

Assignment Title:

Team Plan - Revisited

By

Blue Team

Joshwa Edsall, Michael Moore,

Kristen Netter, Courtney Roberts, Steven Trudell

OPSC-590

Fall 2 2024

By submitting this assignment, I acknowledge that I have read and agree to abide by the Champlain College Academic Honesty Policy. I declare that all work within this assignment is my own or appropriately attributed. I accept that failure to follow the academic honesty policy may result in a failure grade, or expulsion from Champlain College.

Date Due: December 1, 2024

Date Submitted: December 1, 2024

**Introduction**

This document outlines the follow-on actions the Blue Team will implement during the second phase of the cyber warfare simulation. The Blue Team gained knowledge of both their platforms and the Red Team’s platforms in the simulated environment. With this gain of knowledge, the Blue Team modified tools and techniques from their initial plan submitted previously and has additional modifications they will implement in the second phase. Since our initial plan we now know the versions and configurations of our pfSense firewall, our Kali Linux, and our Windows system. Even with this additional information, the Blue Team is aware that documentation is still critical throughout all phases of the simulation.

**Step 1 System Admin**

1. Snapshots were taken of the pfSense Firewall, Kali Linux, and Windows terminals at the start of phase one. Snapshots will also be taken at the start of phase two to ensure the Blue Team has golden images in case there is a need to reset the terminals. These images will assist if there are any issues after modifications or patch updates. The images will also assist with rebuilding the terminals in case of an intrusion or malicious activity.
2. Default settings and default passwords were changed to control access to the environment. The Blue Team will continue to monitor user accounts, to include guest accounts, to ensure there are no unauthorized access.
3. System Monitoring (sysmon) was configured on the Windows terminal; Sysmon will continue to be monitored for the detection of malicious actions. A different system monitoring tool will need to be identified and researched for the Kali Linux environment as sysmom created by Microsoft does not have a Kali Linux version. Potential options for system monitoring on Kali Linux include Gnome, Glances, or Conky. Gnome, Glances, and Conky are system monitoring dash boards, which once installed would assist in identifying unauthorized changes to the terminal.
4. Atera was selected as the patch management software as it offered a 30-day free trial. Atera offered a device scan, which identified the Windows terminal, the Kali Linux terminal, and the pfSense firewall. The Windows and Kali Linux terminals were linked to Atera to identify patches to be updated. While the pfSense Firewall was identified by Atera, it is the decision of the Team Lead for the pfSense Firewall whether to link up the systems. The Blue Team does not want to contradict what each team member is working on, but to enhance each other’s efforts.
   1. The use of Atera was a modification from the original plan to use Heimdal Patch & Asset Management. The selection of Atera of Heimdal was due to the free trial lengths.
5. Additional hardening of the environment will be implemented to prevent intrusions. The additional hardening includes the following:
   1. Blocking applications recommended by Microsoft. The applications identified can be used by malicious attackers to circumvent application allow policies. A full list of applications can be found here: [Applications that can bypass App Control and how to block them | Microsoft Learn](https://learn.microsoft.com/en-us/windows/security/application-security/application-control/app-control-for-business/design/applications-that-can-bypass-appcontrol)
   2. Implementation of AppLocker, an application whitelisting feature built into Windows. Application whitelisting is one of the top 10 mitigation strategies for information assurance and a recommendation of NSA.
   3. Windows Event Forwarding, a repository of host content to aid in collecting security relevant Windows event logs; a companion to spotting the adversary with Windows Event Log Monitoring. This repository contains account lockouts, account login with explicit credentials, create profile fail, etc. This information will assist in identifying suspicious and malicious behavior.
6. All system modifications will be documented. This will allow quick identification of potential errors or misconfigurations. This will specifically assist with sysmon, with the notes of when Blue Team members changed modifications. If the Red Team makes a modification, the Blue Team will be able to identify that change based on the notes.

**Step 2 - System hardening and firewall Rule Configuration**

For phase three, the initial steps for system hardening and firewall rule configuration will remain with a few additions (See Addendum B for initial steps from phase one and two plans). The steps which are being integrated into phase three align with the best practices outlined by Cybersecurity and Infrastructure Security Agency (CISA) and the National Institute of Standards and Technology (NIST). Those steps are as follows:

1. Utilize CISA Windows hardening checklist to ensure all baseline configurations are secure. Additional information on the baseline can be found here: [GitHub - cisagov/ScubaGear: Automation to assess the state of your M365 tenant against CISA's baselines](https://github.com/cisagov/ScubaGear)
2. Revisit and refine firewall rules within pfSense to enhance traffic filtering.
3. Reconfigure snort to ensure accurate altering and minimal false positives, while also leveraging preconfigured rulesets from pfSense.
4. Leverage the NIST SP 800-43 Rev 4 Guide to Enterprise Patch Management Technologies as a baseline for vulnerability management. The full guide can be found here: [Guide to Enterprise Patch Management Planning: Preventive Maintenance for Technology](https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-40r4.pdf)

**Step 3 - Threat Detection**

During phase one and two, Elastic was used for the Security Information and Event Management (SIEM); unfortunately, the free trial for Elastic expired, a new trial will be established to continue to use Elastic as the Blue Team’s SIEM. Once the new trial is established the following steps will be completed:

* + - 1. Establish Elastic on Windows using secondary credentials to establish the SIEM.
      2. Integrate the Windows system and Sysmon to collect logs.
      3. Establish the SIEM components into the Windows policy rules.
      4. Set rules for critical and high alerts.
      5. Enable rules for the Blue Team to identify critical alerts.

Wazuh, a free and open-source security platform, which offers protection for endpoints, a SIEM environment, and Extended Detection and Response (XDR) is being considered for a secondary option. Wazuh monitors systems and application configurations to ensure compliance with security policies, standards, and hardening guidelines. The inclusion of the XDR provides comprehensive security to detect, analyze, and respond to threats.

Since a baseline of behavior was established within the Blue Teams environment, the team will establish a behavior analysis methodology to identify the Red Team’s efforts to attempt to breach the Blue Team’s environment. Behavioral analysis programs spot suspicious behavior through constant analysis of user and malicious actor behavior (Xcitium). The method analyzes network traffic, databases, and user activities to establish a baseline of behavior and sends alerts when there is a change in baseline behaviors (Stanham, 2023).

**Step 4 - Reconnaissance, initial access, persistence, evasion, data exfiltration**

Since both Blue and Red Teams have had a chance to harden their environments, the original plan of manipulating the Red Team environment based on default settings is no longer viable. The Blue Team will try to leverage identified vulnerabilities from the Blue Team assessment of their environment to attack the Red Team’s platform.

Social Engineering is and will always be a concern for any cyber security professional as humans are the weakest link in any cyber defense. Because of that the Blue Team will attempt a social engineering campaign. Evilginx, a man-in-the-middle attack framework, is being research and may be a viable option to conduct a social engineering campaign. Evilginx is used to phish login credentials and session cookies to bypass 2-factor authentication (Evilginx. 2024).

In the initial plan for phase one and two, the Blue Team planned to employ social engineering tactics to gain unwitting or witting insider access to Red Team’s environment. This is still in play for phase three. An email will be drafted to impersonate a team member who is reassigned to the Red Team, requesting access to their environment under the pretext of needing operational information. If unsuccessful, another spear phishing email will be sent from a different Blue Team member containing a malicious link or file, purporting to be urgent information regarding the pfSense Firewall or Kali Linux Virtual Machine. The goal is to trick a Red Team member into granting access or executing the payload, which would provide Blue Team with unvetted access to the target environment.

As a contingency, the Blue Team will also explore the possibility of recruiting an insider threat from the Red Team. This strategy could involve bribing or coercing a Red Team member to intentionally leave an access point open or provide critical information such as passwords. If successful, this would allow Blue Team to bypass standard defenses entirely.

The Blue Team still plans to implement a targeted exploitation strategy to gain and maintain access to critical systems within the Red Team’s network while avoiding detection. The approach will incorporate both technical and social engineering elements aimed at undermining the Red Team’s defenses. During phase one and two of the war games, the Blue Team conducted reconnaissance and discovery to identify the Red Team’s environment. The Blue Team will continue to conduct reconnaissance on the Red Team’s environment, using tools like nmap and Shodan, to identify open services and vulnerabilities.

Once the network has been mapped, Blue Team will focus on exploiting web application vulnerabilities, including identifying weaknesses like SQL Injection (SQLi), Cross-Site Scripting (XSS), and Remote File Inclusion (RFI). Tools like Burp Suite and OWASP ZAP will be used to scan critical web services such as login forms, which often serve as entry points for an attacker. This process will allow Blue Team to manipulate identified vulnerabilities from their assessment to gain a foothold within Red Team's platform.

Additionally, the team will initiate credential harvesting and brute force attacks against weak authentication mechanisms using tools like Hydra and Medusa. This phase will focus on common services such as SSH and FTP, taking advantage of any unchanged default credentials to compromise user accounts, escalate privileges, and pivot deeper into the network.

Following initial access, the team will implement privilege escalation and lateral movement techniques. Using tools like Metasploit, LinEnum, and PowerShell Empire, Blue Team will exploit weak file permissions, kernel vulnerabilities, or misconfigurations to elevate privileges to root or administrator levels. Once administrative access is secured, Blue Team will move laterally across the network, compromising additional systems and critical services.

A key part of the offensive strategy will also involve targeting unpatched vulnerabilities (CVE exploits) in the network. Leveraging well-known CVEs through Metasploit or custom scripts, the team will aim to compromise any outdated systems or applications within Red Team's environment, particularly those vulnerable to high-severity attacks. The primary focus will be on older systems that may not have been updated or patched by the Red Team, maximizing the chances of successful exploitation.

This multifaceted offensive strategy, blending technical exploitation with social engineering, ensures that the Blue Team will be able to penetrate the Red Team’s defenses and maintain operational control throughout the exercise. By targeting both infrastructure and personnel vulnerabilities, the Blue Team will increase the likelihood of success in disrupting the Red Team’s operations while maintaining a stealthy and strategic approach.

**Step 5 - Incident Response**

The Blue Team understands the importance of incorporating Sysmon, process monitoring (Procmon), and Event Viewer in its incident response. Integrating all the aforementioned data points with the SIEM will provide a greater breath of what is occurring in the environment.

The Blue Team will incorporate Windows Debugger and hashing tools to identify malicious files and/or applications. All unnecessary accounts will be disabled and/or deleted. The Blue Team will leverage team members’ knowledge in regard to cleaning and patching software or other issues.

The Blue Team will continue to follow the steps presented in their initial plan for incident response as these steps are best practices. Please note these steps may not incorporate all necessary actions and these phases may run concurrently.

1. *Identification*

Before incident response can start, the identification of an incident is necessary. The first responsibility is to pinpoint the signs of an incident, which can be indicators of compromise (IoC) or indicators of attack (IoA) (Cranford, 2023). If multiple incidents are discovered, the Blue Team will prioritize incidents based on criticality level and handle the most critical incidents first. The Blue Teams IR lead will notify the rest of the team of the incident and coordinate a team response. As mentioned previously, documenting is essential throughout the simulation. Documentation is one of the most critical components throughout IR so that the team understands when, where, and how the event occurred and to improve in follow-ups.

1. *Containment*

After the identification of an incident, the Blue Team will work to contain the incident to prevent further damage to their environment. If an attack is not contained, then the incident can become worse and crash the network infrastructure. Steps in the containment phase include, but are not limited to, blocking traffic (i.e., working with firewall team leads), deactivating compromised accounts, and other actions to save resources and keep other network infrastructure up and running (Exabeam). It is crucial to maintain a balance between practical IT security efforts and regular organizational needs. Note that containment efforts will change depending on the type of attack. For example, a ransomware attack will require a different response than an unauthorized login.

1. *Eradication*

When the IR team has contained the incident, they will take measures to remove malicious files and artifacts in the environment. The IR team, firewall, and system admins will take necessary actions, including but not limited to uninstalling malicious software, deleting malicious files, deleting unauthorized accounts, implementing required patching, and other essential actions (Exabeam). Similar to containment, different eradication actions may need to be taken for various incidents.

1. *Recovery*

Recovery is where the environment is running as designed without signs of infection or an ongoing cyber incident. Actions may include working with the pentest team to ensure that the initially discovered loopholes are closed and tested. This ensures that there is a stable environment. System Admin takes a snapshot of the current stable environment to use as a fallback in the event we need to restore our systems. IR team will support other efforts as needed.

**Step 6 - After Action, Lessons Learn, Weekly Summary.**

After Actions and Lessons Learned are a critical part in any cyber security plan. Discussing what happened, to ensure it does not happen again and improving on responses builds a stronger cyber security team. Due to that, the After Action and Lessons Learn section was not modified as it will be implemented within phase three.

*After-Action Review (AAR):*

A review of what happened, what went well and what could be improved.

*Lessons Learned Meeting:*

What was learned from the incident, which could involve feedback on detection, containment, eradication, and recovery efforts.

*Documentation and Incident Summary Report:*

Documenting all evidence of an incident, to include the timeline, causes, affected systems, detection and response measures, and the impact of the incident. The report will include findings, corrective actions, and areas of improvement.

*Root Cause Analysis (RCA):*

Identifying the root cause of the incident to prevent recurrence by analyzing logs, reviewing configurations, and examining attack vectors to identify how the threat actor infiltrated the system.

*Weekly Summary:*

Summarize incident findings and actions.

*Reflection:*

Evaluate whether existing tools, alerts, and processes were effective in detecting and containing the incident. This may reveal gaps in monitoring, alerting thresholds, or response workflows.

*Security Posture Improvements:*

Implement corrective actions. This could include a new plan of action, additional focused training for the team, or introducing new detection mechanisms/tools needed to mitigate similar incidents in the future.

*Procedure Updates:*

Update incident response plans to reflect the insights gained from the incident. If new threats are identified, ensure that response procedures include steps to address them.

**Summary**

The initial plan for phase one and two was modified for implementation during phase three. While some of the initial plans were removed or modified as knowledge was gained on the Blue Team’s environment, some portions of the initial plan remained. The steps for hardening the environment, incident response and offensive actions were not modified; however, additional steps and/or tools were added. These additional steps and/or tools will bolster what was already in place for the defense of the Blue Team’s environment and the attack phase of the Red Team’s environment.

**References**

Abu-Dabaseh, F., & Alshammari, E. (2018). Automated Penetration Testing: An Overview. Computer Science & Information Technology. Retrieved from https://doi.org/10.5121/csit.2018.80610

Atumu, M. (2024). Tactical server hardening checklist for Windows and Linux security. Retrieved from https://www.liquidweb.com/blog/server-hardening-checklist/#:~:text=Operating%20System%20(OS)%20hardening%20is,Additional%20important%20steps%20include:

Canada Centre for Cyber Security. (2021). Top 10 IT security actions to protect Internet connected networks and information (ITSM.10.089). Retrieved from https://www.cyber.gc.ca/en/guidance/top-10-it-security-actions-protect-internet-connected-networks-and-information-itsm10089

Crandford, J.J. (2023). Incident Response Plan: Frameworks and Steps. Retrieved from https://www.crowdstrike.com/en-us/cybersecurity-101/incident-response/incident-response-steps/

Exabeam. (n.d.). SANS Incident Response: 6-Step Process & Critical Best Practices. retrieved from <https://www.exabeam.com/explainers/incident-response/sans-incident-response-6-step-process-critical-best-practices/>

Evilginx. (2024). Introduction: Evilginx. Retrieved from

Fortinet. (n.d.). What Are Honeypots (Computing)? retrieved from https://www.fortinet.com/resources/cyberglossary/what-is-honeypot

Gartner. (n.d.). Security Information And Event Management (SIEM). retrieved from <https://www.gartner.com/en/information-technology/glossary/security-information-and-event-management-siem>

GitHub. (2019). Windows Event Monitoring Guidance. GitHub. Retrieved from [Event-Forwarding-Guidance/Events at master · nsacyber/Event-Forwarding-Guidance · GitHub](https://github.com/nsacyber/Event-Forwarding-Guidance/tree/master/Events)

GitHub. (2019). AppLocker Guidance. GitHub. Retrieved from [GitHub - nsacyber/AppLocker-Guidance: Configuration guidance for implementing application whitelisting with AppLocker. #nsacyber](https://github.com/nsacyber/AppLocker-Guidance)

Holik, F., Horalek, J., Marik, O., Neradova, S., & Zitta, S. (2014). Effective penetration testing with Metasploit framework and methodologies. 2014 IEEE 15th International Symposium on Computational Intelligence and Informatics(CINTI). Retrieved from https://doi.org/10.1109/cinti.2014.7028682

Kidd, C. (n.d.). SIEM: Security Information & Event Management Explained. retrieved from https://www.splunk.com/en\_us/blog/learn/siem-security-information-event-management.html

LaBarge, R. (2012). Cloud Penetration Testing. International Journal on Cloud Computing: Services and Architecture. Retrieved from https://doi.org/10.5121/ijccsa.2012.2604

Netwrix. (n.d.). Linux Server Hardening and Security Best Practices. Retrieved from https://www.netwrix.com/linux\_hardening\_security\_best\_practices.html

Russinovich, M. & Garnier, T. (2024). Sysmon v.15.15.Retrieved from https://learn.microsoft.com/en-us/sysinternals/downloads/sysmon

SolarWinds. (2024). What is Packet Capture (PCAP)? Retrieved from <https://www.solarwinds.com/resources/it-glossary/pcap>

Souppaya, M. & Scarfone, K. (2022). Guide to Enterprise Patch Management Planning: Preventive Maintenance for Technology. NIST SP 800-40r4.

Stanham, L. (2023). Behavioral Analytics. retrieved from https://www.crowdstrike.com/en-us/cybersecurity-101/exposure-management/behavioral-analytics/?srsltid=AfmBOoqYJ6aDcgvQqmej4xTI-VljPEr3yr7DqCOfpw9WZYmW-2N9HPpU

Tatam, R. (2023). Automatic system hardening: Checklist + examples to ensure system security. Retrieved from https://www.puppet.com/blog/system-hardening

Udochukwu ThankGod Ikechukwu Igwenagu, Abidemi Ayodotun Salami, Abayomi Shamsudeen Arigbabu, Cassandra Esambe Mesode, Tunbosun Oyewale Oladoyinbo, & Oluwaseun Oladeji Olaniyi. (2024). Securing the Digital Frontier: Strategies for Cloud Computing Security, Database Protection, and Comprehensive Penetration Testing. Journal of Engineering Research and Reports, 26(6), 60–75. https://doi.org/10.9734/jerr/2024/v26i61162

Xcitium. (n.d.). What are Types of Detection? retrieved from https://www.xcitium.com/what-are-types-of-detection/

**Appendix A –** Offensive Planned Procedure and Responses

1. **Reconnaissance:**
   1. *Objective:*Gather as much information as possible about the target network before launching any attacks.
   2. *Steps:* 
      1. Passive Reconnaissance:
         1. Use tools like **whois, nslookup,** and **theHarvester** to gather domain, DNS, and email information without alerting the defenders.
         2. Conduct OSINT (Open-Source Intelligence) by checking publicly available information related to the target organization.
      2. Active Reconnaissance:
         1. Use network scanners such as **NMap** to map the network and identify live hosts, open ports, and services running on each machine.
         2. Utilize tools like **Netcat** to manually test services for potential weaknesses.
      3. Planned Response*:*
         1. Gather this information and map out potential entry points.
         2. Identify possible social engineering or phishing entry points based on the information gathered.
2. **Scanning and Enumeration:**
   1. *Objective:*Identify specific vulnerabilities within the discovered services or systems that can be exploited.
   2. *Steps:*
      1. Port Scanning:Use Nmap’s service/version detection (nmap –sV) to gather detailed information on open ports and associated services.
      2. Vulnerability Scanning:Use tools like Nikto and OpenVAS to scan web applications and systems for known vulnerabilities.
      3. Enumeration*:*Tools like enum4linux for SMB enumeration or RPCClient to pull detailed information from target systems.
      4. Planned Response:
         1. If vulnerabilities are found (e.g., outdated services, weak configurations), document them and move to the exploitation phase.
         2. If none are found, escalate with more aggressive scanning techniques.
3. **Exploitation:**
   1. *Objective:*Exploit vulnerabilities to gain access or control over the target systems.
   2. *Steps:*
      1. Web Application Attacks:
         1. SQL Injection:Use SQLMap to identify and exploit SQL injection vulnerabilities in web applications.
         2. Cross-Site Scripting (XSS):Use Burp Suite to identify XSS and injection points that can be leveraged to execute malicious scripts.
      2. Buffer Overflow Attacks:
         1. For any services with outdated versions, attempt buffer overflow attacks using Metasploit’s modules (for example, exploit/windows/smb/ms08\_067\_netapi ).
      3. Password Cracking:
         1. Use Hydra for brute-forcing login pages (e.g., SSH, FTP, web forms)
         2. JohntheRipper or Hashcat to crack password hashes, if any, are captured through network sniffing or compromised credentials.
      4. Planned Response:
         1. If an exploit succeeds, escalate privileges using **Privilege Escalation Exploits** like Linux PrivChecker or Windows Privilege Escalation scripts (e.g., exploit/windows/local/ms16\_032\_secondary\_logon\_handle\_privsec ).
         2. If exploitation fails, pivot by trying lateral movement using credentials or accessing other systems in the network.
4. **Post-Exploitation:**
   1. *Objective:*Maintain access and move laterally through the network, gathering sensitive data and expanding control.
   2. *Steps:*
      1. Establish Persistence:
         1. Use backdoor techniques like creating scheduled tasks or cron jobs to maintain access to compromised systems.
         2. Tools like Netcat or Meterpreter can be used to create persistent shell access.
      2. Privilege Escalation:
         1. Escalate privileges from user to root/admin using known exploits (e.g., Sudo Exploits or Kernel Exploits).
      3. Data Extraction:
         1. Search for sensitive files containing credentials, such as configuration files, password lists, or database dumps.
         2. Use tools like Mimikatz (on Windows) to extract plaintext passwords from memory or LaZagne to extract passwords from browsers and other applications.
      4. Planned Response:
         1. Ensure that any critical data is exfiltrated securely (e.g., using FTP over encrypted channels or Netcat in reverse shell mode).
         2. Document all discovered credentials or sensitive data for further exploitation or post-engagement reporting.
5. **Lateral Movement:**
   1. *Objective:*Move through the network to gain access to additional systems or sensitive data.
   2. *Steps:*
      1. Credential Reuse:
         1. Utilized captured credentials (e.g., through phishing and password cracking) to access other machines on the network.
      2. Pass-the-Hash:
         1. If hashes were captured, use Pass-the-Hash techniques with Metasploit to move across machines in the Windows environment.
      3. SSH Pivoting:
         1. Use SSH tunnels to pivot into other networks or systems based on the compromised machine.
      4. Planned Response:
         1. Always ensure traffic remains encrypted while moving laterally.
         2. Document newly accessed systems and continue to scan for vulnerabilities within these machines.
6. **Cleanup and Reporting:**
   1. *Objective:*Cover tracks and prepare a detailed offensive report on the findings and exploited vulnerabilities.
   2. *Steps:*
      1. Log Clearing:
         1. Remove traces of activity from logs, especially on Linux systems (e.g., using logrotate or manually clearing bash history and logs)
      2. Exit Strategy:
         1. Remove backdoors, malware, or any tools left on the compromised systems before disengaging.
      3. Report:
         1. Document all vulnerabilities, successful exploitation techniques, data gathered, and the impact of these actions.
      4. Planned Response:
         1. The report should include actionable remediation steps for the Blue Team. Provide proof of concepts (PoCs) for each vulnerability exploited, along with detailed logs of the attacks conducted.
7. **Offensive Tools Used:**
   1. Nmap: For network discovery and active scanning.
   2. SQLMap: For automated SQL injection discovery and exploitation.
   3. Burp Suite: For web application testing and XSS/CSRF vulnerability identification.
   4. Hydra: For password brute-force attacks.
   5. Metasploit Framework: For exploiting known vulnerabilities and privilege escalation.
   6. Netcat: For establishing reverse shells and persistence.
   7. Mimikatz/LaZagne: For extracting credentials post-exploitation.
   8. Pass-the-Hash techniques: For lateral movement across Windows machines

**Addendum B - System hardening and firewall Rule Configuration from phase one and two**

1. Keep both Windows and Linux systems up-to-date with the latest versions.

1. Uninstall unnecessary software is not needed on both Windows and Linux.
2. Configure pfSense to block incoming traffic and issue alerts for specific incidents, such as failed firewall rules or suspicious login attempts. This strategy ensures a strong first line of defense against any potential unauthorized access by the Red Team.
3. Rename our built-in admin account for both environments. The built-in admin account names are well known and could potentially be targeted by the Red Team trying to infiltrate our environment.
4. Regularly update our servers with the latest updates and security patches.
5. Identify open network-accessible ports and shut down any processes associated with them.
6. Set account login and lockout policies to limit the number of failed login attempts, reducing the risk of a brute force attack.
7. Regularly conduct network and vulnerability scans and tests to find and fix any issues.