MANDIANT®

WHITE PAPER

Ransomware Protection and Containment Strategies

Practical Guidance for Endpoint Protection, Hardening and Containment

Table of Contents

Overview

Ransomware is a common method of cyber extortion or disruption for financial gain. This type of attack can instantly disrupt access to files, applications or systems until the victim pays the ransom (and the attacker restores access with a decryption key) or the organization restores and reconstitutes from backups. Once ransomware is invoked within an organization, most variants utilize privileged accounts and trust relationships between systems for lateral dispersion.

Ransomware is commonly deployed across an environment in two ways:

1. Manual propagation by a threat actor after they have penetrated an environment and have administrator-level privileges broadly across the environment:
   - Manually run encryptors on targeted systems.
   - Deploy encryptors across the environment using Windows batch files (mount C$ shares, copy the encryptor, and execute it with the Microsoft PsExec tool).
   - Deploy encryptors with Microsoft Group Policy Objects (GPOs).
   - Deploy encryptors with existing software deployment tools utilized by the victim organization.
2. Automated propagation:
   - Credential or Windows token extraction from disk or memory.
   - Trust relationships between systems — and leveraging methods such as Windows Management Instrumentation (WMI), SMB, or PsExec to bind to systems and execute payloads.
   - Unpatched exploitation methods (e.g., EternalBlue — addressed via Microsoft Security Bulletin MS17-010).[1]

The purpose of this document is to provide practical endpoint security controls and enforcement measures which can limit the capability for a ransomware or malware variant to impact a large scope of systems within an environment. If there is an active outbreak, depending upon the propagation method that the variant is leveraging, implementing many of the recommendations within this document can potentially disrupt and contain the event.

While the scope of recommendations contained within this document are not all encompassing, they represent the most practical controls for endpoint containment and protection from a ransomware outbreak. If implemented proactively, the scope of controls outlined within this document can protect an organization from being impacted by a ransomware event that disrupts operations and impacts a large scope of systems.

1 Microsoft (March 14, 2017). Microsoft Security Bulletin MS17-010 - Critical.

Endpoint Hardening

Endpoint Segmentation

Tactic: Lateral dispersion amongst systems using standard Windows Operating System protocols

Windows Firewall

During a ransomware event, many variants utilize privileged and trusted accounts to bind to systems within an environment. Commonly, Server Message Block (SMB) is utilized for the communication channel between systems. While SMB is typically required within a Windows operating environment (e.g., workstation to Domain Controllers or File Servers), the scope of SMB communications permitted directly between systems can be restricted and minimized (e.g., workstation-to-workstation).

During a ransomware event, a Windows Firewall policy can be configured to restrict the scope of communications permitted between common endpoints within an environment. This firewall policy can be enforced locally or centrally via Group Policy. At a minimum, the common ports and protocols that should be blocked between workstation-to-workstation—and workstations to non-Domain Controllers and non-File Servers include:

- SMB (TCP/445, TCP/135, TCP/139)
- Remote Desktop Protocol (TCP/3389)
- Windows Remote Management / Remote PowerShell (TCP/80, TCP/5985, TCP/5986)
- WMI (dynamic port range assigned through DCOM)

Using Group Policy, the settings listed in Table 1 can be configured for the Windows Firewall to restrict inbound communications for endpoints in a managed environment.

Group Policy Setting Path:

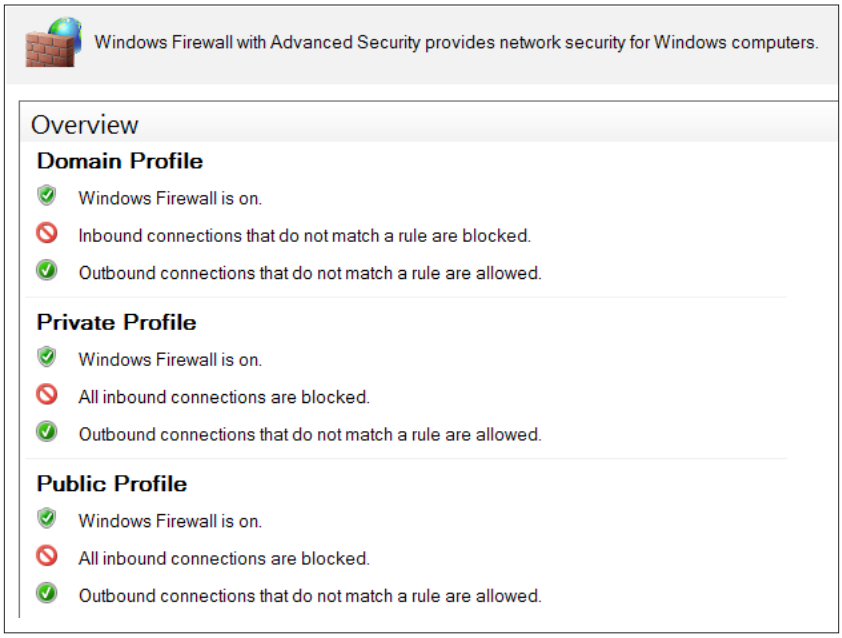
- Computer Configuration > Policies > Windows Settings > Security Settings > Windows Firewall with Advanced Security

Table 1. Windows Firewall recommended configuration state.

Profile Setting	Firewall State	Inbound Connections	Log Dropped Packets	Log Successful Connections	Log File Path	Log File Maximum Size (KB)
Domain	On	Block all connections that do not match a preconfigured rule	Yes	Yes	%systemroot%\system32\LogFiles\Firewall\pfirewall.log	4,096
Private	On	Block All Connections	Yes	Yes	%systemroot%\system32\LogFiles\Firewall\pfirewall.log	4,096
Public	On	Block All Connections	Yes	Yes	%systemroot%\system32\LogFiles\Firewall\pfirewall.log	4,096

Figure 1.

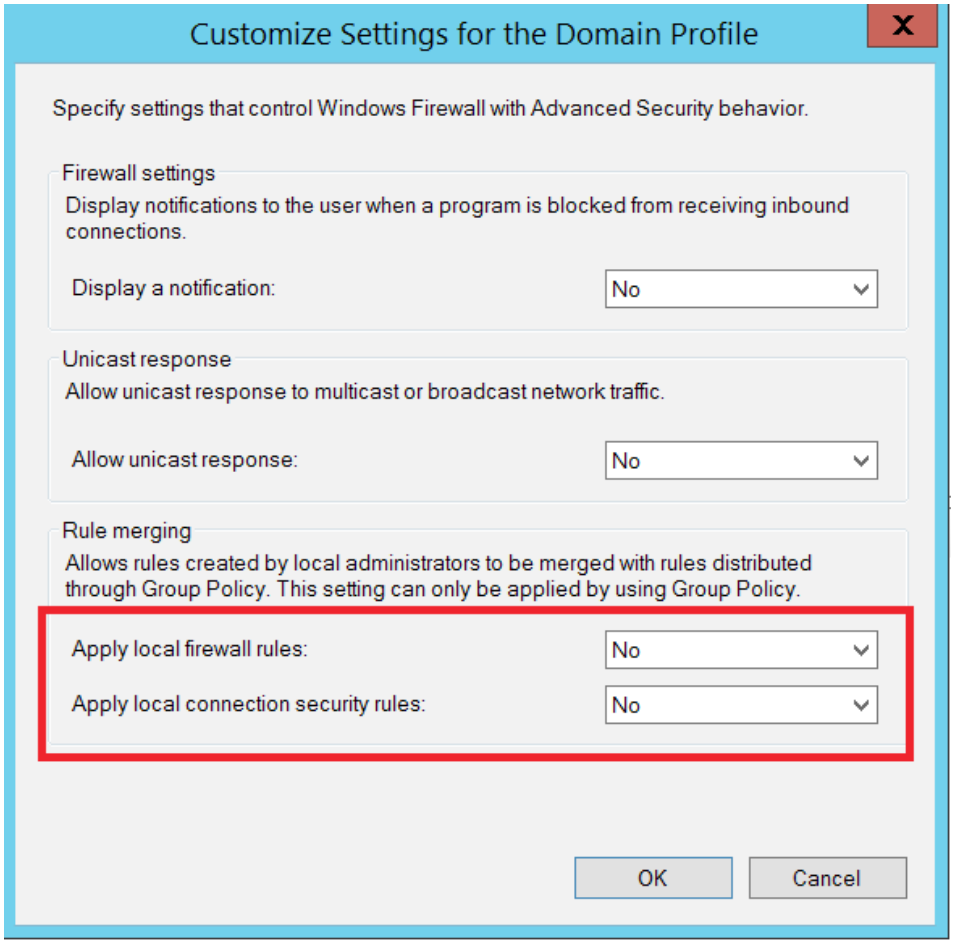
Windows Firewall recommended configurations.



Additionally, to ensure that only centrally managed firewall rules are enforced during a containment event (and cannot be overridden by a nefarious actor), the settings for "Apply local firewall rules" and "Apply local connection security rules" can be set to "No" for all profiles.

Figure 2.

Windows Firewall domain profile customized settings.

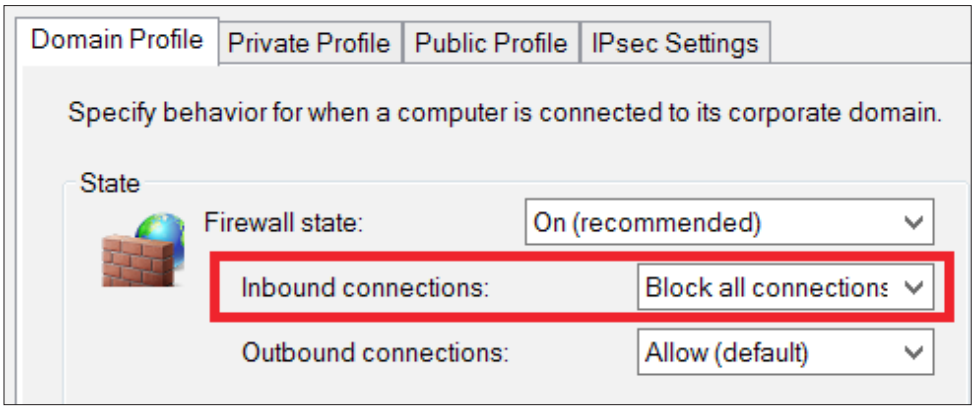


To quickly contain and isolate systems, the centralized Windows Firewall setting of "Block all connections" (Fig. 3) will prevent any inbound connections from being established to a system. This is a setting that can be enforced on workstations and laptops - but will likely impact operations if enforced for servers; although if ransomware is spreading throughout an environment, it may be a necessary step for quick containment.

Note: Once the event has been contained and deemed "safe" to re-establish connectivity amongst systems within an environment, via Group Policy, the "Inbound Connections" setting can be changed back to "Allow" if necessary.

Figure 3.

Windows Firewall - "Block all connections" settings.



The protocols and ports listed in Table 2 represent the most common avenues for lateral movement and propagation. If blocking all inbound connectivity for common endpoints is not practical for containment, at a minimum, the protocols and ports listed in Table 2 should be considered for blocking using the Windows Firewall.

For any specific applications that may require inbound connectivity to end-user endpoints, the local firewall policy should be configured with specific IP address exceptions for origination systems that are authorized to initiate inbound connections to such devices.

Table 2. Windows Firewall suggested block rules.

Protocol / Port	Windows Firewall Rule	Command Line Enforcement
SMB **TCP/445, TCP/139, TCP/135**	Predefined Rule: • File and Print Sharing	`netsh advfirewall firewall set rule group="File and Printer Sharing" new enable=no`
Remote Desktop Protocol **TCP/3389**	Predefined Rule: • Remote Desktop	`netsh advfirewall firewall set rule group="Remote Desktop" new enable=no`
WMI	Predefined Rule: • Windows Management Instrumentation (WMI)	`netsh advfirewall firewall set rule group="windows management instrumentation (wmi)" new enable=no`

Windows Remote Management / PowerShell Remoting
 	
TCP/80, TCP/5985, TCP/5986
	Predefined Rule:
	• Windows Remote Management
	• Windows Remote Management (Compatibility)
	
	Port Rule:
	• 5986
		`netsh advfirewall firewall set rule group="Windows Remote Management" new enable=no`
		
		Via PowerShell:
		`Disable-PSRemoting -Force`

Figure 4.

Windows Firewall suggested rule blocks via Group Policy.

Name	Group	Profile	Enabled	Action
WinRm via HTTPs - Block Inbound		All	Yes	Block
File and Printer Sharing (Echo Request - ICMPv4-In)	File and Printer Sharing	All	Yes	Block
File and Printer Sharing (Echo Request - ICMPv6-In)	File and Printer Sharing	All	Yes	Block
File and Printer Sharing (LLMNR-UDP-In)	File and Printer Sharing	All	Yes	Block
File and Printer Sharing (NB-Datagram-In)	File and Printer Sharing	All	Yes	Block
File and Printer Sharing (NB-Name-In)	File and Printer Sharing	All	Yes	Block
File and Printer Sharing (NB-Session-In)	File and Printer Sharing	All	Yes	Block
File and Printer Sharing (SMB-In)	File and Printer Sharing	All	Yes	Block
File and Printer Sharing (Spooler Service - RPC)	File and Printer Sharing	All	Yes	Block
File and Printer Sharing (Spooler Service - RPC-EPM...	File and Printer Sharing	All	Yes	Block
Remote Desktop - Shadow (TCP-In)	Remote Desktop	All	Yes	Block
Remote Desktop - User Mode (TCP-In)	Remote Desktop	All	Yes	Block
Remote Desktop - User Mode (UDP-In)	Remote Desktop	All	Yes	Block
Windows Management Instrumentation (ASync-In)	Windows Managemen...	All	Yes	Block
Windows Management Instrumentation (DCOM-In)	Windows Managemen...	All	Yes	Block
Windows Management Instrumentation (WMI-In)	Windows Managemen...	All	Yes	Block
Windows Remote Management (HTTP-In)	Windows Remote Ma...	All	Yes	Block
Windows Remote Management (HTTP-In)	Windows Remote Ma...	All	Yes	Block

Additionally, the Windows Firewall can be configured to block specific binaries from making outbound connections on endpoints. During ransomware response engagements, Mandiant has identified legitimate Windows binaries being leveraged to download backdoors and encryptors from both internal and external locations. To protect against this tactic, an organization can leverage a series of Windows Firewall rules to block specific binaries from making outbound connections from an endpoint.

Using powershell.exe and bitsadmin.exe as examples, the figure below provides configurations of leveraging Windows Firewall rules to deny the ability for specific binaries to establish outbound connections from an endpoint.

Figure 5.

Windows Firewall rule example to block specific binaries from making outbound connections on an endpoint.

Outbound Rules

Name	Description
Bitsadmin - Outbound Blocking	
This rule might contain some elements that cannot be interpreted by the current version of GPMC reporting module	
Enabled	True
Program	%systemroot%\System32\bitsadmin.exe
Action	Block
Authorized computers	
Protocol	Any
Local port	Any
Remote port	Any
ICMP settings	Any
Local scope	Any
Remote scope	Any
Profile	All
Network interface type	All
Service	All programs and services
Group	
PowerShell - Outbound Blocking	
This rule might contain some elements that cannot be interpreted by the current version of GPMC reporting module	
Enabled	True
Program	%systemroot%\System32\WindowsPowerShell\v1.0\powershell.exe
Action	Block
Authorized computers	
Protocol	Any
Local port	Any
Remote port	Any
ICMP settings	Any
Local scope	Any
Remote scope	Any
Profile	All
Network interface type	All
Service	All programs and services
Group	
PowerShell - Outbound Blocking	
This rule might contain some elements that cannot be interpreted by the current version of GPMC reporting module	
Enabled	True
Program	%systemroot%\SysWOW64\WindowsPowerShell\v1.0\powershell.exe
Action	Block
Authorized computers	
Protocol	Any
Local port	Any
Remote port	Any
ICMP settings	Any
Local scope	Any
Remote scope	Any
Profile	All
Network interface type	All
Service	All programs and services
Group	

RDP Hardening

Remote Desktop Protocol (RDP) is a common method used by malicious actors to remotely connect to systems, laterally move from the perimeter onto a larger scope of systems, and deploy malware. External-facing systems with RDP open to the Internet have elevated risk. Malicious actors may exploit RDP to gain initial access into an organization, perform lateral movement, invoke ransomware, and potentially access and steal data.

Proactively, organizations should scan their public IP address ranges to identify systems with RDP (TCP/3389) and other protocols (SMB – TCP/445) open to the Internet. At a minimum, RDP and SMB should not be directly exposed for ingress and egress access to/from the Internet. If required for operational purposes, explicit controls should be implemented to restrict the source IP addresses which can interface with systems using these protocols.

Enforce Multi-Factor Authentication

If external-facing RDP must be utilized for operational purposes, multi-factor authentication should be enforced for connectivity. This can be accomplished either via the integration of a third-party multi-factor authentication technology or by leveraging a Remote Desktop Gateway and Azure Multi-Factor Authentication Server using RADIUS.[2]

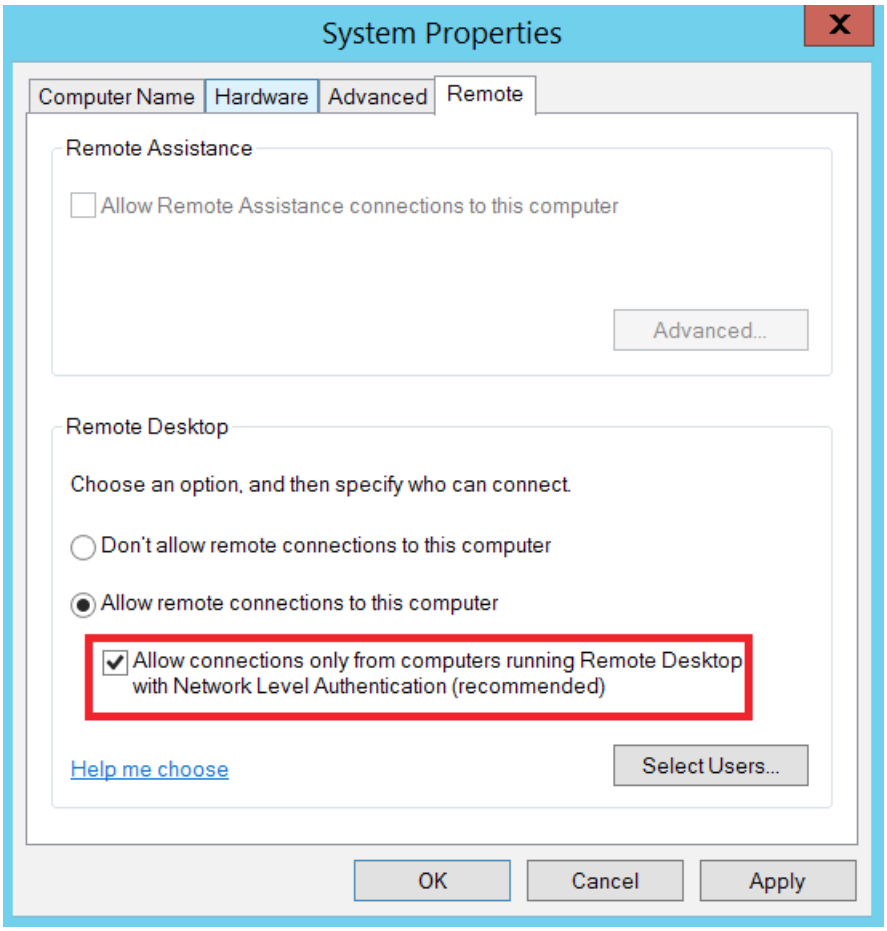
Leverage Network Level Authentication

For external-facing RDP servers, Network Level Authentication (NLA) provides an extra layer of pre-authentication before a connection is established. NLA is also useful for protecting against brute force attacks, which often target open internet-facing RDP servers.

NLA can be configured either via the User Interface (UI) (Fig. 6) or via Group Policy (Fig. 7).

Figure 6.

Enabling NLA via the UI.



2 Microsoft (July 10, 2018). Remote Desktop Gateway and Azure Multi-Factor Authentication Server using RADIUS.

Using Group Policy, the setting for NLA can be enabled via:

- Computer Configuration > Policies > Administrative Templates > Windows Components > Remote Desktop Services > Remote Desktop Session Host > Security > Require user authentication for remote connections by using Network Level Authentication

Figure 7.

Enabling NLA via Group Policy.

Setting	State	Comment
Server authentication certificate template	Not configured	No
Set client connection encryption level	Not configured	No
Always prompt for password upon connection	Not configured	No
Require secure RPC communication	Not configured	No
Require use of specific security layer for remote (RDP) connections	Not configured	No
Do not allow local administrators to customize permissions	Not configured	No
Require user authentication for remote connections by using Network Level Authentication	Enabled	No

Some caveats about leveraging NLA for RDP:

- The Remote Desktop client v7.0 (or greater) must be leveraged.
- NLA utilizes CredSSP to pass authentication requests from the initiating system. CredSSP stores credentials in LSA memory on the initiating system—and these credentials may remain in memory even after a user logs off from the system. This provides a potential exposure risk for credentials in memory on the source system.
- On the RDP server, users permitted for remote access using RDP must be assigned the "Access this computer from the network" privilege when NLA is enforced. This privilege is often explicitly denied for user accounts to protect against lateral movement techniques.

Restrict Administrative Accounts from Leveraging RDP on Internet-Facing Systems

For external-facing RDP servers, highly-privileged domain and local administrative accounts should not be permitted access to interface with the servers using RDP (Fig. 8).

This can be enforced using Group Policy, configurable via the following setting:

- Computer Configuration > Policies > Windows Settings > Security Settings > Local Policies > User Rights Assignment > Deny log

Figure 8.

Group Policy configuration for restricting highly privileged domain and local administrative accounts from leveraging RDP.

Local Policies/User Rights Assignment

Policy	Setting
Deny access to this computer from the network	Local account and member of Administrators group, MCWHIRT\Domain Admins, MCWHIRT\Enterprise Admins, MCWHIRT\Schema Admins, MCWHIRT\Tier0-DomainAdmins, MCWHIRT\Tier0-ExchangeAdmins, MCWHIRT\Tier1-ServerAdmins, MCWHIRT\Tier1-ServiceAccounts
Deny log on as a batch job	Local account and member of Administrators group, MCWHIRT\Domain Admins, MCWHIRT\Enterprise Admins, MCWHIRT\Schema Admins, MCWHIRT\Tier0-DomainAdmins, MCWHIRT\Tier0-ExchangeAdmins, MCWHIRT\Tier1-ServerAdmins, MCWHIRT\Tier1-ServiceAccounts
Deny log on as a service	Local account and member of Administrators group, MCWHIRT\Domain Admins, MCWHIRT\Enterprise Admins, MCWHIRT\Schema Admins, MCWHIRT\Tier0-DomainAdmins, MCWHIRT\Tier0-ExchangeAdmins, MCWHIRT\Tier1-ServerAdmins, MCWHIRT\Tier1-ServiceAccounts
Deny log on locally	MCWHIRT\Domain Admins, MCWHIRT\Enterprise Admins, MCWHIRT\Schema Admins, MCWHIRT\Tier0-DomainAdmins, MCWHIRT\Tier0-ExchangeAdmins, MCWHIRT\Tier1-ServerAdmins, MCWHIRT\Tier1-ServiceAccounts
Deny log on through Terminal Services	Local account and member of Administrators group, MCWHIRT\Domain Admins, MCWHIRT\Enterprise Admins, MCWHIRT\Schema Admins, MCWHIRT\Tier0-DomainAdmins, MCWHIRT\Tier0-ExchangeAdmins, MCWHIRT\Tier1-ServerAdmins, MCWHIRT\Tier1-ServiceAccounts

Disable Administrative / Hidden Shares

```
Tactic: Lateral dispersion amongst systems via binding to
  administrative shares for tool or malware deployment
```

Some ransomware variants will attempt to identify administrative or hidden network shares, including those that are not explicitly mapped to a drive letter—and use these for binding to endpoints throughout an environment. As a containment step, an organization may need to quickly disable default administrative or hidden shares from being accessible on endpoints. This can be accomplished by either modifying the registry, stopping a service, or by using the "Microsoft Security Guide" Group Policy template from the Microsoft Security Compliance Toolkit.[3]

Common administrative and hidden shares on endpoints include:

- ADMIN$
- C$
- D$
- IPC$

Note: Disabling administrative and hidden shares on servers, specifically Domain Controllers, may significantly impact the operation and functionality of systems within a domain-based environment.

Additionally, if PsExec is utilized in an environment, disabling the admin (ADMIN$) share can restrict the capability for this tool to be utilized to remotely interface with endpoints.

Registry Method:
Using the registry, administrative and hidden shares can be disabled on endpoints (Fig. 9 and Fig. 10).

Workstations:

```
HKLM\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters
DWORD Name = "AutoShareWks"
Value = "0"
```

Figure 9.
Registry value for disabling administrative shares on workstations.

Servers:

```
HKLM\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters
DWORD Name = "AutoShareServer"
Value = "0"
```

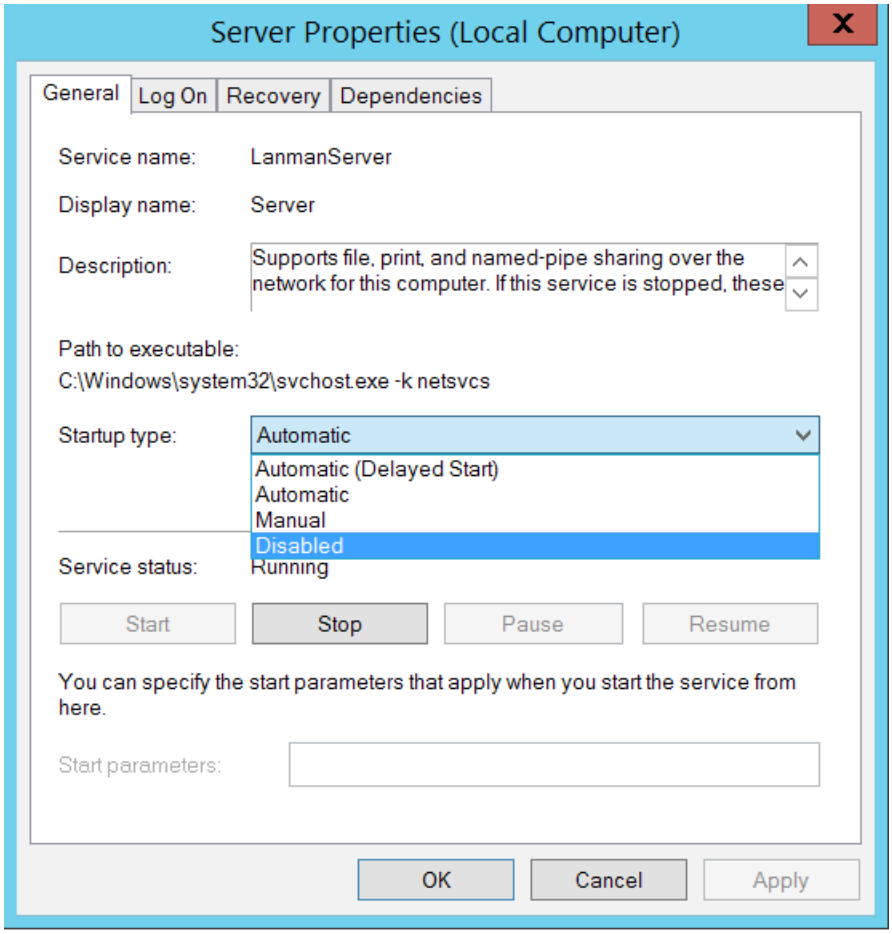
Figure 10.
Registry value for disabling administrative shares on servers.

3 See https://www.microsoft.com/en-us/download/details.aspx?id=55319

Service Method:
By stopping the "Server" service on an endpoint, the ability to access any shares hosted on the endpoint will be disabled (Fig. 11).

Figure 11.

"Server" Service Properties.



Group Policy Method:
Using the "MSS (Legacy)" Group Policy template, administrative and hidden shares can be disabled on either a server or workstation using Group Policy settings (Fig. 12).

- Computer Configuration > Policies > Administrative Templates > MSS (Legacy) > MSS (AutoShareServer)
- Computer Configuration > Policies > Administrative Templates > MSS (Legacy) > MSS (AutoShareWks)

Figure 12. Disabling administrative and hidden shares via the "MSS (Legacy)" Group Policy template.

Setting	State	Comment
MSS: (AutoAdminLogon) Enable Automatic Logon (not recommended)	Not configured	No
MSS: (AutoReboot) Allow Windows to automatically restart after a system crash (recommended e...	Not configured	No
MSS: (AutoShareServer) Enable Administrative Shares (recommended except for highly secure envi...	Disabled	No
MSS: (AutoShareWks) Enable Administrative Shares (recommended except for highly secure enviro...	Disabled	No

Disable SMB v1

```
Tactic: Lateral dispersion amongst systems via vulnerability exploitation or
legacy protocol abuse
```

In addition to patching for known vulnerabilities impacting common protocols (e.g., SMB)[4], disabling SMB v1 on endpoints can reduce the mass propagation methods used by specific ransomware variants.

SMB v1 can be disabled on Windows 7 and Windows Server 2008 R2 (and above) using either PowerShell (Fig. 13), a registry modification, or by using the "Microsoft Security Guide" Group Policy template from the Microsoft Security Compliance Toolkit.[5]

PowerShell Method:

Figure 13.
PowerShell command to disable SMB v1.

```
Set-SmbServerConfiguration -EnableSMB1Protocol $false
```

Registry Method:
Using the registry, SMB v1 can be disabled on endpoints (Fig. 14 and Fig. 15).

Figure 14.
Registry key and value for disabling SMB v1 server (listener).

Disable SMBv1 Server:

```
HKLM\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters
Registry entry: SMB1
REG_DWORD = "0" (Disabled)
```

4 Microsoft (October 10, 2017). Microsoft Security Bulletin MS17-010 - Critical.
5 See https://www.microsoft.com/en-us/download/details.aspx?id=55319

Figure 15.

Registry key and value for disabling SMB v1 client.

Disable SMBv1 Client:

```
HKLM\SYSTEM\CurrentControlSet\services\mrxsmb10
Registry entry: Start
REG_DWORD = “4” (Disabled)

HKLM\SYSTEM\CurrentControlSet\Services\LanmanWorkstation
Registry entry: DependOnService
REG_MULTI_SZ: “Bowser”,“MRxSmb20”,“NSI”
```

Group Policy Method:

Using the “Microsoft Security Guide” Group Policy template, SMB v1 can be disabled using the settings noted below.

- Computer Configuration > Policies > Administrative Templates > MS Security Guide > Configure SMB v1 Server

Figure 16. Disabling SMB v1 server via the “MS Security Guide” Group Policy template.

Setting	State	Comment
Configure SMB v1 server	Disabled	No
Configure SMB v1 client driver	Enabled	No
Configure SMB v1 client (extra setting needed for pre-Win8.1/2012R2)	Enabled	No

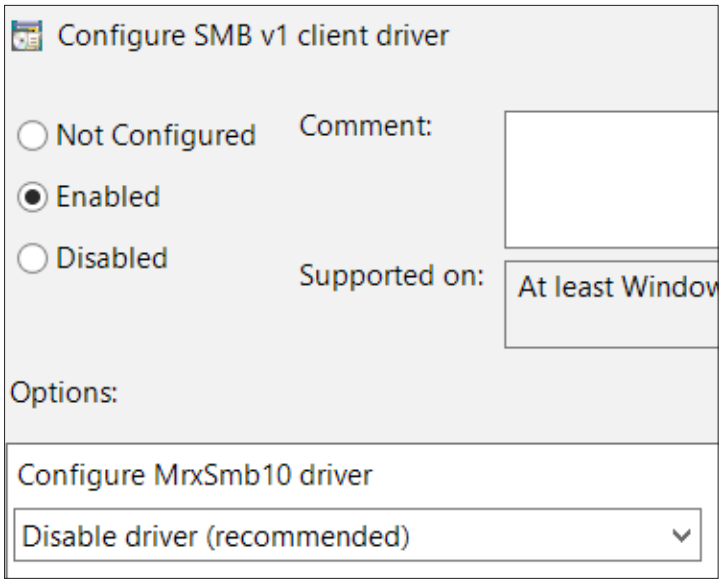
- Computer Configuration > Policies > Administrative Templates > MS Security Guide > Configure SMB v1 Client Driver
  - Enabled
    - Configure MrxSMB10 driver
      - Disable driver

Figure 17. Disabling SMB v1 client driver via the “MS Security Guide” Group Policy template.

Setting	State	Comment
Configure SMB v1 server	Disabled	No
Configure SMB v1 client driver	Enabled	No
Configure SMB v1 client (extra setting needed for pre-Win8.1/2012R2)	Enabled	No

Figure 18.

Disabling SMB v1 client driver via the “MS Security Guide” Group Policy template – additional setting.



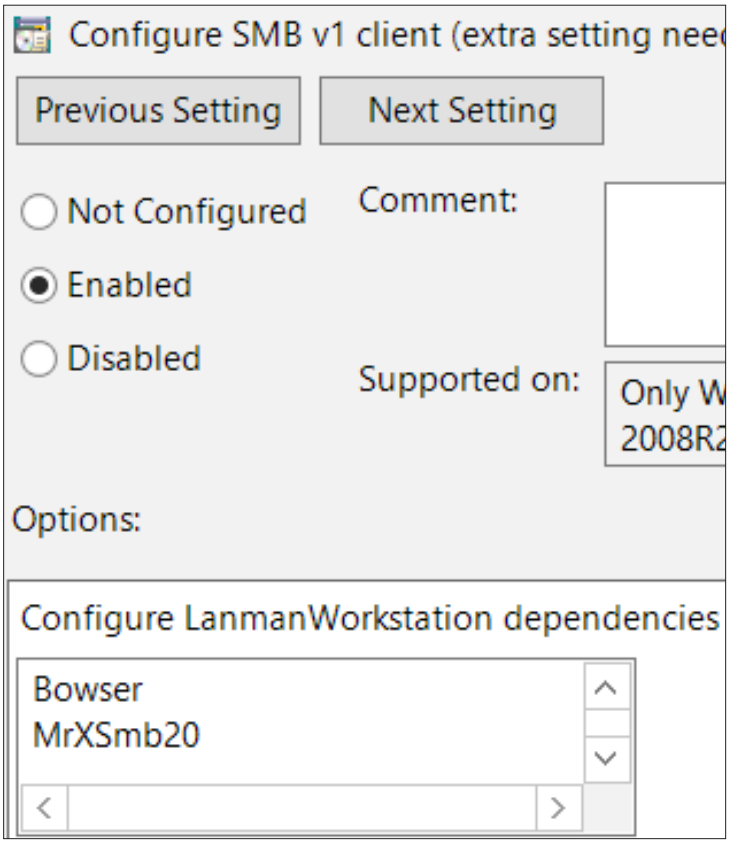
- Computer Configuration > Policies > Administrative Templates > MS Security Guide > Configure SMB v1 Client (extra setting needed for pre-Win8.1/2012R2)
  - Enabled
    - Configure LanmanWorkstation Dependencies
      - Bowser
      - MrxSMB20
      - NSI

Figure 19. Disabling SMB v1 client extra settings via the "MS Security Guide" Group Policy template.

Setting	State	Comment
Configure SMB v1 server	Disabled	No
Configure SMB v1 client driver	Enabled	No
Configure SMB v1 client (extra setting needed for pre-Win8.1/2012R2)	Enabled	No

Figure 20.

Disabling SMB v1 client driver via the "MS Security Guide" Group Policy template—additional settings ensuring that the "MRxSmb10" option is not present.



Hardening Windows Remote Management (WinRM)

```
Tactic: Lateral dispersion between systems via Windows Remote Management (WinRM)
and PowerShell remoting
```

Manual operators may leverage Windows Remote Management (WinRM) to propagate ransomware throughout an environment. WinRM is enabled by default on all Windows Server operating systems (since Windows Server 2012 and above), but disabled on all client operating systems (Windows 7 and Windows 10) and older server platforms (Windows Server 2008 R2).

PowerShell Remoting (PS Remoting) is a native Windows remote command execution feature that's built on top of the WinRM protocol.

If WinRM has ever been enabled on a client (non-server) operating system, then the following configurations will exist on an endpoint, and will not be remediated solely through the PowerShell command noted in Figure 21.

- WinRM listener configured
- Windows Firewall exception configured

These items will need to be disabled manually through the commands in Figure 24 and Figure 25.

Figure 21.

PowerShell Command to disable WinRM / PowerShell Remoting on an endpoint.

PowerShell:

```
Disable-PSRemoting -Force
```

Note: Disabling PowerShell Remoting does not prevent local users from creating PowerShell sessions on the local computer - or for sessions destined for remote computers.

After running the command, the message recorded in Figure 22 will be displayed.

Figure 22. Warning message after disabling PSRemoting.

```
PS C:\WINDOWS\system32> Disable-PSRemoting -Force
WARNING: Disabling the session configurations does not undo all the changes made by the Enable-PSRemoting or
Enable-PSSessionConfiguration cmdlet. You might have to manually undo the changes by following these steps:
    1. Stop and disable the WinRM service.
    2. Delete the listener that accepts requests on any IP address.
    3. Disable the firewall exceptions for WS-Management communications.
    4. Restore the value of the LocalAccountTokenFilterPolicy to 0, which restricts remote access to members of the
Administrators group on the computer.
```

Figures 23-26 show how to enforce the additional steps for disabling WinRM via PowerShell.

Figure 23.

PowerShell command to stop and disable the WinRM Service.

Stop and disable the WinRM Service.

```
Stop-Service WinRM -PassThruSet-Service WinRM -StartupType Disabled
```

Figure 24.

PowerShell commands to delete a WSMAN listener.

Disable the listener that accepts requests on any IP address.

```
dir wsman:\localhost\listener
Remove-Item -Path WSMan:\Localhost\listener\<Listener name>
```

Figure 25.

PowerShell command to disable firewall exceptions for WinRM.

Disable the firewall exceptions for WS-Management communications.

```
Set-NetFirewallRule -DisplayName 'Windows Remote Management (HTTP-In)'
-Enabled False
```

Figure 26.

PowerShell command to configure the registry key for LocalAccount TokenFilterPolicy.

Restore the value of the LocalAccountTokenFilterPolicy to "0" (zero), which enforces UAC token filtering (admin approval mode) for the built-in administrator (RID 500) account.

```
Set-ItemProperty -Path
HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\policies\system -Name
LocalAccountTokenFilterPolicy -Value 0
```

Group Policy Method:

- Computer Configuration > Policies > Administrative Templates > Windows Components > Windows Remote Management (WinRM) > WinRM Service > Allow remote server management through WinRM

If the above Group Policy setting is configured as "Disabled", the WinRM service will not respond to requests from a remote computer, regardless of whether or not any WinRM listeners are configured.

- Computer Configuration > Policies > Administrative Templates > Windows Components > Windows Remote Shell > Allow Remote Shell Access

This Group Policy setting will manage the configuration of remote access for all supported shells to execute scripts and commands.

Credential Exposure and Usage Hardening

Remote Usage of Local Accounts

> Tactic: Lateral movement and propagation using the built-in local administrator account on endpoints

Local accounts that exist on endpoints are often a common avenue leveraged by attackers to laterally move throughout an environment. This tactic is especially impactful when the password for the built-in local administrator account is configured to the same value across multiple endpoints.

To mitigate the impact of local accounts being leveraged for lateral movement, Microsoft Security Advisory KB2871997[6] introduced two (2) well-known SIDs that can be leveraged within Group Policy settings to restrict the usage of local accounts for lateral movement.

- S-1-5-113: NT AUTHORITY\Local account
- S-1-5-114: NT AUTHORITY\Local account and member of Administrators group

Specifically, the SID "S-1-5-114: NT AUTHORITY\Local account and member of Administrators group" is added to an account's access token if the local account is a member of the BUILTIN\Administrators group. **This is the most beneficial SID to stop an attacker (or ransomware variant) that propagates using credentials for any local administrative accounts.**

Note: For SID "S-1-5-114: NT AUTHORITY\Local account and member of Administrators group", if Failover Clustering is utilized, this feature should leverage a non-administrative local account (CLIUSR) for cluster node management. If this account is a member of the local Administrators group on an endpoint that is part of a cluster, blocking the network logon permissions can cause cluster services to fail. Be cautious and thoroughly test this configuration on servers where Failover Clustering is utilized.

Step 1 – Option 1: S-1-5-114 SID

To mitigate the usage of local administrative accounts from being used for lateral movement, utilize the SID "S-1-5-114: NT AUTHORITY\Local account and member of Administrators group" within the following settings:

- Computer Configuration > Policies > Windows Settings > Security Settings > Local Policies > User Rights Assignment
  - Deny access to this computer from the network (SeDenyNetworkLogonRight)
  - Deny log on as a batch job (SeDenyBatchLogonRight)
  - Deny log on as a service (SeDenyServiceLogonRight)
  - Deny log on through Terminal Services (SeDenyRemoteInteractiveLogonRight)
  - Debug Programs (SeDebugPrivilege)—permission used for attempted privilege escalation and process injection

Step 1 – Option 2: UAC Token-Filtering

An additional control that can be enforced via Group Policy settings pertains to the usage of local accounts for remote administration and connectivity during a network logon. If the full scope of permissions (referenced in Option 1 above) cannot be implemented in a short timeframe, consider applying the UAC token-filtering method to local accounts for network-based logons.

These configurations can be enforced via the previously mentioned "Microsoft Security Guide" Group Policy template from the Microsoft Security Compliance Toolkit.[7]

Group Policy Setting:

- Computer Configuration > Policies > Administrative Templates > MS Security Guide > Apply UAC restrictions to local accounts on network logons

6 Microsoft (May 13, 2014). Microsoft Security Advisory: Update to improve credentials protection and management; May 13, 2014.
7 See https://www.microsoft.com/en-us/download/details.aspx?id=55319

Once enabled, the registry value (Fig. 27) will be configured on each endpoint:

Figure 27.

Registry key and value for enabling UAC restrictions for local accounts.

```
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\
LocalAccountTokenFilterPolicy
REG_DWORD = “0” (Enabled)
```

When set to “0”, remote connections with high integrity access tokens are only possible using either the plaintext credential or password hash of the RID 500 local administrator, dependent upon on the setting of “FilterAdministratorToken.”

The “FilterAdministratorToken” setting can either enable (1) or disable (0) (default) “Admin Approval” mode for the RID 500 local administrator. When enabled, the access token for the RID 500 local administrator account is filtered and therefore User Account Control (UAC) is enforced for this account (which can ultimately stop attempts to leverage this account for lateral movement across endpoints).

Group Policy Setting:

- Computer Configuration > Policies > Windows Settings > Security Settings > Local Policies > Security Options > User Account Control: Admin Approval Mode for the built-in Administrator account

Once enabled, the registry value (Fig. 28) will be configured on each endpoint:

Figure 28.

Registry key and value for requiring admin approval mode for local administrative accounts.

```
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\
FilterAdministratorToken
REG_DWORD = “1” (Enabled)
```

Note: It’s also prudent to ensure that the default setting for “User Account Control: Run all administrators in Admin Approval Mode” (“EnableLUA” option) is not changed from Enabled (Default) to Disabled. If this setting is disabled, all UAC policies are also disabled. With this setting disabled, it is possible to perform privileged remote authentication using plaintext credentials or password hashes with any local account that is a member of the local administrators group.

Group Policy Setting:

- Computer Configuration > Policies > Administrative Templates > MS Security Guide > User Account Control: Run all administrators in Admin Approval Mode

Once enabled, the registry value (Fig. 29) will be configured on each endpoint. This is the default setting.

Figure 29.

Registry key and value for enabling UAC restrictions for local accounts.

```
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\EnableLUA
REG_DWORD = “1” (Enabled)
```

UAC access token filtering will not affect any domain accounts in the local Administrators group on an endpoint.

Step 2: LAPS

Once the usage of local accounts has been blocked for remote authentication and access to remote endpoints, an organization must align a strategy to enforce password randomization for the built-in local administrator account. For many organizations, the easiest way to accomplish this task is by deploying and leveraging Microsoft Local Administrator Password Solutions (LAPS).[8]

Reduce the Exposure of Privileged and Service Accounts

```
Tactic: Lateral movement and propagation using domain-based accounts
```

Privileged Account Logon Restrictions

For ransomware to be deployed throughout an environment, privileged and service accounts credentials are commonly utilized for lateral movement and mass propagation. Until a thorough investigation has been completed, it may be difficult to determine the specific credentials that are being utilized by a ransomware variant for connectivity to a large scope of systems within an environment.

For any accounts that have privileged access throughout an environment, the accounts should not be utilized on standard workstations and laptops, but rather from designated systems (e.g., Privileged Access Workstations (PAWS)) that reside in restricted and protected VLANs and Tiers. Explicit privileged accounts should be defined for each Tier, and only utilized within the designated Tier.

The recommendations for restricting the scope of access for privileged accounts is based upon Microsoft's guidance for securing privileged access.[9]

As a quick containment measure, consider blocking any accounts with privileged access from being able to login (remotely or locally) to standard workstations, laptops, and common access servers (e.g., virtualized desktop infrastructure).

The settings referenced below are configurable via the Group Policy path of:

- Computer Configuration > Policies > Windows Settings > Security Settings > Local Policies > User Rights Assignment

Accounts delegated with local or domain privileged access should be explicitly denied access to standard workstations and laptop systems within the context of the following settings (which can be configured using Group Policy settings similar to what are depicted in Fig. 30):

- Deny access to this computer from the network (also include S-1-5-114: NT AUTHORITY\Local account and member of Administrators group)
- Deny log on as a batch job
- Deny log on as a service
- Deny log on locally
- Deny log on through Terminal Services

Figure 30.

Example of Privileged Account access restrictions for a standard workstation using Group Policy settings.

Local Policies/User Rights Assignment

Policy	Setting
Deny access to this computer from the network	Local account and member of Administrators group, MCWHIRT\Domain Admins, MCWHIRT\Enterprise Admins, MCWHIRT\Schema Admins, MCWHIRT\Tier0-DomainAdmins, MCWHIRT\Tier0-ExchangeAdmins, MCWHIRT\Tier1-ServerAdmins, MCWHIRT\Tier1-ServiceAccounts
Deny log on as a batch job	Local account and member of Administrators group, MCWHIRT\Domain Admins, MCWHIRT\Enterprise Admins, MCWHIRT\Schema Admins, MCWHIRT\Tier0-DomainAdmins, MCWHIRT\Tier0-ExchangeAdmins, MCWHIRT\Tier1-ServerAdmins, MCWHIRT\Tier1-ServiceAccounts
Deny log on as a service	Local account and member of Administrators group, MCWHIRT\Domain Admins, MCWHIRT\Enterprise Admins, MCWHIRT\Schema Admins, MCWHIRT\Tier0-DomainAdmins, MCWHIRT\Tier0-ExchangeAdmins, MCWHIRT\Tier1-ServerAdmins, MCWHIRT\Tier1-ServiceAccounts
Deny log on locally	MCWHIRT\Domain Admins, MCWHIRT\Enterprise Admins, MCWHIRT\Schema Admins, MCWHIRT\Tier0-DomainAdmins, MCWHIRT\Tier0-ExchangeAdmins, MCWHIRT\Tier1-ServerAdmins, MCWHIRT\Tier1-ServiceAccounts
Deny log on through Terminal Services	Local account and member of Administrators group, MCWHIRT\Domain Admins, MCWHIRT\Enterprise Admins, MCWHIRT\Schema Admins, MCWHIRT\Tier0-DomainAdmins, MCWHIRT\Tier0-ExchangeAdmins, MCWHIRT\Tier1-ServerAdmins, MCWHIRT\Tier1-ServiceAccounts

8 See https://www.microsoft.com/en-us/download/details.aspx?id=46899
9 Microsoft (February 13, 2019). Active Directory administrative tier model.

Service Account Logon Restrictions

Organizations should also consider enhancing the security of domain-based service accounts - to restrict the capability for the accounts to be used for interactive, remote desktop, and where possible, network-based logons.

On endpoints where the service account is not required for interactive or remote logon purposes, Group Policy settings can be used to enforce recommended logon restrictions for limiting the exposure of service accounts.

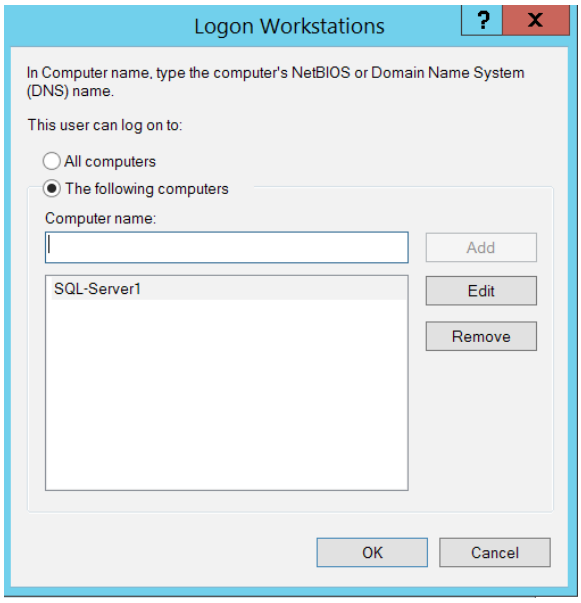
- Computer Configuration > Policies > Windows Settings > Security Settings > Local Policies > User Rights Assignment
  - Deny log on locally (SeDenyInteractiveLogonRight)
  - Deny log on through Terminal Services (SeDenyRemoteInteractiveLogonRight)

Additional recommended logon hardening for service accounts (on endpoints where the service accounts is not required for network-based logon purposes):

- Computer Configuration > Policies > Windows Settings > Security Settings > Local Policies > User Rights Assignment
  - Deny access to this computer from the network (SeDenyNetworkLogonRight)

If a service account is only required to be leveraged on a single endpoint to run a specific service, the service account can be further restricted to only permit the account's usage on a predefined listing of endpoints.

Figure 31. Option to restrict an account to logon to specific endpoints.



- Active Directory Users and Computers > Select the Account Tab
  - "Log On To" button > Select the proper scope of computers for access (Fig. 31)

Protected Users Security Group

By leveraging the "Protected Users" security group for privileged accounts, an organization can minimize various risk factors and common exploitation methods for exposing privileged accounts on endpoints.

Beginning with Microsoft Windows 8.1 and Microsoft Windows Server 2012 R2 (and above), the "Protected Users" security group was introduced to manage credential exposure within an environment. Members of this group automatically have specific protections applied to their accounts, including:

- The Kerberos ticket granting ticket (TGT) expires after 4 hours, rather than the normal 10-hour default setting.
- No NTLM hash for an account is stored in LSASS since only Kerberos authentication is used (NTLM authentication is disabled for an account).
- Cached credentials are blocked. A Domain Controller must be available to authenticate the account.
- WDigest authentication is disabled for an account, regardless of an endpoint's applied policy settings.
- DES and RC4 can't be used for Kerberos pre-authentication (Server 2012 R2 or higher); rather Kerberos with AES encryption will be enforced.
- Accounts cannot be used for either constrained or unconstrained delegation (equivalent to enforcing the "Account is sensitive and cannot be delegated" setting in Active Directory Users and Computers).

To provide Domain Controller-side restrictions for members of the "Protected Users" security group, the domain functional level must be Windows Server 2012 R2 (or higher). Microsoft Security Advisory KB2871997[10] adds support for the protections enforced for members of the "Protected Users" security group to Windows 7, Windows Server 2008 R2, and Windows Server 2012 systems.

Note: Service accounts (including Managed Service Accounts) should NOT be added to the "Protected Users" security group — as authentication will fail.

10 Microsoft (May 13, 2014). Microsoft Security Advisory: Update to improve credentials protection and management; May 13, 2014.

Cleartext Password Protections

```
Tactic: Obtaining cleartext credentials in memory for credential harvesting
```

In addition to restricting access for privileged accounts, controls should be enforced that minimize the exposure of credentials and tokens in memory on endpoints.

On older Windows Operating Systems, cleartext passwords are stored in memory (LSASS) to primarily support WDigest authentication. WDigest should be explicitly disabled on all Windows endpoints where it is not disabled by default.

By default, WDigest authentication is disabled in Windows 8.1+ and in Windows Server 2012 R2+.

Beginning with Windows 7 and Windows Server 2008 R2, after installing Microsoft Security Advisory KB2871997,[11] WDigest authentication can be configured either by modifying the registry or by using the "Microsoft Security Guide" Group Policy template from the Microsoft Security Compliance Toolkit.[12]

Figure 32.

Registry key and value for disabling WDigest authentication.

```
HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\WDigest\
UseLogonCredential
REG_DWORD = "0"
```

Registry Method:

Another registry setting that should be explicitly configured is the "TokenLeakDetectDelaySecs" setting (Fig. 33), which will clear credentials in memory of logged off users after 30 seconds, mimicking the behavior of Windows 8.1 and above.

Figure 33.

Registry key and value for enforcing the "TokenLeakDetect DelaySecs" setting.

```
HKLM\SYSTEM\CurrentControlSet\Control\Lsa\TokenLeakDetectDelaySecs
REG_DWORD = "30"
```

Group Policy Method:

Using the "Microsoft Security Guide" Group Policy template, WDigest authentication can be disabled via a Group Policy setting (Fig. 34).

- Computer Configuration > Policies > Administrative Templates > MS Security Guide > WDigest Authentication

Figure 34. Disabling WDigest authentication via the "MS Security Guide" Group Policy template.

Setting	State	Comment
Configure SMB v1 server	Not configured	No
Configure SMB v1 client driver	Not configured	No
Configure SMB v1 client (extra setting needed for pre-Win8.1/2012R2)	Not configured	No
Extended Protection for LDAP Authentication (Domain Controllers only)	Not configured	No
Turn on Windows Defender protection against Potentially Unwanted Applications (DEPRECATED)	Not configured	No
Enable Structured Exception Handling Overwrite Protection (SEHOP)	Not configured	No
Apply UAC restrictions to local accounts on network logons	Not configured	No
WDigest Authentication (disabling may require KB2871997)	Disabled	No
Lsass.exe audit mode	Not configured	No
LSA Protection	Not configured	No
Remove "Run As Different User" from context menus	Not configured	No
Block Flash activation in Office documents	Not configured	No

11 Microsoft (May 13, 2014). Microsoft Security Advisory: Update to improve credentials protection and management; May 13, 2014.
12 See https://www.microsoft.com/en-us/download/details.aspx?id=55319

Additionally, an organization should verify if any applications are explicitly listed in the "Allow" keys (Fig. 35) - as this would permit the tspkgs / CredSSP providers to store cleartext passwords in memory.

Figure 35. Additional registry key for hardening against cleartext password storage.

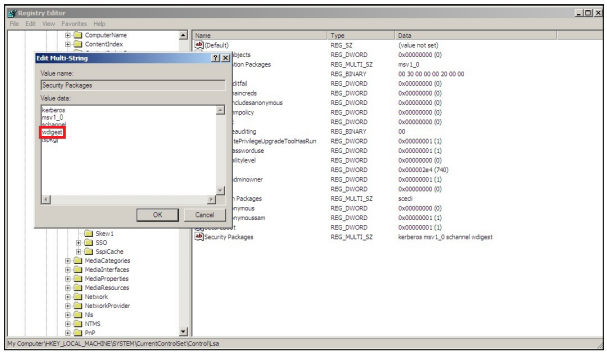
```
HKLM\SYSTEM\CurrentControlSet\Control\Lsa\Credssp\PolicyDefaults
```

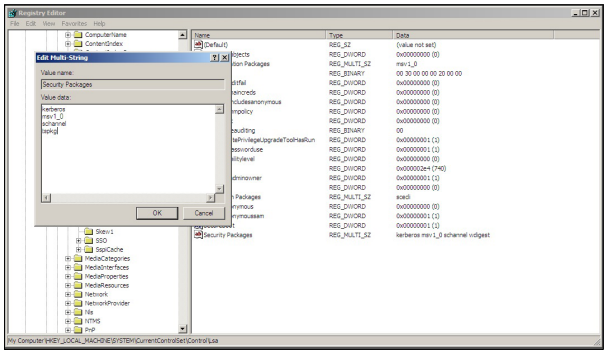
As Microsoft Security Advisory KB2871997[13] is not applicable for Windows XP, Windows Server 2003, and Windows Server 2008, to disable WDigest authentication on these platforms, prior to a system reboot, WDigest needs to be removed from the listing of LSA security packages within the registry (Fig. 36 and Fig. 37).

Figure 36. Registry key to modify LSA security packages.

```
HKLM\System\CurrentControlSet\Control\Lsa\Security Packages
```

Figure 37. LSA security package registry key before and after the removal of WDigest authentication from the listing of providers.





By default, Group Policy settings are only reprocessed and reapplied if the actual Group Policy was modified prior to the default refresh interval.

Many attackers will manually "enable" WDigest authentication on endpoints by directly modifying the registry (UseLogonCredential configured to a value of "1"). Even on endpoints where WDigest authentication is automatically disabled by default, it is recommended to enforce the Group Policy settings noted in Figure 33—and configure automatic policy reprocessing for the configured settings on an automated basis.

- Computer Configuration > Policies > Administrative Templates > System > Group Policy > Configure security policy processing
  - Enabled - Process even if the GPOs have not changed
- Computer Configuration > Policies > Administrative Templates > System > Group Policy > Configure registry policy processing
  - Enabled - Process even if the GPOs have not changed

13 Microsoft (May 13, 2014). Microsoft Security Advisory: Update to improve credentials protection and management; May 13, 2014.

Domain Controller Isolation and Recovery Planning

In the event of a ransomware outbreak, an organization must ensure that they have a practiced plan in place to quickly isolate key systems, and ensure that at least one Domain Controller can be quickly taken offline and safely isolated for each domain within managed and trusted forests. If the disk partition that houses the Active Directory database (%SYSTEMROOT%\ntds\ntds.dit) and SYSVOL (%SYSTEMROOT%\SYSVOL) on all Domain Controllers were to be encrypted, this will impact the availability of domain services and functionality for all domain-based applications and services, including authentication, name resolution, and GPO processing. If Domain Controller backups are either encrypted or are not current, an organization may be faced with a complete rebuild of an entire forest, which can further impact downtime.

When Mandiant is engaged to help contain an active ransomware deployment, the first steps recommended that an organization take are to isolate at least one Domain Controller (preferable one that holds FSMO roles) and ensure that offline backups of SYSVOL (%SYSTEMROOT%\SYSVOL*) and GPOs are available and current.

Figure 38. Command to determine a Domain Controller that holds a FSMO role.

```
netdom query fsmo
```

Figure 39. PowerShell command to backup all GPOs in a domain.

```
backup-gpo -domain “domain.local” -all -path “c:\temp\gpo-backups”
```

Proactively, in the event that either an authoritative or non-authoritative Domain Controller restoration is required, an organization should ensure that the Directory Services Restore Mode (DSRM) password is set to a known value on all Domain Controllers. If an organization does not have the DSRM password available, the password can be set to a known value by following the process outlined in the figure below. The steps will need to be initiated on each Domain Controller.

Figure 40.

Command to set the DSRM password on a Domain Controller.

```
PS C:\Windows\system32> ntdsutil
C:\Windows\System32\ntdsutil.exe: set drsm password
Reset DRSM Administrator Password: reset password on server null
Please type password for DS Restore Mode Administrator Account: ****************
Please confirm new password: **************
Password has been set successfully.

Reset DRSM Administrator Password: q
C:\Windows\System32\ntdsutil.exe:
```

When restoring Active Directory from previous Domain Controller backups is the only viable option to restore domain services, an organization must first ensure that they have a working and tested backup plan and strategy to guarantee the availability and integrity of the schema and domain services that need to be reconstituted. The following best practices should be proactively reviewed by an organization:

- **Offline backups:** ensure that offline Domain Controller backups are secured and stored separately from online backups.
- **Encryption:** backup data should be encrypted both during transit (over the wire) and when at rest or mirrored for offsite storage.
- **Configure alerting for backup operations:** backup products and technologies should be configured to detect and provide alerting for operations that are critical to the availability and integrity of backup data (e.g., deletion of backup data, purging of backup metadata, restoration events, media errors).
- **Data Retention:** backup products and technologies should ensure that backups are retained for a pre-defined period-of-time - before overwriting or purging data.
- **Enforce role-based access control:** Access to backup media and the applications that govern and manage data backups should utilize role-based access controls, to restrict the scope of accounts that have access to the stored data and configuration parameters.
- **Testing and verification:** An organization should periodically test and verify that data can be restored and reconstituted from both online and offline media sources. Both authoritative and non-authoritative Domain Controller restoration processes should be documented and tested.

Group Policy Object (GPO) Permissions and Monitoring

A common tactic utilized by ransomware operators is to deploy encryptors by modifying an existing GPO configuration, or by creating a new GPO, and linking either at the root of the domain or to a large scope of Organizational Units (OUs) that contain computer objects. By leveraging scheduled tasks, startup / logon scripts, or software installation package settings within GPOs, ransomware operators are able to leverage native functionality within Active Directory to accomplish their mission, without the need to directly interface with each endpoint to invoke encryptors across an enterprise environment.

Proactively, organizations should review the scope of configured GPOs, and the last modified timestamp of a GPO to ensure that all modifications align to authorized and expected activities.

Figure 41. PowerShell command to review the scope of configured GPOs – including the last modified timestamp

```
get-gpo -all | export-csv -path "c:\temp\gpo-listing-all.csv" -NoTypeInformation
```

Additionally, organizations should review permissions for existing GPOs—specifically focusing on the scope of accounts and groups that have the ability to modify GPOs within a domain. Any accounts or security groups that have the ability to modify a large scope of GPOs, or GPOs that are linked-to and enforce security settings for a large scope of endpoints (e.g., Default Domain Policy) should be carefully protected, and deemed to be privileged within a domain.

Figure 42. PowerShell commands to list existing GPOs and assigned permissions.

```
$permissions = Foreach ($GPO in (Get-GPO -All | Where {$_.DisplayName -like "*"}))
    {
        Foreach ($Permission in (Get-GPPermissions $GPO.DisplayName -All | Where {$_.Permission
-like "*"}))
        {
          New-Object PSObject -property @{GPO=$GPO.DisplayName;Trustee=$Permission.Trustee.
Name;Permission=$Permission.Permission}
        }
    }
$permissions | Select GPO,Trustee,Permission | Export-CSV c:\temp\GPO-Permissions.csv
-NoTypeInformation
```

GPO modifications can be proactively detected by reviewing Security event logs on Domain Controllers for Event ID 5136 - which requires that "Audit Directory Service Changes" auditing be enabled. Figure 6 provides an example of a Security event log detection for the Default Domain Policy GPO (well-known GUID of 31B2F340-016D-11D2-945F00C04FB984F9) being modified, and a Scheduled Task (client side extension of AADCED64-746C-4633-A97C-D61349046527) being added.

Figure 43. Event ID 5136 detection for GPO modifications.

Conclusion

Ransomware poses a serious threat to organizations, as attackers continue to utilize this tactic to monetize breaches. This whitepaper provided practical guidance on protecting against ransomware attacks and containing ongoing ransomware events. This whitepaper should not be considered a comprehensive guide on every tactic and control that can be used for this purpose, but it can serve as a valuable resource for organizations faced with this challenge. It is based on years of experience of helping our clients protect against and recover from ransomware attacks—and it can help your organization do the same.

To learn more about FireEye, visit: **www.FireEye.com/mandiant**

FireEye, Inc.
601 McCarthy Blvd. Milpitas, CA 95035
408.321.6300/877.FIREEYE (347.3393)
info@FireEye.com

About Mandiant Solutions
Mandiant Solutions brings together the world's leading threat intelligence and frontline expertise with continuous security validation to arm organizations with the tools needed to increase security effectiveness and reduce business risk.

MANDIANT®