:
    - Patches, hotfixes, rollups are released often: We shouldn't be overwhelmed by the patches rolled out. Big organizations may be able to patch very often unlike smaller or medium size organizations which makes prioritizing important for them.
    - Need to understand the applications/software/hardware in the environment: If we are to use the same software across different line of business it makes work easy for patching teams who otherwise may have to patch different software making the work cumbersome. Need to understand the soft wares needed as per requirement.
    - Need a continuous plan in place to keep up with/ stay ahead of possible issues: A continuous plan ensures that we are reducing the overall risk in the long term.
    - With continuous management, the risk profile will be lower.
    - Easier to deal with zero-day patches/hotfixes: Patches, registry key changes released (like often in case of spectre) management becomes easier with continuous management.
- Risk Response
    - Second part of vulnerability management: What are risk specific to us?
    - How do we address risks?
    - Remediate, mitigate or accept: We may have to accept the risk based on prioritization.
    - Missing patch, need to Install the patch or not: We need to ensure what we pact doesn't break anything.
    - Think about the test environment: Using a test environment for a dry run before patching an actual production environment ensures stability of the environment.

Lesson 1.3: Building a VulnMgmt Program
Skills Learned From This Lesson: Forming VM teams, VM maturity, VM Program

- Learning Objectives

    - How to build a cohesive vulnerability management team.
    - What is involved in a vulnerability Management program?
    - How to implement a vulnerability management program for an organization.
- Building a Team:
    - System admin
    - Network Admin/Engineer
    - Security Operations
    - Security Architect
    - CISO/Executive Leadership

  A mix of people from multiple backgrounds helps to give teams different viewpoints. Involvement of leadership helps to push the team objectives.
- Vulnerability Management program

  There can be several levels of maturity associated with a vulnerability management program
    - Patch management is not equal to Vulnerability management.
    - Need to take a holistic view of the environment: What are the number and the different kinds of servers in our environment? What different applications do we have? How many domain admins/ privilege users we have.
    - Need understanding of all software, hardware, and other equipment/applications used by the organizations: Do we have cloud instances? How our IAM look like.
    - Need to be involved in threat intelligence: Threat intelligence helps in shaping the vulnerability management program. We need to identify what vulnerabilities are targeted against our industry in particular and need to prioritize them.
    - Use risk-based approach: Use various risk based frameworks and tools.
- Vulnerability Management Maturity:
    - Vulnerability Scanning process: What we are using IP/Netbios/DNS for scanning to keep track of vulnerable servers.
    - Asset Discovery and inventory: We need to maintain a proper inventory of assets and application is used.
    - Threat detection (Vulnerabilities / Risk exposure): What is the risk exposure? What are the vulnerabilities and how they expose our environment? Signup for threat feeds like from US-CERT for new vulnerabilities disclosed.

- Reporting and remediation: Valuable report needs to be produced based on prioritization. We need to focus on critical and high vulnerabilities before moving on to Medium risk vulnerabilities.

Lesson 1.4: Security Teams/Responsibilities

Skills Learned From This Lesson: Responsibilities of SOC, Tracking vulnerabilities, Forming a VM Team

- Learning Objectives
  - Who is involved in vulnerability management and what are their responsibilities?
  - What role SOC has to play in VM?
  - How can a team keep track of vulnerabilities?
- Who all are in the team?
  - Monitoring roles: People who are familiar with Data analysis. They can understand the vulnerabilities relevant to your environment. Any automation will be helpful. They reach out to remediation teams for fixing the vulnerabilities.
  - Remediation roles: People who are going to analyze the impact of vulnerabilities (remediated vs not remediated). They come up with complementary controls in case remediation isn't a solution.
  - Authorization roles: Change management people. They coordinate between different teams and ensure that a proper process is followed. Also involve Risk management team which manages the risk factor of the vulnerabilities. Architects who can ensure that the new processes being implemented are as per the accepted standards.
  - Stakeholders: People at executive positions. They drive the VM project.

- How can the SOC help?
  - Maintain security monitoring tools: Alert teams in case of suspicious activities.
  - Investigate suspicious activities: In case if something feels out of line investigate it. As 1 system alerting of a malware can be a common event. In case we find multiple system alerting we should investigate for a possible attack scenario.
  - Executive leadership can empower SOC to work with IT/Infrastructure: Enable coordination between teams. IT focuses on keeping systems up and running. Security ensures minimum possible risk. Leadership should enable a coordination to achieve both.

    - CISO: Determine strategies, policies, procedure to achieve a successful VM implementation.
- Continuous Monitoring Team
    - Vulnerabilities are found daily: People should be scanning and looking into the vulnerabilities on a daily basis.
    - Vulnerability scans are only one component: Scans are not everything. We shall be looking into other things too like source code review. A sql injection can be carried out on a web facing server. We need to look after such cases too.
    - VM programs need to evolve with attackers: As attackers and their TTPs evolve, we should be evaluating that too.
    - Threat Intelligence: Threat intelligence is essential to weed out the vulnerabilities that are less critical. With proper threat intelligence our VM can be very efficient and well driven.

Lesson 1.5: Executive Leadership Role
Skills Learned From This Lesson: Executive management responsibilities, forming a VM Team, Risk committees, Security Policies

- Learning Objectives:
    - Who on the executive team can help VM:
    - How organizations size dictate VM decisions?
    - How a risk committee may help resolve prioritizations issues?
    - How to create an effective security policy?
- Executive management considerations:
    - Drive and prioritize VM: Keep the program on track by planning what needs to be done
    - If VM is not done from top-down, teams may focus on other projects and priorities: Security needs to be prioritized by executive leadership.
    - Not all vulnerabilities need to be remediated: Critical and High vulnerabilities are important to be done. But this isn't all, we need to prioritize based on vulnerabilities that create highest risk specifically to our environment also. We need to ensure that PII data is secured appropriately.
    - Need a technical/advisory team to assist in determining risk profile: An advisory team will help in portray risk appropriately.
- Organization size:

    - Depends on size of leadership team and the size of IT.
    - Smaller Teams- CEO/CIO may need to be involved to drive VM.
    - Mid sized- May have CISO or security management who can take control. CISO will drive the VM and prioritization.
    - Larger Teams- Most likely will have Director of Incident response or Threat Intelligence, who may dictate which threats to address first. In this kind of structure we may have multiple layers of security. A more mature model.
- Risk Committee/Leadership:
    - Risk Committee: Could be an advisory board. Prioritize risk based on their environment.
    - Consider a group of SME's to meet with executive leadership weekly/monthly. If risk modelling is done from the initiation of the project, it ensures that the vulnerability management goes smoothly moving on later stages.
    - Doesn't need to be hours of reports/suggestions: We need to prioritize on what needs to look into immediately.
    - Focus on top 10 threats(Ten most vulnerable hosts, Ten exploitable vulnerabilities with largest footprint). It ensures highest risk is managed on priority.
- Security Policy
    - Approved security policy: Ensures Top down approach discussed earlier
    - Responsibilities and SLAs: Needed to prioritize the critical vulnerabilities remediation.
    - Handle SAAS/Cloud platform vs on premise vulnerabilities: Vulnerabilities differ if we use cloud or own infrastructure. If using a cloud based platform we need to identify what security is being handled by the cloud vendor.
    - Include SME's in policy development: SMEs have a good idea of the environment and will help to drive a VM program based on their in depth knowledge.

Module 2: Tools/Technology

Lesson 2.1: Patch Management Software

Skills Learned From This Lesson: Patch management, Lifecycle, Patch management software

- Learning Objectives:
    - What is patch management?
    - What is the patch management lifecycle?

- What software/methods can assist in Patch management?

- Patch management:
  Following are some important facts and figures associated with patch management emphasizing its importance:
  - 16,500 vulnerabilities reported in 2018
  - Much more difficult for small and medium size organizations: With using many different software it becomes difficult to patch them. Clouds present their own challenges.
  - 57% cyberattack victims believe patches would have prevented attacks.
  - 34% from the same study say that they knew about the vulnerability: If they would have remediated clearly saved.
  - Time between vulnerability disclosure and exploitation: Vary from vulnerability to vulnerability. Citrix vulnerability CVE-2019-19781 was quickly materialized in exploitation. While bluekeep vulnerability CVE-2019-0708 was critical one but not seen being exploited. It is important to patch a vulnerability asap if it is being exploited in the wild.
- Importance of Patch management:
  According to Ponemon institute research 74% companies can't patch fast enough due to lack of resources. It is very difficult for a small IT team to manage all so it is idle to have a team to focus on just patching.
- Lifecycle:
  - Discovery: All the vulnerabilities needs to be discovered
  - Categorize/Prioritize: Categorize on what type of patch it is-OS/Software?
  - Policy Creation/Updates: Create policies and automatic updates where possible.
  - Monitoring(When are new patches released): Track the patch release dates like Microsoft release patches on 2nd Tuesday of month.
  - Testing: Testing patches before deploying ensures that nothing breaks down post patching. Can be difficult for a smaller organization to achieve.

  - Configuration Management(Documentation): Good documentation of patching helps in keeping a track.
  - Roll out: Rolling out a patch. Sometimes if a laptop that is not patched gets attached to a network, it may take a very long time for it to patch due to multiple patches to be installed. So lookout for such cases.
  - Audit: Audit the efficiency of patches. How many actually closed the vulnerabilities.
  - Report/Analyze: Analyze the patches that get failed. Report actual patches closing vulnerabilities.
  - Review/Optimize-back to discovery: Repeat the cycle.
- Software Types:
  - Configuration management software: Use software like Service Now where you can track the flow of tickets for patching vulnerabilities.
  - Patch management software: Software like SCCM, Shavlik. Use tools like this to automate patching.
  - Alerting software- US-CERT: Use feeds for when patches are released.
  - Automation products or programs: Automate the patches on systems if possible.
  - Managed service providers: They can help to upgrade the systems in the environment.

Lesson 2.2: Security Scanning Software

Skills Learned From This Lesson: challenges in Vulnerability scanning, Code review, improving vulnerability scanning process

- Learning Objectives:
  - What is Security scanning software and how it aids in VM?
  - Overcoming common challenges with Vulnerability scanning.
  - What code reviews can do to aid security scanning in software development.
- Security Scanning:

    - Performed with vulnerability scanning software: Using software to scan your environment.
    - Need to have credential IP's/hostnames, appropriate security settings in place to allow scanning: Scans need to be authenticated to ensure best scanning output.
    - Look for common vulnerabilities: CVSS score needs to be taken into consideration for determine criticality.
    - Help to identify and prioritize vulnerabilities: Vulnerabilities needs to be prioritized based on your unique environment and the impact it may have if exploited.
- Code Reviews
    - Audit the source code: Helps in integrating the security from start.
    - Proper security controls are present and working: Ensures that a standard is being followed.
    - Static analysis vs dynamic analysis
    - Must have balanced human review and tool usage: Good to have someone from the team having developing experience.
    - Build security at inception.
- Common Problems:
    - Unauthenticated IPs/ Partial authentication doesn't give a full picture of present vulnerabilities.
    - Missing IPs: Ensure the maximum number and correct IP are scanned.
    - Scans that are conducted using hostnames needs to be ensured are scanning the correct host.
    - High number of false positives needs to be examined and report only for which needs to be remediated.
    - Too many vulnerabilities: Need to prioritize what is needed for time.
    - Inability to prioritize remediation efforts hurts as more critical vulnerabilities might be left open, offering a perpetuator foothold into the environment.
- Resolving Common issues:
    - Check often for authentication issues: Scans must be authenticated, check with infrastructure teams in case of issues.
    - Discovery scans: Help to identify new or missing IPs in the environment.
    - Scan more frequently: Scans should be conducted daily or at least weekly to detect open vulnerabilities.

- Allow your team to identify false positives and accept the risk in case the vulnerability can't be closed.
- Focus on the critical vulnerabilities first and look for the exploitable vulnerabilities on the critical assets as they may have more impact.
- Focus on remediation efforts as if done intensively for a period of time, multiple vulnerabilities can be closed and focus can be shifted on the long term goal of how to manage the vulnerabilities.

Lesson 2.3: Ticketing/Tracking software
Skills Learned From This Lesson: Vulnerability tracking, Tracking workflow, tools used in vulnerability tracking

- Learning Objectives:
  - How ticketing/Tracking software is used to aid vulnerability management.
  - Why is it so important to use tracking software?
  - How to create effective workflows for vulnerability management.
- Tracking Vulnerabilities:
  - First: Identify vulnerabilities. Vulnerabilities via scanners or code reviews need to be identified which will be worked upon.
  - Second: Vulnerabilities should be added to tracking/ticketing software to keep track of remediation
  - Third: Can be sent to appropriate teams for remediation or even executive management to summarize the efforts needed.
  - Can use tools like excel, SharePoint, Jira or serviceNow to track the vulnerabilities.
- Importance of tracking vulnerabilities:
  - Without tracking we can't know what is remediated and what still persists.
  - Need to ensure that proper teams are notified/aware of the vulnerabilities.
  - Once a team acts on a vulnerability, it needs to be verified, and tracking gets easier using the ticketing tools.
  - Teams can keep instructions/remediation steps handy in case a similar issue is discovered in future.
  - All teams kept informed and helped in maintaining a streamlined process and coordinated properly.
- Effective Workflows:

- Ticket is built with vulnerability or affected software: Track each vulnerability
- Approval from management or executive leadership helps in maintaining the ownership and responsibilities.
- Once vulnerabilities are identified, teams are notified of the affected systems.
- Each team may have their own step/process in VM.
- Will need to test each patch before deploying it in production.
- Once each team has completed their piece, they need to affirm the remediation.
- If more work is needed, a ticket is sent back to the appropriate team to act upon.

Lesson 2.4: Required Technical skills

Skills Learned From This Lesson: Security Analyst skills, Security engineer skills, System administrator skills, Network administrator skills

- Learning Objectives:
  - Technical skills required to build a well-rounded vulnerability management team.
  - An analysis of different skills/job positions to include: Security Analyst, Security Engineer, Systems Administrator, Network Engineer, Security Leadership.
- Security Analyst:
  - First responders to incidents or threats.
  - Detect threats, investigate and respond.
  - Must have good data analysis, as well as technical analysis to interpret the technical controls appropriately.
  - Implement additional security measures from management.
  - May have responsibilities in disaster recovery plans.
  - Typically a 24*7 role or have on call rotation.
- Security Engineer:
  - Maintenance of security tools.
  - Recommendations and implementations for new tools.
  - May specialize in SIEM.
  - Security architecture and system development.
  - Work with the operations team to ensure that systems are patched.
  - Document requirements and procedures.
- System Administrator:
  - Responsible for managing systems in an enterprise.

    - May specialize in many applications or OS levels (Windows, Linux).
    - Typically responsible for patch management. Important to involve them as they need know how important it is to patch the vulnerabilities and not feel like a redundant job.
    - Report to CIO or IT management, depending on size of team.
    - Usually wear a lot of hats", work with users and security team.
- Network Engineer:
    - Focus on network administration.
    - Usually focus on one type of network technology.
    - Usually work with security engineers to design secure networks.
    - Support on premise deployment and work for IT management.
- Security Leadership:
    - Deep understanding of their respective private/public regulations and laws.
    - May be security manager of SOC or CISO depending on the size of organization.
    - Proficient in all areas of Cybersecurity, usually report to CIO or CEO.
    - Combination of technical skills and security expertise.
    - Ability to communicate complex problems to upper management in a meaningful way.

Module 3: Common Problem in Vulnerability Management

Lesson 3.1: Patching Cycles

Skills Learned From This Lesson: Software cost models, reducing redundancy in software used, EOL software

- Learning Objectives:
    - Dissimilar patching cycles associated with software.
    - Schedule patch management to improve workflows.
    - Remove EOL software to increase efficiency.
- Patching Schedules:
    - Different type and number of applications in the environment: Control the use of different type of software to minimize efforts.
    - Control the application installations, if not possible identify all the versions out there in the environment to reduce efforts for all the different versions.

  - How many different hardware/firmware are we using?
  - If we are using physical/virtual desktops? Use virtual desktops preferably as it helps save licensing costs and makes patch management easy.
  - How many servers and different OS level are we using? Try to reduce the different versions in the same environment to make patching manageable.
  - Transitions from using older OS like Windows 7 to Windows 10 and make use of the same OS to maintain consistency.
- Improving workflows:
  - Determine level of efforts based on different number of levels and number of applications.
  - How many resources you have in hand decides how easy it will be to maintain the patching levels. You can't expect a quite handful of people to manage a very large environment.
  - Schedule updates in combination, like updating Java and OS together will help in managing the downtime. Thus makes the job of patching easy by combining multiple patches.
  - Do you have private/public sector requirements: Decide the patching tool and schedule that suits your requirement the best.
  - Could you use a different patching program?
  - If behind on patches, consider adding more dedicated resources for a short period of time, It may require pushing back some projects.
- Reduce Redundancy:
  - Need to manage only applications that are required in the environment. Ask users to remove any software that isn't required anymore.
  - How many OS versions do you manage: windows, Linux, Mac? Don't diverse the type of OS used in the environment if not needed.
  - Mobile devices: Android and Apple used in the environment?
  - Can users/organizations share licensing? Helps in maintaining cost and diversity.
  - If multiple organizations/departments: Meet to determine needs and use the shared licensing model to reduce cost and manage patching.
  - Have IT/Security teams make recommendations on software based on what is a safer option and maintains security well.
  - Get rid of End of Life software as soon as possible or have compensatory controls around them to maintain security.

Lesson 3.2: Software/Hardware Requirements

Skills Learned From This Lesson: Software requirements, Hardware requirements, EOL software, technical refresh

- Learning Objectives:
  - How to determine what your software needs are in an organization.
  - How to consolidate/eliminate hardware.
  - How EOL software can affect an organization.
  - What executive leadership needs to know about tech refreshes?
- Software Needs:
  - Every software needs patches/updates. Installing different versions of the same software like Wireshark makes management of software difficult. Try to maintain the same version of software if used all over the environment.
  - Architecture/Design teams should be involved as they help in understanding what the software needs are?
  - Vulnerability management SME's can determine more secure software when a need arises based on how often it is patched and maintained.
  - Can different teams/organizations use similar software: It helps in minimizing the iterations of same software and licensing cost.
  - Use threat modelling techniques to address software concerns.
- Hardware considerations:
  - Identify all the hardware still in use.
  - Still using on-premise or disaster Recovery sites. Hardware at the Recovery site too needs to be patched.
  - Every switch, router, storage device requires software updates.
  - Decommission anything that is not in use as it will only increase the attack surface area.
  - Need to identify and maintain the most critical infrastructure components.

- End of Life(EOL) Software:
  - Maintain a list of all software and keep the list updated, including EOL dates.
  - EOL software has major implications on VM, so need to act upon on priority.
  - Upgrading software should be part of any 5 year business plan.

    - Identify system owners responsible for upkeep and track to reach out for updates if not done timely or not being maintained.
    - Licensing/technical costs should be identified early to allocate appropriate funding
    - Technical staff can be working on Application/OS upgrade to ensure systems are up to date and a plan in place if upgrade is required.
- Technical Refresh:
    - Desktop virtualization: Helps in reducing the overall hardware and only one image needs to be maintained.
    - Application virtualization: Easier to maintain updates and upgrades.
    - Cloud storage: Can reduce costs/improve management, manage just one image.
    - Image management: Can help in reducing administrator overhead.
    - Application White Listing: Can limit the amount of applications installed/run and thus cut the overhead.
    - Segmentation: Helps in maintaining the security of individual segments easier and prevent attacker traversal if one segment is compromised.

Lesson 3.3: Private Sector Requirements

Skills Learned From This Lesson: Cyber security regulations, NIST standards, State regulations

- Learning Objectives:
    - What Private sector requirements you may face in vulnerability management.
    - Different organizations which specialize in private sector cybersecurity requirements.
    - Global and emerging Cybersecurity Policies.
    - Takeaways for Executive Leadership.

- Private Sector
    - NIST Guidance can be used for various aspects and verticals of cybersecurity.
    - Cybersecurity Insurance: Using external help to manage security.
    - ISO 27000 Series with IEC (International Electrotechnical Commission): Lot of great resources and guidance.
    - IEEE: Institute of Electrical and Electronic Engineers has lots of good research.
    - Most vendors like CISCO and CITRIX have very good Best Practice Guides.
- NIST/US-CERT Guidance

- Risk Management Framework(RMF)
- Cybersecurity Framework(CSF)
- NIST 800-53rdr4- Security and privacy controls.
- NIST 800-400-Guide to Enterprise Management Technologies.
- US-CERT Supplemental Resources Guide vol.4: Vulnerability Management.

- Emerging Cybersecurity Regulations
  - NYDFS Cybersecurity Regulations (23 NYCCR 500)-Financial Institutions.
  - DOD CMMI- Cybersecurity Regulation for Governance Contractors.
  - Check “Definitive Guide” from Digital Guardian for each state.
  - Strictest states (by cybersecurity/breach laws)include California, Alabama, Illinois, New Jersey, New York, Ohio, Oregon, South Carolina, South Dakota, Texas, Utah.
- Executive Leadership Takeaways
  - If your business deals with multiple states, you need to be familiar with law/regulations with the area.
  - Have a Vulnerability Management specialist to keep track of changing laws/regulations and ensure your partners do follow a standard for VM.
  - Be aware of what’s going on with NYFDS Cybersecurity regulations.

Lesson 3.4: Public sector Regulations

Skills Learned From This Lesson: Cyber security regulations, NIST standards, State regulations

- Learning Objectives:
  - What Public Sector requirements you may face in vulnerability management.
  - Relevant NIST Guidance with Vulnerability management.
  - Takeaways for Executive Leadership.
- Public Sector Requirements:
  - US-CERT: United States Computer Emergency Readiness Team.
  - NIST: National Institute of Standards and Technology.
  - DHS/CISA- Cybersecurity and Infrastructure Security Agency.
- NIST Guidance:
  - NIST SP 800-63B: Digital Identity Guidelines (Authentication and Lifecycle Management).
  - NIST SP 800-53 r5 (Draft: and Privacy controls.

  - NIST SP 800-161: Supply Chain Risk Management Practices.
  - NISTIR 8179: Criticality Analysis Process Model: Prioritizing systems and components.
  - NIST 800-63- Password Guidelines
  - NIST SP 800-207(Draft)-Zero Trust Architecture
  - NIST SP800-144: Guidelines on security and Privacy in Public Cloud Computing.
- Executive Leadership Takeaways:
  - Have a very technical VM lead to keep up with changing guidance.
  - Need to stay up-to-date with latest regulations/technology.
  - Need to think holistically: VM requirements are vast.
  - VM remediation efforts: Could be a project/assign resources to prioritize remediation.

Lesson 3.5: Vulnerability scoring Methodologies
Skills Learned From This Lesson: CVSS, Tenable VPR, OWASP, Vulncat

- Learning Objectives:
  - Different vulnerability scoring methodologies
  - CVSS scoring
  - Tenable VPR
  - OWASP Risk Rating Methodology
  - VulnCat
- Common Vulnerability Scoring System:
  - Industry standard for providing a numeral score.
  - Current version at 3.1
  - Ranked on Low, Medium, High and Critical
  - Assigns score based on Base, Temporal and Environment Metrics.
  - Vulnerabilities are stored in NVD-National Vulnerability Database.
- Tenable VPR
  - VPR: Vulnerability Priority Rating
  - Use combination of ML and threat intel
  - Helps to determine theoretical risk from actual risk.
  - Uses “Predictive Prioritization”
  - Only available in Tenable.sc
  - Rapid7 has a similar methodology in insightVM.

- OWASP Risk Rating
    - Identify a Risk(Threat agent, attack , etc)
    - Factors for Estimating Likelihood(Skill level, motive, ease of exploit, etc)
    - Factors for estimating Impact(Technical or Business Factors)
    - Determining severity(Informal Method vs Repeatable Method)
    - What to fix?(Most severe)
    - Customize Risk Rating Model(Add/weigh factors)
- VulnCat-Software Security Errors
    - Taxonomy for Security Errors.
    - Vulnerability prioritization should include software security misconfiguration.
    - Order of importance.
    - Input Validation and Representation.
    - API abuse.
    - Security Features.
    - Time and State.
    - Errors.
    - Code quality.
    - Encapsulation

Lesson 3.6: Remediation/Prioritization

Skills Learned From This Lesson: Identification of vulnerabilities, Prioritization, process automation

- Learning Objectives:
    - Ways to identify/determine all vulnerabilities in an environment.
    - How to prioritize vulnerabilities for effective remediation.
    - How to automate vulnerability identification and remediation efforts.
    - How Executives can support the remediation efforts.
- Identification:
    - Can't remediate vulnerabilities effectively if we don't know what we have.
    - First need a full asset list-hardware and software.
    - What projects are currently ongoing-new sw / hw deployments
    - Conduct and update /full inventory before prioritization.
    - Configuration management is a big component of vulnerability management.
- Prioritization:

  - Lots of tools-but what is important to you?
  - Need to know the criticality of each system.
  - Business needs vs customer needs/requirements.
  - Do you have SLA's to meet for customers: Try to use out of business hours for patching to not impact the system availability.
  - Do you follow federal/private sector regulations?
  - Critical systems first but think about exploitable vulnerabilities.
  - Exploitable vulnerabilities may be identified as a medium/high/Critical
  - Considerations for vulnerability Chaining: Vulnerabilities can be used in combination for an attack.
- Automation:
  - Using API tools and Python to automate reporting/alerting.
  - Consider patching immediately.
  - Patch test environment or secondary environment immediately before remediating the production systems.
  - Using virtualization/cloud technology need to maintain only one image to update which makes patching easier.
- Executive Support:
  - With each product/new application comes vulnerabilities and overhead, try to minimize the surface are for attack.
  - Think about new technologies and consolidation of efforts
  - Support security/IT staff remediation efforts.
  - Request reports for top 10 vulnerable systems or top 10 exploitable vulnerabilities to identify key areas that needs prioritization.

Module 4: Solving Vulnerability Management Issues

Lesson 4.1: Aligning Teams

Skills Learned From This Lesson: Coordinating between teams, Developer skills from VM perspective, Leadership skills

- Learning Objectives

  - How members of the security team can work more efficiently.
  - How infrastructure and security teams can combine efforts.
  - How to bring developers into security conversations.
  - How executive leadership can partner with security management
- Security Team
  - Smaller organization-send IT/Helpdesk to train how they can help in addition to their roles.
  - Medium-to large sized organizations-make sure teams communicate with each other and coordinate.
  - ISSE's and ISSO's should work together. Share reports and coordinate. Have defined lines of channel for communication.
  - Encourage transparency in vulnerabilities and issues: People should know whom to reach out to.
- Infrastructure/IT
  - Make sure to include security in weekly meetings and projects to ensure that it is not ignored and considered regularly in the processes.
  - Including security early on alleviates extra overhead and solves the otherwise popping issues.
  - Need to find a balance between security and functionality. Security can't be ignored in the name of functionality.
  - Documentation needs to be clear and to the point, something people can refer to.
- Developers
  - Code must be secure- request gate reviews.
  - Secure early on- develop using latest software/OS versions
  - Send developers to security training.
  - Encourage use of security tools (to automate the process and save time) before code goes to security (find bugs).
- Executive Leadership
  - Stay involved. Weekly operation meetings.
  - Before starting new projects-get all teams involved.
  - Hire a Vulnerability management SME who can work with all the teams and can act as go to person in case of any queries.
  - Build vulnerability management practices into every section of the organization.

Lesson 4.2: Consolidating Products

Skills Learned From This Lesson: Business use case based management, consolidating hardware, consolidating software

- Learning Objectives
    - Scenario 1: Large organization-many departments
    - Scenario 2: Smaller organization-growing quickly
    - Consolidate software to improve VM.
    - Consolidate hardware to improve VM.
- Scenario 1: Business A
    - Large organization –many smaller departments under the umbrella.
    - Each department/section wants its own financial/timekeeping software.
    - Department heads can't agree on one solution.
    - Each have own funding and autonomy to purchase software.
    - Executive leadership should chip in here and ensure that departments coordinate so that there is efficient cost handling and coordination between the teams. Using different software when needs can be managed by a single software only increase the operational cost.
- Scenario 2: Business B
    - Smaller organization- but growing fast, lots of new customers.
    - Individuals are performing many job functions.
    - Don't have enough people to have a full-fledged security team.
    - Need to use open source or free tools as the company is growing.
    - Can hire external teams that provide security as service. It needs to be kept in mind that if the data of customers is compromised in any way it will ultimately result in the loss of business.
- Software/Applications
    - First find out what is installed/purchased.
    - Are users using the applications: Need new requirements? Approve usage of new software only with appropriate business justification.
    - What is the process for purchasing new software: People should not be allowed to buy newer software every now and then as it will only increase the attack surface area over the environment.

    - Legacy software-research and add tech fresh to upgrade software when it gets EOL.
- Hardware
    - What do you have? What can be decommissioned? Remove unsupported EOL products.
    - Consider virtualizing desktop/servers as it helps in cost and support management.
    - Zero clients (vs towers/laptops) come with much less risk.
    - Cloud is a great option-depends on requirements. Not having on premise devices makes management easy and focused with smaller teams. Also it helps in making data backups easy.

Lesson 4.3: Risk analysis/Profile

Skills Learned From This Lesson: Risk, Risk analysis, Risk profile.

- Learning Objectives
    - What Risk Identifications means to Risk Assessment.
    - Why Risk Analysis is crucial to Vulnerability management.
    - Determining your Risk Profile and what it means and what it means to your organization.
- Risk Identification
    - Tangible and intangible sources: Is the risk realistic?
    - Threats vs Opportunities: Someone actually poses a risk?
    - Limitations of knowledge/Reliability: What gaps are there? How reliable is the information being received?
    - Changes in external/internal practices: Need to ensure best practices.
    - Emerging risks/threats: What are the threats lurking over us and actual harm our organization.
    - Biases/assumptions/beliefs: No assumptions or beliefs should be involved in the analysis. We should have data to back up our analysis.
- Risk Analysis
    - Risk Management techniques (including VM): We need to check what creates a risk for the organization.

    - Quantitative: Available data to produce numerical value->predict probability. Based on the data and analysis methods we determine the probability of an actual risk realization.
    - Qualitative: Subjective assessment of risk likelihood against outcomes->overall risk.
    - Delphi Technique, SWIFT Analysis, Decision Tree, Bow-Tie, Probability/Consequence Matrix.
- Risk Evaluation/Profile
    - Compare risk analysis results to risk criteria. Not all risks created are equal.
    - Additional controls required: Do we need compensatory controls?
    - Need a holistic view of risk to the organization.
    - Can help determine boundaries for systems: I just need to focus on risk over systems under my control. But under a holistic view all are interlinked
    - Crucial for stakeholders to make risk-based decisions: Stakeholders need to make risk based decisions.
- Executive Leadership
    - Large organizations have been hit with data breaches: It is important to consider the Cyber risk.
    - Quality risk assessments take time.
    - A 200 page report may not be helpful - ask for most critical assets/risks: Highlight the biggest risks.
    - Need to understand the probability of attack/threats: It is a matter of time, the question isn't will we be attacked but when we will be attacked?

Lesson 4.4: Automating Tasks

Skills Learned From This Lesson: Scanning and Reporting, Automation, Documentation

- Learning Objectives:
    - How to Automate Vulnerability Scanning and Reporting
    - Automating security tasks and threat identification.
    - Using documentation/scripts to help with turnover.
    - Takeaways for executive leadership.
- Scanning and Reporting

    - Created schedule scans for systems-daily/weekly: Make sure the reports are evaluated and feed to the required stakeholders.
    - Make sure to also run discovery scans-new systems/changes: Identify any rogue asset.
    - Attach custom reports to scans-send to system owners or ISSOs
    - Leaves time to research vulnerabilities and emerging threats: Gives security team time for maturing the process and focuses on other things needing their attention.
- Security Tasks
    - Program alerts for malicious file detection/IP traffic: Make sure for important alerts and relevant alerts.
    - Automate alerting for all security systems (server or app availability): Alert for critical functionalities like security devices should be up.
    - Create daily, weekly, monthly alerts for vulnerabilities(new/remediated)
    - Work with IT team to get weekly patching reports: Get regular patching reports to check the systems that have unpatched vulnerabilities
- Documentation/Scripts
    - Script all the things.
    - Don't forget to comment on those scripts to reuse and refer in future.
    - Document why/where scripts are
    - Document security infrastructure-who, what, where, when, why should be what it should be based upon.
- Executive Leadership
    - Encourage IT and security teams to create/maintain documentation: Helps in tracking the process and new people to catch up quickly.
    - Ask for weekly reports on vulnerabilities.
    - Create milestones for (and motivate) teams on automating tasks: Offer incentives to motivate.
    - Hire IT and Security personnel with scripting/automation skills.

Lesson 4.5: Improving Overall Security

Skills Learned From This Lesson: Product consolidation, Teamwork, Vulnerability remediation

- Learning Objectives:
    - Why consolidating products can improve VM.

    - How to effectively prioritize vulnerability remediation.
    - Building teams that communicate and work together.
    - Final takeaways for Executive Leadership and teams.
- Product Consolidation
    - Get rid of EOL software. Encourage teams to find new solutions: Use Open source tools if needed.
    - Virtualization or cloud can lower risk profile as helps in managing the assets better.
    - When creating new IT projects-consolidate old hardware.
    - Have a software specialist research best solutions.
- Vulnerability Remediation
    - Figure out what is critical to your business- data/PII/Applications. Accordingly design the strategies.
    - First identify what vulnerabilities you have.
    - Conduct risk assessment-include all teams.
    - Can't fix everything-focus on most critical / exploitable vulnerabilities
- Teamwork
    - IT, Developers, Administrative, and security teams need to work together.
    - Security is everyone's responsibility.
    - Have a security Liaison in each team-meet weekly
    - Communication- sounds easy, more difficult to implement.
    - Train employees on Vulnerability Management.
- Final Takeaways
    - Awareness is the key to effective vulnerability management.
    - Hire a vulnerability management expert to help to get better insights.
    - Send all teams to security training.
    - Improve communication about vulnerabilities
    - Not a one-time exercise ->Continuous monitoring