

# ConnectWise Control Comprehensive Security Best Practice Guide

This guide was created to help Partners with an instance of ConnectWise Control properly lock down host systems in a manner to offer better protection from a security incident. The guide itself is broken into three elements: Operating System, Network and Application. Each of these areas should be reviewed and implemented.

Please note this document will be updated frequently. Ensure you have the most up to date copy.

This guide addresses the following:

- Microsoft Windows Server 2016 & 2019
- Microsoft IIS 10.x
- ConnectWise Control

This guide serves as an enhancement (or addition to) the ConnectWise Control Security Guide:

[https://docs.connectwise.com/ConnectWise\\_Control\\_Documentation/Get\\_started/Security\\_guide](https://docs.connectwise.com/ConnectWise_Control_Documentation/Get_started/Security_guide)

The linked ConnectWise Control Security Guide contains steps to configure and secure the following:

- Securing Session Traffic
- Cloud Administrator Lockout
- Security Configurations
  - Controlling User Permissions
    - Restrict a host to access a single remote machine
    - Restrict access to remote machines by organization
  - Two-Factor Authentication
  - Configuring SSL
    - Cloud Instances
    - On-Premises
- Configuring access to Your ConnectWise Control Server
  - Blocking and Restricting Access to Your ConnectWise Control Site
  - Automatically Force a Host to Disconnect from a Session
- User Authentication Options
  - Internal Authentication
  - Windows active Directory & LDAP
  - External User Authentication
- Logging and Auditing
  - Video or "Extended" Auditing
  - Login Auditing
- Revoke User Access
- Recommended Extensions for Security
  - Security Toolkit

- Report Manager or Reports Page
- Guest Security
  - Exiting a Support Session
  - Consent to Control

## Partner On premises and cloud installations

This section is intended for partners that have installed ConnectWise Control on an on-premises server or in on a self-hosted cloud virtual private server (VPS). For ConnectWise Hosted Control partners, this guidance in this section is just for your information, since you are unable to make these changes at the server level. All the actions below have been taken in your instance on the hosted platform.

### Operating System Hardening Guidelines (Before application install)

Review the Security Technical Implementation Guides (STIGs) as a methodology to secure Microsoft Server 2016 and 2019.

For AWS cloud instances, many of the High and Medium standards are addressed inside the AWS Standard Server AMI.

The user account and STIGs information below are strongly recommended for the ConnectWise Control server.

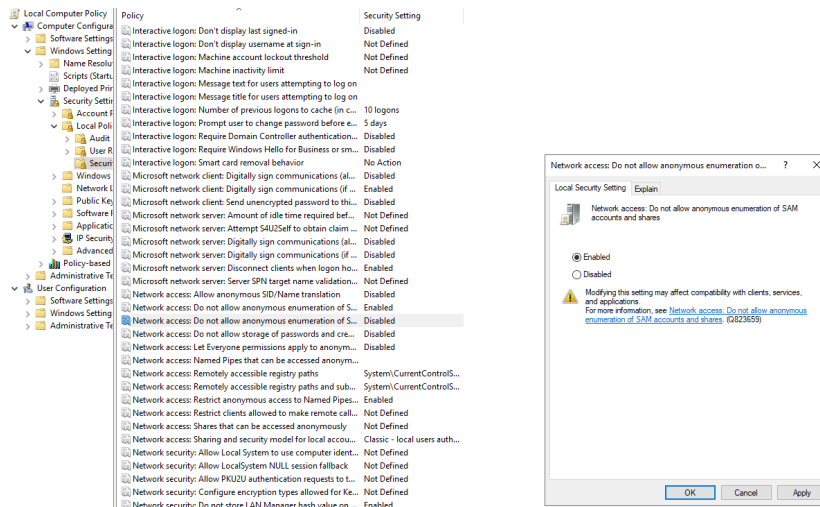
The IT Nation Secure team is recommending Partners implement the STIGs located here:

- Server 2016 - [https://www.stigviewer.com/stig/windows\\_server\\_2016/](https://www.stigviewer.com/stig/windows_server_2016/)
- Server 2019 - [https://www.stigviewer.com/stig/windows\\_server\\_2019/](https://www.stigviewer.com/stig/windows_server_2019/)
- IIS 10 - [https://www.stigviewer.com/stig/microsoft\\_iis\\_10.0\\_server/](https://www.stigviewer.com/stig/microsoft_iis_10.0_server/)

#### *STIG Items to Modify*

Run Microsoft Group Policy Editor from the "gpedit.msc" command. Group Policy Editor controls a wide range of options and can be used to enforce settings and change the defaults for applicable users and services.

1. *Disable Anonymous Network Access.*  
Do not allow anonymous enumeration of SAM accounts and shares.  
Configure the policy value for **Computer Configuration > Windows Settings > Security Settings > Security Options > Network access: Do not allow anonymous enumeration of SAM accounts and shares** to **Enabled**.



2. *Disallow Autoplay for non-volume devices.*

Configure the policy value for **Computer Configuration > Administrative Templates > Windows Components > AutoPlay Policies > Disallow Autoplay for non-volume devices** to **Enabled**

- For Server 2016: **Computer Configuration > Windows Settings > Security Settings > Local Policies > Security Options**

3. *Set the default behavior for AutoRun.*

Configure the policy value for **Computer Configuration > Administrative Templates > Windows Components > AutoPlay Policies > Set the default behavior for AutoRun** to **Enabled** with **Do not execute any autorun commands** selected.

4. *Turn off AutoPlay.*

Configure the policy value for **Computer Configuration > Administrative Templates > Windows Components > AutoPlay Policies > Turn off AutoPlay** to **Enabled** with **All Drives** selected.

The above setting is discussed in some detail within the Certify Fundamentals course available from the ConnectWise University.

5. *Ensure NO ONE is added to **Act as part of the operating system** in the GPO.*

Navigate to **Local Computer Policy > Computer Configuration > Windows Settings > Security Settings > Local Policies > User Rights Assignment.**

If any accounts or groups (to include administrators), are granted the **Act as part of the operating system** user right, the accounts should be removed immediately from this policy object.

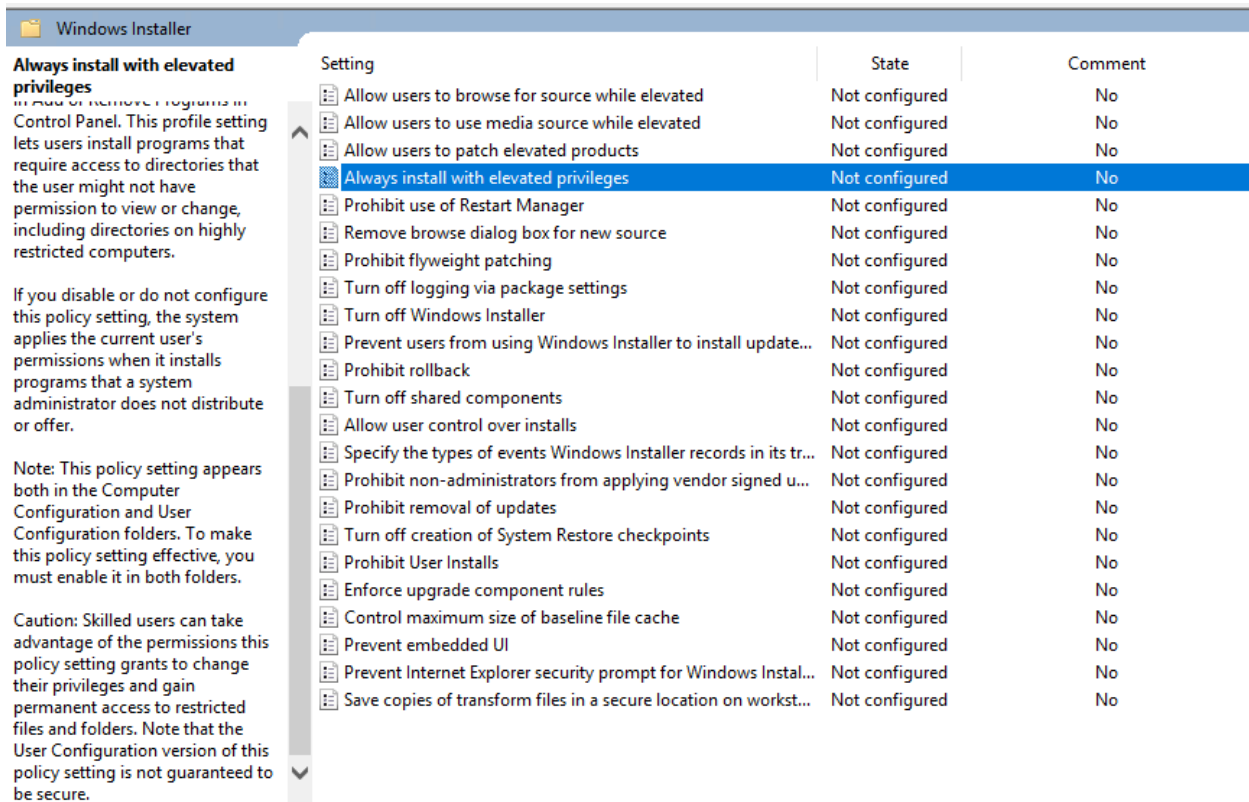
6. *Disable **Always install with elevated privileges**.*

Configure the policy value for **Computer Configuration > Administrative Templates > Windows Components > Windows Installer > Always install with elevated privileges** to **Disabled**.

This setting appears both in the **Computer Configuration** and **User Configuration** folders. To make this setting effective, it must be configured within both folders.

The “Not Configured” setting will use the user’s current permission set. This is part of the reason why having TWO accounts (a normal USER and a separate Privileged account) is very important.

Please also note the “Caution” item in the graphic below noting that skilled users can take advantage of the permissions this setting grants in order to change their privileges and gain permanent access to restricted files and folders.



Setting	State	Comment
Allow users to browse for source while elevated	Not configured	No
Allow users to use media source while elevated	Not configured	No
Allow users to patch elevated products	Not configured	No
<b>Always install with elevated privileges</b>	<b>Not configured</b>	<b>No</b>
Prohibit use of Restart Manager	Not configured	No
Remove browse dialog box for new source	Not configured	No
Prohibit flyweight patching	Not configured	No
Turn off logging via package settings	Not configured	No
Turn off Windows Installer	Not configured	No
Prevent users from using Windows Installer to install update...	Not configured	No
Prohibit rollback	Not configured	No
Turn off shared components	Not configured	No
Allow user control over installs	Not configured	No
Specify the types of events Windows Installer records in its tr...	Not configured	No
Prohibit non-administrators from applying vendor signed u...	Not configured	No
Prohibit removal of updates	Not configured	No
Turn off creation of System Restore checkpoints	Not configured	No
Prohibit User Installs	Not configured	No
Enforce upgrade component rules	Not configured	No
Control maximum size of baseline file cache	Not configured	No
Prevent embedded UI	Not configured	No
Prevent Internet Explorer security prompt for Windows Instal...	Not configured	No
Save copies of transform files in a secure location on workst...	Not configured	No

**Always install with elevated privileges**  
 In addition to removing programs in Control Panel, this profile setting lets users install programs that require access to directories that the user might not have permission to view or change, including directories on highly restricted computers.

If you disable or do not configure this policy setting, the system applies the current user's permissions when it installs programs that a system administrator does not distribute or offer.

Note: This policy setting appears both in the Computer Configuration and User Configuration folders. To make this policy setting effective, you must enable it in both folders.

Caution: Skilled users can take advantage of the permissions this policy setting grants to change their privileges and gain permanent access to restricted files and folders. Note that the User Configuration version of this policy setting is not guaranteed to be secure.

7. *Do not use administrative accounts with applications that access the Internet*

Microsoft Windows Server administrative accounts must not be used with applications that access the Internet, such as web browsers, or with potential Internet sources, such as email.

Using applications that access the Internet or have potential Internet sources using administrative privileges exposes a system to compromise. If a flaw in

an application is exploited while running as a privileged user, the entire system could be compromised. Web browsers and email are common attack vectors for introducing malicious code and must not be run with an administrative account. Since administrative accounts may generally change or work around technical restrictions for running a web browser or other applications, it is essential that policy require administrative accounts to not access the Internet or use applications such as email. The policy should define specific exceptions for local service administration. These exceptions may include HTTP(S)-based tools that are used for the administration of the local system, services, or attached devices. Whitelisting can be used to enforce the policy to ensure compliance.

### *Disable Server Headers in Microsoft IIS*

Web applications may unintentionally disclose information about their underlying technologies through headers, error messages, version numbers, or other identifying information. An attacker can use that information to research vulnerabilities in those technologies to attack the application to breach the system.

1. Open PowerShell as Administrator:
2. To validate that IIS Server Headers are not disabled, copy and paste:

```
Get-WebConfigurationProperty -pspath machine/webroot/apphost -filter 'system.webserver/security/requestfiltering' -name 'removeServerHeader'
```

3. Use the following PowerShell command to disable server headers in IIS10:

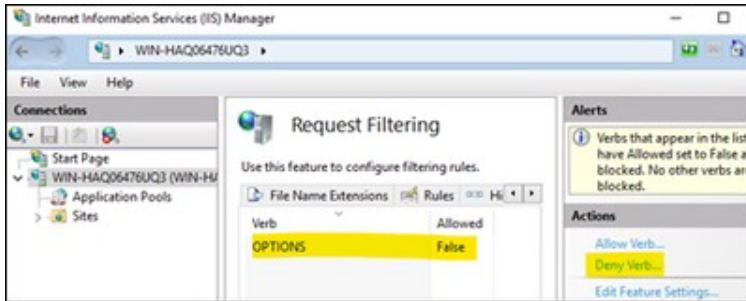
```
Set-WebConfigurationProperty -pspath MACHINE/WEBROOT/APPHOST -filter "system.webServer/security/requestFiltering" -name "removeServerHeader" -value "True"
```

4. Verify server headers via the following from an elevated command prompt:

```
curl -H 'Host:' https://localhost/aspnet_client --http1.0 -I
```

### *Disable HTTP Options for Microsoft IIS*

1. Follow the steps below to disable OPTIONS method.
2. Open IIS Manager
3. Click on the Control server's name
4. Double click on **Request Filtering**
5. Go to **HTTP Verbs** tab
6. On the right side, click **Deny Verb**
7. Type **OPTIONS**.
8. Click **OK**

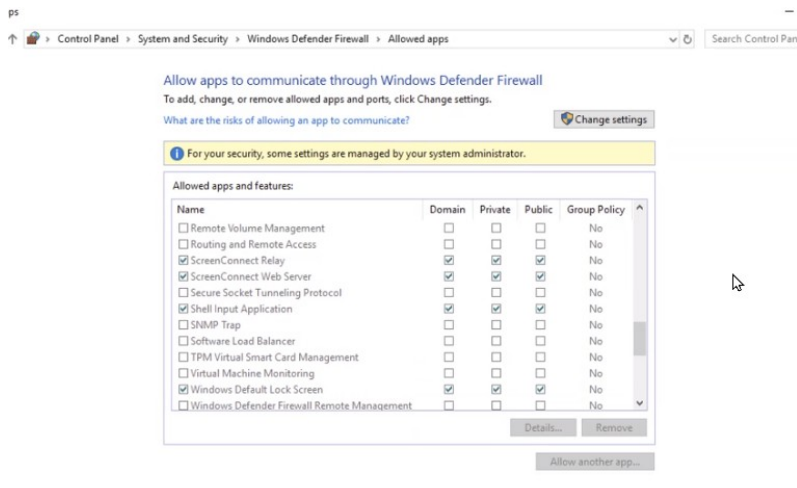


## Network Hardening Guidelines

### *ConnectWise Control Establishes Firewall Rules During Installation*

By default, upon installation, ConnectWise Control adds the following applications to the Microsoft Windows Defender Firewall.

- ScreenConnect Relay (http: 8041)
- ScreenConnect Web Server (tcp: 8040)



### *Recommended Additional Network Restrictions*

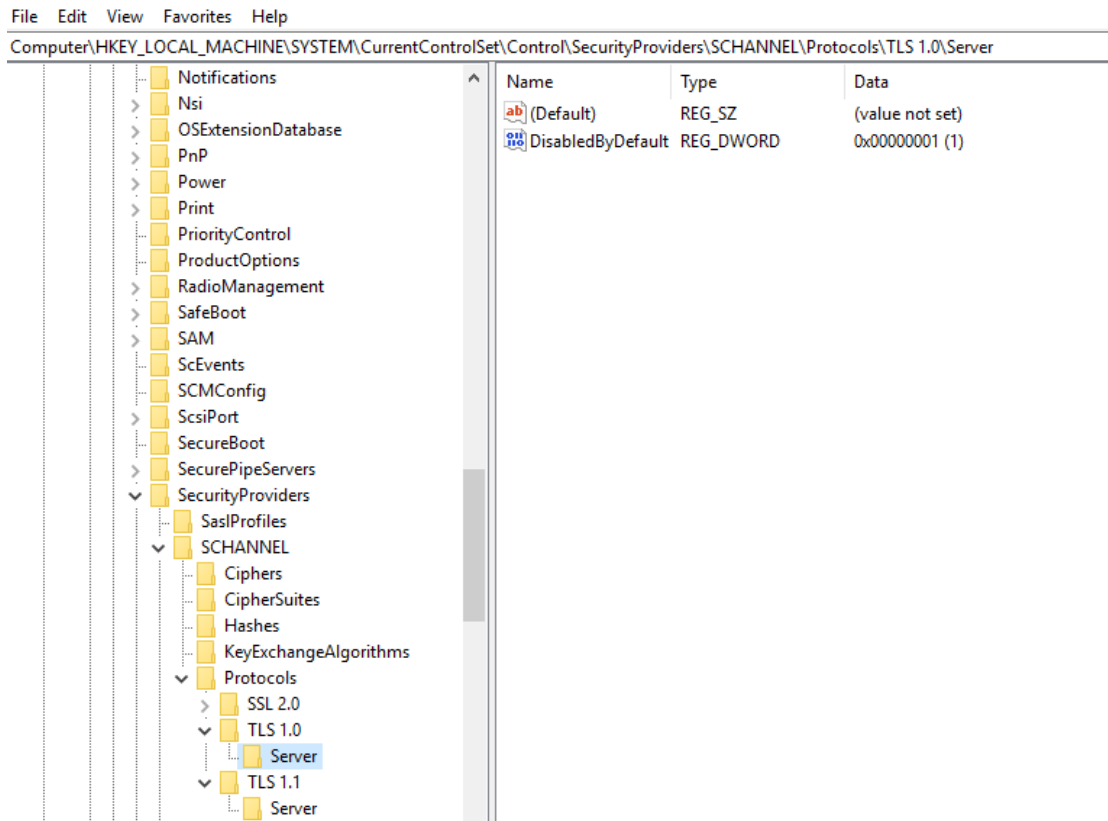
If not connected to Active Directory, the following ports on the ConnectWise Control server can be blocked.

- Port 135: MSRPC. Remove firewall rule to allow TCP-135
- Port 139: NetBios. Remove firewall rule to allow TCP-139
- Port 445: Microsoft-ds. Remove a firewall rule to allow TCP-445

Disabling TLS 1.0 and 1.1 in the Microsoft Windows registry:

1. Open Registry Editor.

2. Go to **HKLM SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols**. Note that the TLS 1.0 or TLS 1.1 entries do not exist in the registry by default.
3. Create a new subkey called **TLS 1.0** under **Protocols**.
4. Create a new subkey called **Server** under **TLS 1.0**.
5. In the Server key, create a DWORD **DisabledByDefault** entry. Set the value to 1.
6. Repeat steps 3 through 5 for **TLS 1.1**.
7. Reboot the server.



## Control Application Architecture & Security

A ConnectWise Control server uses three services: the Session Manager, the Relay, and the Web Server. The software is typically installed onto a single server, with the 3 services running inside a single executable. The .NET process model allows the 3 services to run independently in different .NET AppDomains, providing required isolation between the services.

### Session Manager





The Session Manager service is the data store for sessions. It also controls the creation, destruction, and access to the sessions. Custom applications can interface directly with this service to manage sessions.

It is implemented as a .NET WCF (Windows Communication Foundation) service. By default, it is exposed through a named pipe binding, which is highly efficient, but only processes on the same machine can communicate with it. WCF allows you to show SOAP endpoints among many other protocol options.

## Relay

The Relay service handles all communication between the host and guest clients. This includes session control messages, screen data, file data, and mouse/keyboard data. All data flowing to and from the service is encrypted.

It is implemented with raw TCP sockets. It listens for connections from host and guest clients on a single port (8041 by default). It communicates with the Session Manager Service for session information.

## Web Server

The Web Server service provides a user-friendly interface for hosts and guests to connect before initiating remote control. Hosts are provided a control panel to create, delete, and join sessions. A short wizard helps guests begin the connection.

The Web Server service is implemented as an ASP.NET application. ConnectWise Control does not depend on Microsoft IIS, but can work on IPs/ports alongside IIS, as the same Windows subsystem is used — the HTTP.SYS kernel driver. By default, the Control Web Server services listens on port 8040, and it communicates with the Session Manager service for session information.

Control Server Architecture:

[https://docs.connectwise.com/ConnectWise\\_Control\\_Documentation/On-premises/On-premises\\_knowledge\\_base/Server\\_architecture](https://docs.connectwise.com/ConnectWise_Control_Documentation/On-premises/On-premises_knowledge_base/Server_architecture)

## SSL Certificate Installation

Although ConnectWise Control encrypts all Relay session traffic by default, the Web Server HTTP traffic is not encrypted unless configured with SSL. SSL provides an additional layer of security for key exchange and the comfort of your users. ConnectWise Control does not use IIS, Apache, or any other web platform for SSL.

Refer to the linked article that describes the steps necessary to set up your ConnectWise Control on-premises installation with SSL:

[https://docs.connectwise.com/ConnectWise\\_Control\\_Documentation/On-premises/Advanced\\_setup/SSL\\_certificate\\_installation](https://docs.connectwise.com/ConnectWise_Control_Documentation/On-premises/Advanced_setup/SSL_certificate_installation)

## Application ConnectWise Control Hardening Guidelines





- Set the Control system password with a minimum of 12 characters with a complex mix of uppercase, lowercase, numbers, and symbols.
- Set the MySQL Root password with a minimum of 12 characters with a complex mix of uppercase, lowercase, numbers, and symbols.

<<<<< This page intentionally left blank. >>>>>