REPORT
WinRM Secure Configuration and Validation

v1.0.0

Author:

Eldon Gabriel

October 15, 2025

TABLE OF CONTENTS

Disclaimer: This report documents my personal work completing an MCSI lab exercise. It reflects my independent understanding and configuration of Windows 10 Local Group Policy settings in a controlled environment. No MCSI video content, lab materials, or proprietary instructions have been shared or distributed. All information presented follows MCSI's disclosure and academic integrity policies.

REVISION HISTORY

Version	Date	Author	Description of Changes
v1.0.0	10/15/2025	Eldon G.	Initial draft.

1.0 WINRM SECURE CONFIGURATION PROJECT

1.1 Project Description

This project explains how I set up and checked the security of Windows Remote Management (WinRM) on a Windows 10 Virtual Machine (VM). The goal was to ensure that the system adhered to secure remote management protocols. This means using encryption, turning off unsafe login methods, and ensuring that Group Policy rules are set for the WinRM service and client settings.

1.2 System Preparation

- The **WinRM service** was verified to be stopped.
- The WinRM service was started manually using the `Start-Service` command with the `winrm` parameter.
- Executed `winrm quickconfig` to initialize the WinRM setup.
- The network profile was adjusted to **private** using the PowerShell cmdlet. `Set-NetConnectionProfile -Name "Network 2" -NetworkCategory Private`
- Ran `winrm quickconfig` to enable firewall exceptions and local admin rights for remote management.

1.3 WinRM Listener Configuration Validation

The listener was validated using the following:

```powershell
winrm enumerate winrm/config/listener
```

Result:

- Transport: HTTP
- Port: 5985
- Enabled: True
- ListeningOn: 127.0.0.1, 192.168.1.x (Private network)

This confirms that WinRM is configured to securely accept inbound management connections.

1.4 WinRM Configuration Inspection

Executed:

```powershell
winrm get winrm/config
```

Key Observations:

- `AllowUnencrypted = false [Source="GPO"]`
- `Basic = false [Source="GPO"]`
- `Digest = false [Source="GPO"]`
- `Kerberos = true`
- `Negotiate = true`
- `Certificate = true`
- `AllowRemoteAccess = true`
- Listener Port: 5985
- `[Source="GPO"]` indicates the Group Policy enforcement of secure configuration.

These values confirm that Basic and Digest authentications are disabled, while Kerberos and Negotiate methods are active.

1.5 Additional Verification

- Confirm that the WinRM firewall exception is enabled for the private network profile.
- Verified that `LocalAccountTokenFilterPolicy` was set correctly by `winrm quickconfig` to allow remote administrative access.
- Checked the WinRM service type configuration to **Delayed Auto Start** to ensure persistence after reboot.

2.0 CONCLUSION

2.1 Key Takeaways

- WinRM was successfully configured for secure, remote management.
- Group Policy control was confirmed as an authoritative source for configuration enforcement.
- The system meets security standards by using secure authentication and disabling unencrypted traffic.

2.2 Security Implications and Recommendations

- **Enforce via GPO**: Maintain centralized Group Policy enforcement to prevent configuration drifts.
- **Enable HTTPS Listener**: Implement a valid server certificate and enable the HTTPS listener on port 5986 for encrypted communication.
- **Audit Regularly**: Use PowerShell auditing and compliance scripts to verify WinRM configurations across managed endpoints.
- **Limit TrustedHosts**: Restrict `TrustedHosts` to known management servers to reduce the risk of lateral movement.
