

STIG Implementation Report

- **Intern Credit Application For:** Bruce Thornton
Date: 11/10/2025
STIG Finding: WN11-CC-000020
 - **SRG:** [SRG-OS-000480-GPOS-00227](#)
Severity: medium
Vulnerability ID: V-253353 **CCI:** CCI-000366
-

1. Introduction

This report documents the process of identifying, remediating, and verifying the fix for a Windows 11 STIG compliance finding. The selected finding was: STIG ID: WN11-CC-000020 "IPv6 source routing must be configured to highest protection."

2. Initial Scan Results

- Tool: Tenable.sc / Nessus (Windows 11 STIG Audit Policy)
- Finding ID: WN11-CC-000020
- Status: **Failed** (non-compliant)

 **Evidence:** First identified the STIG:

<https://stigaview.com/products/win11/v1r6/WN11-CC-000020/>

Initial scan result:

The screenshot displays the Tenable Vulnerability Management interface. At the top, the header shows 'tenable Vulnerability Management' and 'Scans > Scan Details'. A search bar contains 'WN11-CC-000020' with '1 Results' indicated. Below the search bar, there are three progress bars: 'Failed' (1), 'Warning' (1), and 'Passed' (1). A table lists the scan results:

STATUS	NAME	FAMILY	COUNT
Failed	WN11-CC-000020 - IPv6 source routing must be configured to hi...	Windows Compliance Checks	1

On the right side, a 'Scan Details' panel provides additional information:

- CRITICAL VULNERABILITIES:** 0
- HIGH VULNERABILITIES:** 0
- MEDIUM VULNERABILITIES:** 0
- LOW VULNERABILITIES:** 0
- STATUS:** Completed
- START TIME:** 11/10/2025 at 11:22 AM
- TEMPLATE:** Advanced Network Scan
- SCANNER:** LOCAL-SCAN-ENGINE-01
- TARGETS:** 10.1.0.152

3. Manual Remediation Steps

*IMPORTANT

For WN11-CC-000020, the required control is implemented by configuring the registry value

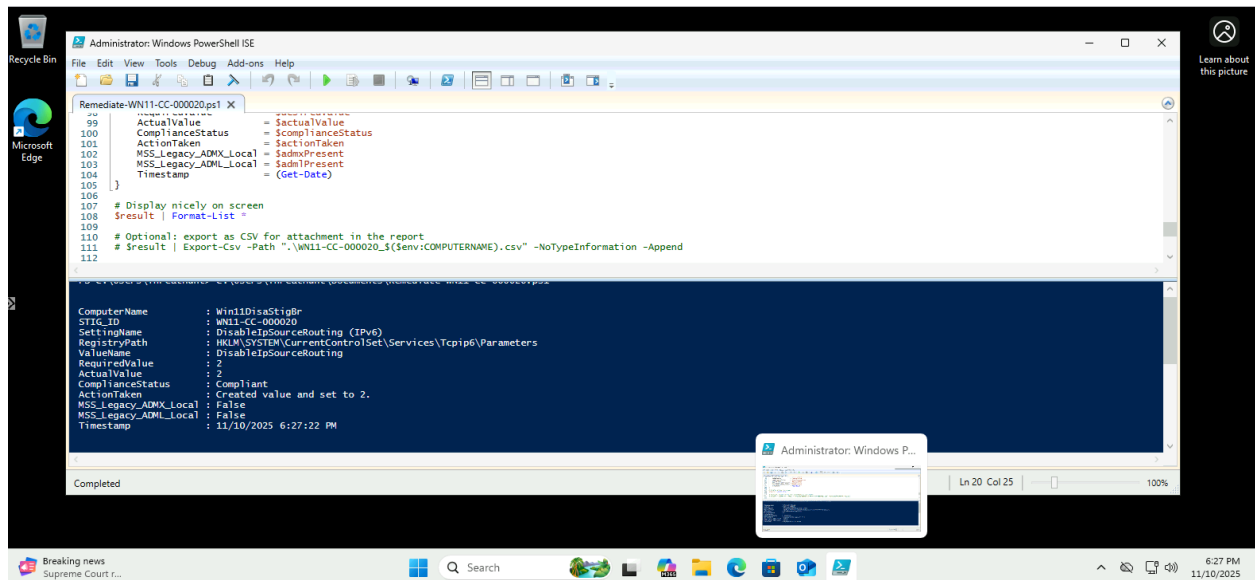
HKLM\SYSTEM\CurrentControlSet\Services\Tcpip6\Parameters\DisableIpSourceRouting (REG_DWORD) to 2, which represents 'Highest protection, source routing is completely disabled.'

On the assessed Windows 11 VM, this value was created/updated manually via Registry Editor and verified via PowerShell.

The MSS-Legacy GPO template is not present on this standalone VM; however, the registry setting reflects the required configuration specified in the STIG check text.”

4. Remediation with PowerShell Script

Save as: Remediate-WN11-CC-000020.ps1 and run as **Administrator** utilizing PowerShell ISE:



Script:

```
<#
```

STIG ID : WN11-CC-000020

Title : IPv6 source routing must be configured to highest protection.

Policy text note:

This policy setting requires the installation of the MSS-Legacy custom templates included with the STIG package. "MSS-Legacy.admx" and "MSS-Legacy.adml" must be copied to the \Windows\PolicyDefinitions and \Windows\PolicyDefinitions\en-US directories respectively.

Technical implementation:

At the host level, the effective control is the following registry value:

HKLM\SYSTEM\CurrentControlSet\Services\Tcpip6\Parameters

Value Name : DisableIpSourceRouting

Type : REG_DWORD

Required : 2 (Highest protection)

This script directly configures and verifies that registry value. In a domain environment, Group Policy with MSS-Legacy templates should be used for long-term enforcement, but this script accurately reflects the host's compliance state.

#>

=====

Configuration

=====

\$regPath = 'HKLM:\SYSTEM\CurrentControlSet\Services\Tcpip6\Parameters'

\$valueName = 'DisableIpSourceRouting'

\$desiredValue = 2

\$stigId = 'WN11-CC-000020'

=====

Optional: Note MSS-Legacy local template presence

(Domain central store templates may not appear here, so this is informational only.)

=====

\$localAdmx = Join-Path \$env:WINDIR 'PolicyDefinitions\MSS-Legacy.admx'

\$localAdml = Join-Path \$env:WINDIR 'PolicyDefinitions\en-US\MSS-Legacy.adml'

\$admxPresent = Test-Path \$localAdmx

\$admlPresent = Test-Path \$localAdml

=====

Helper: Ensure elevated

=====

\$currId = [Security.Principal.WindowsIdentity]::GetCurrent()

\$principal = New-Object Security.Principal.WindowsPrincipal(\$currId)

\$adminRole = [Security.Principal.WindowsBuiltInRole]::Administrator

if (-not \$principal.IsInRole(\$adminRole)) {

Write-Warning "This script should be run from an elevated PowerShell session (Run as Administrator)."

}

=====

Remediation

=====

Ensure the key exists (normally present, but this is defensive)

if (-not (Test-Path \$regPath)) {

```

    New-Item -Path $regPath -Force | Out-Null
}

# Get current value (if it exists)
$current = Get-ItemProperty -Path $regPath -Name $valueName -ErrorAction
SilentlyContinue

if ($null -eq $current) {
    # Value doesn't exist - create it
    New-ItemProperty -Path $regPath -Name $valueName -Value $desiredValue
-PropertyType DWord -Force | Out-Null
    $actionTaken = "Created value and set to $desiredValue."
}
elseif ($current.$valueName -ne $desiredValue) {
    # Value exists but is incorrect - fix it
    Set-ItemProperty -Path $regPath -Name $valueName -Value $desiredValue -Type
DWord
    $actionTaken = "Updated value from $($current.$valueName) to $desiredValue."
}
else {
    $actionTaken = "No change required; value already set to $desiredValue."
}

# =====
# Verification
# =====
$verified = Get-ItemProperty -Path $regPath -Name $valueName -ErrorAction
SilentlyContinue
$actualValue = $verified.$valueName

if ($actualValue -eq $desiredValue) {
    $complianceStatus = 'Compliant'
} else {
    $complianceStatus = 'Non-Compliant'
}

# =====
# Report Output
# =====
$result = [pscustomobject]@{

```

```

ComputerName      = $env:COMPUTERNAME
STIG_ID           = $stigid
SettingName       = 'DisableIpSourceRouting (IPv6)'
RegistryPath      =
'HKLM\SYSTEM\CurrentControlSet\Services\Tcpip6\Parameters'
ValueName         = $valueName
RequiredValue     = $desiredValue
ActualValue       = $actualValue
ComplianceStatus  = $complianceStatus
ActionTaken       = $actionTaken
MSS_Legacy_ADMX_Local = $admxPresent
MSS_Legacy_ADML_Local = $admlPresent
Timestamp         = (Get-Date)
}

# Display nicely on screen
$result | Format-List *

# Optional: export as CSV for attachment in the report
# $result | Export-Csv -Path ".\WN11-CC-000020_$(($env:COMPUTERNAME).csv"
-NoTypeInfoInformation -Append

```

Run “gpupdate /force” and restart.

Scan again,

- Tool: Tenable.sc / Nessus (Windows 11 STIG Audit Policy)
- Finding ID: WN11-CC-000020
- Status: **Passed**

Evidence:

The screenshot displays the Tenable Vulnerability Management interface. The top navigation bar includes the Tenable logo, 'Vulnerability Management', and a breadcrumb trail 'Scans > Scan Details'. A 'Quick Actions' dropdown menu is visible on the right. The main header shows the scan name 'Win11DisaStigBruce' and 'VULNERABILITY MANAGEMENT SCANS'. Below this, there are tabs for 'Vulns by Plugin', 'Audits', 'Vulns by Asset', and 'History'. A search bar contains the identifier 'WN11-CC-000020' with '1 Results' indicated. A summary bar shows '0 Failed', '0 Warning', and '1 Passed'. The main table lists one item with the following details:

STATUS	NAME	FAMILY	COUNT
Passed	WN11-CC-000020 - IPv6 source routing must be configured to hi...	Windows Compliance Checks	1

On the right side, a 'Scan Details' panel provides additional information:

- CRITICAL VULNERABILITIES:** 0
- HIGH VULNERABILITIES:** 0
- MEDIUM VULNERABILITIES:** 0
- LOW VULNERABILITIES:** 0
- STATUS:** Completed
- START TIME:** 11/10/2025 at 12:34 PM
- TEMPLATE:** Advanced Network Scan
- SCANNER:** LOCAL-SCAN-ENGINE-01
- TARGETS:** 10.1.0.152

6. Conclusion:

The finding **WN11-CC-000020** was:

- Detected in the initial Tenable Windows 11 DISA STIG audit scan,
- Remediated by configuring the required registry value via **PowerShell** (without installing the MSS-Legacy ADMX/ADML templates), and
- Successfully **validated as compliant** in a subsequent Tenable scan, which reported the check as **Passed**.

I have demonstrated that Windows STIG control **WN11-CC-000020** can be effectively implemented and verified using direct registry configuration through PowerShell, even in environments where the MSS-Legacy policy templates are not present on the system.