everbridge®

Control Center™ Installation Guide

Version 5.28
March 2021

CNL
AN EVERBRIDGE COMPANY

Contents

About Control Center Installation

This guide describes the installation process of Control Center™ on desktops, laptops, and servers. The Control Center installer is easy to use and provides default options to automate the installation process.

Important: There are two important changes to the installation process that will affect the installation where a Control Center database already exists.

1. **Schema management** – The installer will remove any objects from the databases listed below if they are not part of the current Control Center database model. That is, any existing custom tables, stored procedures etc. will be removed. If custom tables are required, create these in separate databases.
    - Pacific
    - Connection Manager
    - Auditing
    - Atlantic

2. **Database cleanup** – Control Center will delete all Control Center objects in the database that do not comply with the database integrity rules. For example, deleting an object manually from the device table would leave invalid rows in other tables such as the permission tables, therefore such entries will now be removed.

Overview of the Installation Process

At a high-level, the installation process covers the following sections:

- Software Requirements and Prerequisites.
- Installing Control Center and Connection Manager installation.
- Database setup during installation. The steps for setting up SQL server is not covered in this document. See the Microsoft SQL Server Configuration Help for instructions.
- Control Center License Certificate, which is the Software license code used for unlocking features and functionality in Control Center.

Control Center Installers

Control Center comes packaged with several components that enable you to accomplish a specific function within a solution. The Server installer includes all the required components except for the Windows Client, which is provided using a separate installer.

Control Center Server Installer

The Control Center Server installer comes packaged with all server-side components.

The Control Center Server installer includes the following components:

- Alarm Types
  - Alarm Types Service
  - Rules Engine Service
- Auditing Service
- Connection Manager Service
- Data Web Service (IIS)
- Federated Service
- GIS Service
- Location Import Tool
- Remote Deployment Tool
- Security Service
- Server Service
- Video Export Service
- Web Server (IIS)
- Web Service

Control Center Windows Client Installer

The Control Center Windows Client installer includes the Windows Client application, which enables you to connect to a Control Center Server installation. Although the Windows Client can easily be installed on the same computer as the server (for example, for training or demonstrations), it is typically installed on separate computers.

Software Requirements

The following sections describe the software requirements for Control Center, depending on what components you are installing.

Important: When installing Control Center, you must install Control Center Server and Control Center Clients with the same major and minor version numbers. For example, if you are installing Control Center Server version 5.20 then you must install Control Center Client Version 5.20 on your client machines.

Control Center Server Requirements

Before installing Control Center Server, ensure that you have installed the required software listed below.

Supported Operating Systems	• Windows Server 2019 • Windows Server 2016
Supported Databases	• SQL Server 2019 • SQL Server 2017 • SQL Server 2016
Other	• Microsoft Data Tier Application Framework 2019 • Microsoft .NET Framework 4.5 (not 4.5 beta) • Microsoft Message Queuing (MSMQ). See Enabling MSMQ.

Note: Everbridge recommends that you do not install the Express edition of SQL Server as it supports databases only up to 10 GB.

Microsoft Data Tier Application Framework

Control Center installation uses Microsoft Data Tier Application Framework to create and manage the Control Center database in Microsoft SQL Server.

When installing Microsoft SQL Server Data-Tier Application Framework on a 64-bit machine, install both 32-bit and 64-bit versions (install 32-bit first).

Microsoft SQL Server Data-Tier Application Framework requires that you install the following components from the Microsoft website for your respective version of SQL Server:

- SQL SqlDom
- SQLSysClrTypes

If you are on an 64-bit machine, you must install both the 64-bit and 32-bit versions of the SQLSysClrTypes.

Control Center Windows Client Requirements

Before installing Control Center Windows Client, ensure that you have installed the required software listed below.

Supported Operating Systems	• Microsoft Windows 10
Microsoft .NET Framework	• Microsoft .Net Framework 4.5 • Microsoft System CLR Types

Prerequisites for Installing Control Center

After installing the software required for Control Center, you must configure some settings to install Control Center successfully. Everbridge recommend you follow the steps described in the following order:

1. Enable named pipes and TCP/IP for SQL
2. Update Group Policy for a Service Account
3. Configure prerequisites for Windows OS

User Account Requirements

Installing and configuring Control Center requires the following.

- Windows user - responsible for installing and configuring Control Center
- Database user - responsible for creating and administrating the Control Center database
- Service Account - responsible for running the Control Center Services and access to the Control Center database

Windows User

The Windows user must have the following roles:

- **db_owner**
- **db_creator**

Database User

The Database user must also have a **db_owner** role. Everbridge recommends that you use Windows authentication when you supply the database run time credentials when installing Control Center. If you use SQL authentication, you must enter a password that is then stored as plain text in other areas of Control Center, for example, dashboards. For security reasons, therefore, it is better to use Windows authentication.

Service Account

The Service Account requires:

- Local System Admin permissions because of the net.tcp port sharing service.
- Read and Execute, List folder contents and Read permissions to the folder where Control Center is installed
- Read Permissions on CN=\Users in Active Directory Users and Computers (if Active Directory is being used)
- Local Policy - Log on as a Service for the account.

Note: Active Directory groups are not required but can be used.

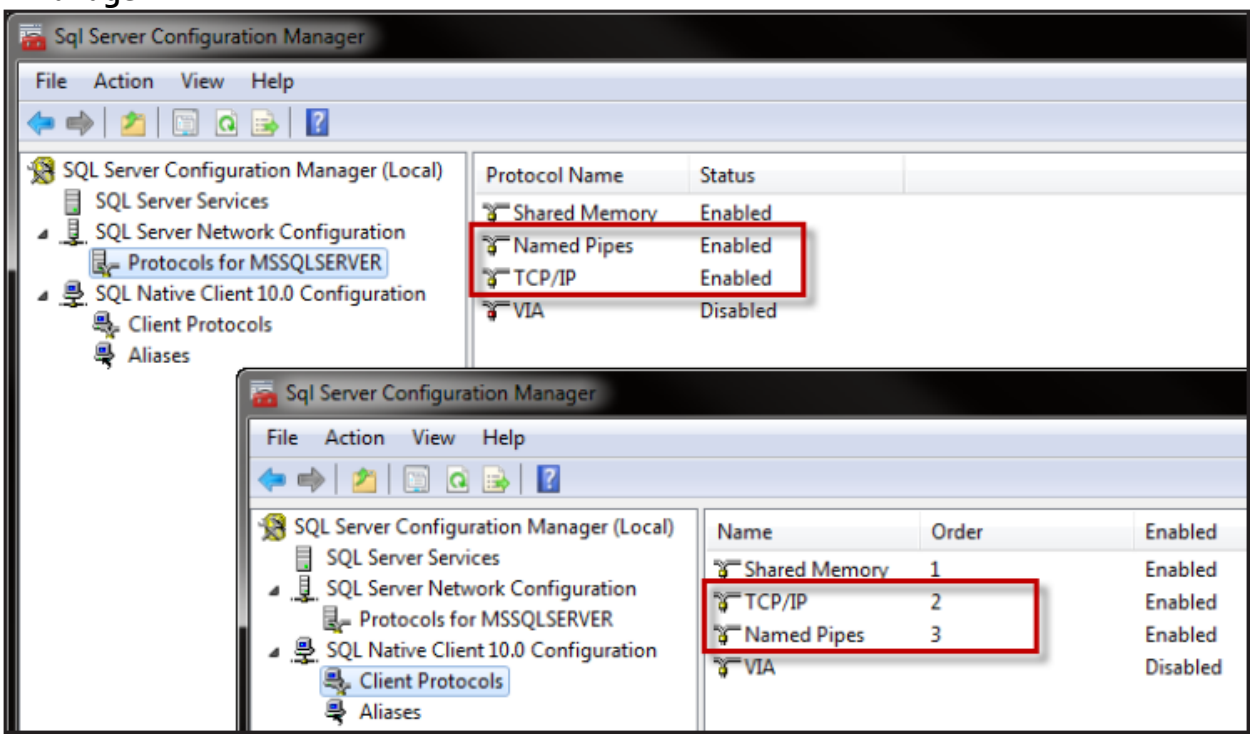
Enabling Named Pipes and TCP/IP for SQL

You must enable named Pipes and TCP/IP communications in Microsoft SQL Server for Control Center to successfully communicate.

Note: This section assumes that you have already installed Microsoft SQL Server.

To enable Named Pipes and TCP/IP:

1. Click **Start** > **All Programs** > **Microsoft SQL Server 2012** > **SQL Server Configuration Manager**.



2. Expand **SQL Server Network Configuration** and enable all Named Pipes and TCP/IP protocols.

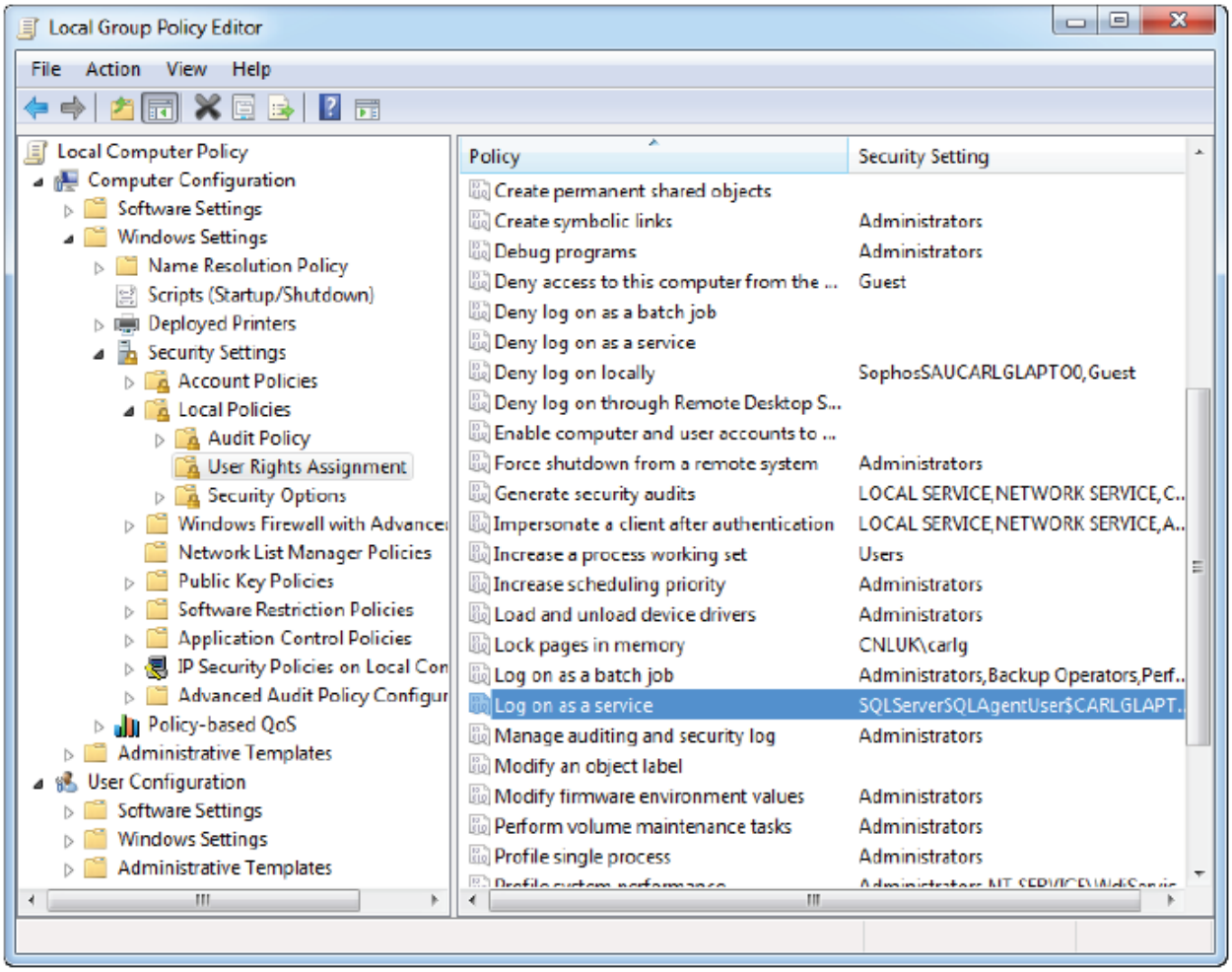
Updating Group Policy for a Service Account

You must run all Control Center services and applications under a known Windows account.

To enable the installer to apply the specified account, update the Log on as a service policy.

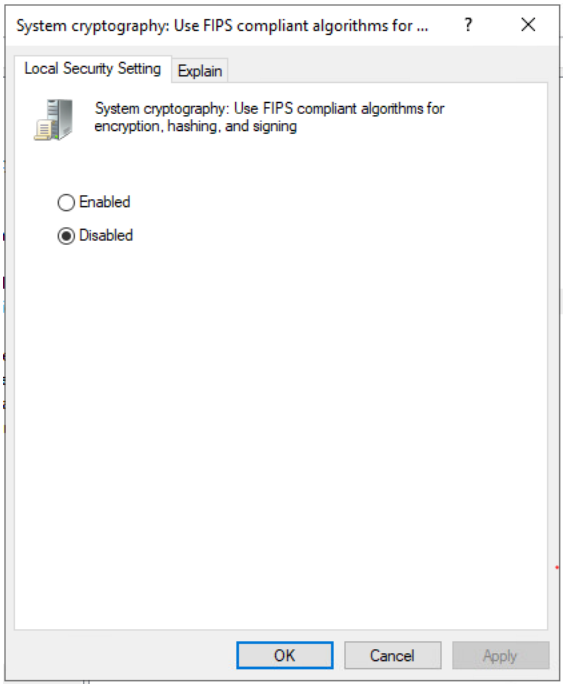
To assign log on as a service policy:

1. Click **Start** > **Run**.
2. Type **gpedit.msc** and then press **Enter**.
3. Expand **Computer Configuration** > **Windows Settings** > **Security Settings** > **Local Policies**, and then click **User Rights Assignment**.
4. On the right pane, right-click **Log on as a service** and select **Properties**.
5. Add the specified users to the policy that are required to run Control Center, and select **OK**.


Local Group Policy Editor
File Action View Help
Local Computer Policy
Computer Configuration
Software Settings
Windows Settings
Name Resolution Policy
Scripts (Startup/Shutdown)
Deployed Printers
Security Settings
Account Policies
Local Policies
Audit Policy
User Rights Assignment
Security Options
Windows Firewall with Advance
Network List Manager Policies
Public Key Policies
Software Restriction Policies
Application Control Policies
IP Security Policies on Local Con
Advanced Audit Policy Configur
Policy-based QoS
Administrative Templates
User Configuration
Software Settings
Windows Settings
Administrative Templates
Policy
Security Setting
Create permanent shared objects
Create symbolic links Administrators
Debug programs Administrators
Deny access to this computer from the ... Guest
Deny log on as a batch job
Deny log on as a service
Deny log on locally SophosSAUCARLGLAPTOO,Guest
Deny log on through Remote Desktop S...
Enable computer and user accounts to ...
Force shutdown from a remote system Administrators
Generate security audits LOCAL SERVICE,NETWORK SERVICE,C...
Impersonate a client after authentication LOCAL SERVICE,NETWORK SERVICE,A...
Increase a process working set Users
Increase scheduling priority Administrators
Load and unload device drivers Administrators
Lock pages in memory CNLUK\carlg
Log on as a batch job Administrators,Backup Operators,Perf...
Log on as a service SQLServerSQLAgentUser$CARLGLAPT...
Manage auditing and security log Administrators
Modify an object label
Modify firmware environment values Administrators
Perform volume maintenance tasks Administrators
Profile single process Administrators

System Cryptography: Use FIPS Compliant Algorithms for Encryption, Hashing, and Signing

You must set the **System Cryptography: Use FIPS complaint algorithms for encryption, hashing and signing** option in your **Local Security Policy** to **Disabled**.



The United States Federal Information Processing Standard (FIPS) 140 standard defines cryptographic algorithms approved for use by US Federal government computer systems for the protection of sensitive data.

Control Center is a software application, built using Microsoft .Net, that runs on the Windows platform.

System cryptography: Use FIPS 140 compliant cryptographic algorithms, including encryption, hashing and signing algorithms is a policy available within Microsoft Windows operating system and disabled by default.

Enabling this policy makes Windows and its subsystems use only FIPS-validated cryptographic algorithms. Enabling FIPS mode also causes the .NET Framework to disallow the use of non-validated algorithms. If FIPS mode is enabled, the .NET Framework disallows the use of all non-validated cryptographic classes. .Net Framework offers multiple implementations of most algorithms, and not all of them have been submitted for validation.

If an application tries to use a cryptographic class that has not been validated, and FIPS mode is enabled, the Framework will raise an exception and not allow the class to be used; this exception will almost always cause the application to fail, if not terminate immediately.

Control Center is using .Net Framework classes that have not been submitted and is therefore likely to fail should the policy be enabled. For this reason, Everbridge recommends the policy to be turned off.

Configuring Prerequisites for Windows Operating Systems

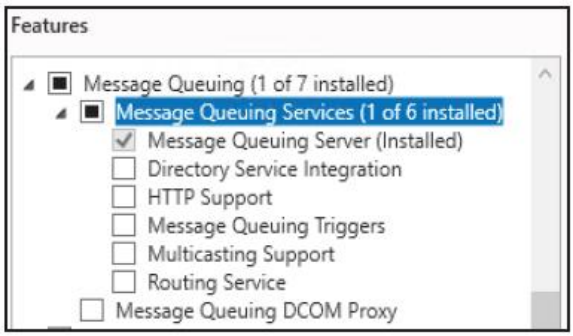
The prerequisites assumed are:

- A clean installation of Microsoft Windows 7 64-bit. The Microsoft Windows installation disk may be required for some steps.
- Microsoft SQL Server and Microsoft .NET Framework 4.5 are installed.

Enabling MSMQ

To enable MSMQ:

1. Open the **Server Manager**.
2. Click **Add roles and features** and leave the default selection as it is.
3. Click **Next** until you locate the **Server Roles** page.



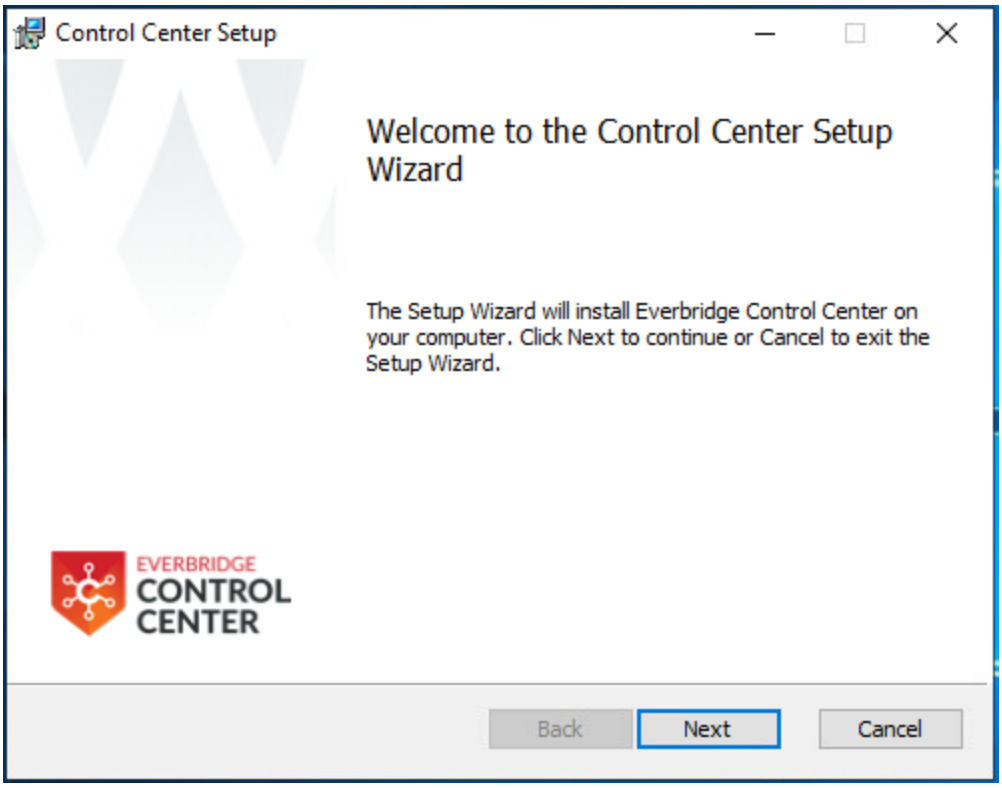
Installing Control Center

Notes:

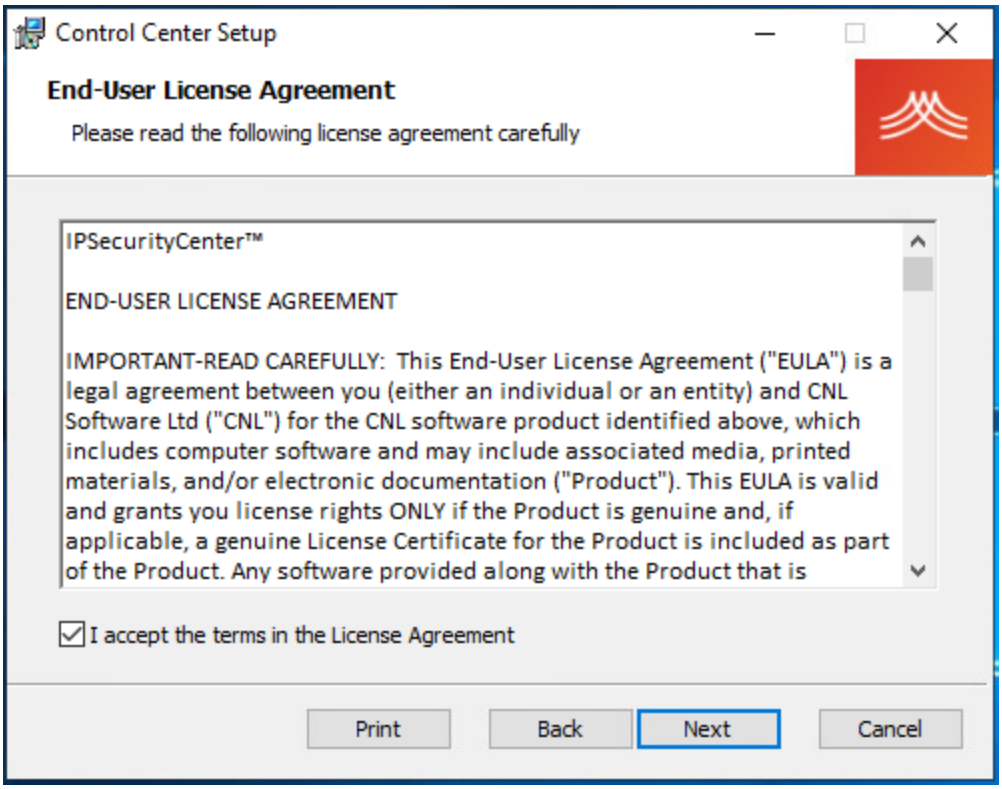
- Before installing Control Center, ensure that you have completed the steps in the Configuring Prerequisites for Control Center.
- The installer will remove any tables in the Control Center database that are not part of the latest Control Center database schema. Do not create custom tables in the Control Center databases as these will be removed on upgrade. If custom tables are needed, create those in a separate database.

Installing Control Center Server:

1. Run the **Everbridge.ControlCenter.Server.Installer.msi** Windows installer package.
2. Read the **Introduction** page and click **Next**.

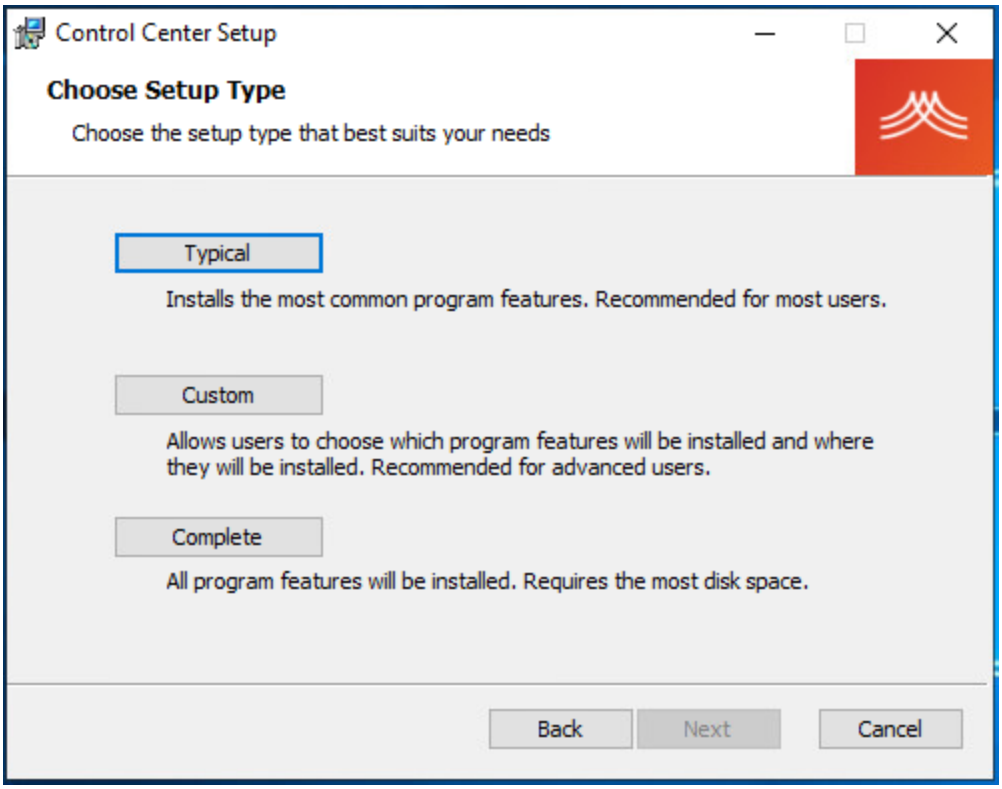


3. Read and accept the license agreement, and then click **Next**.

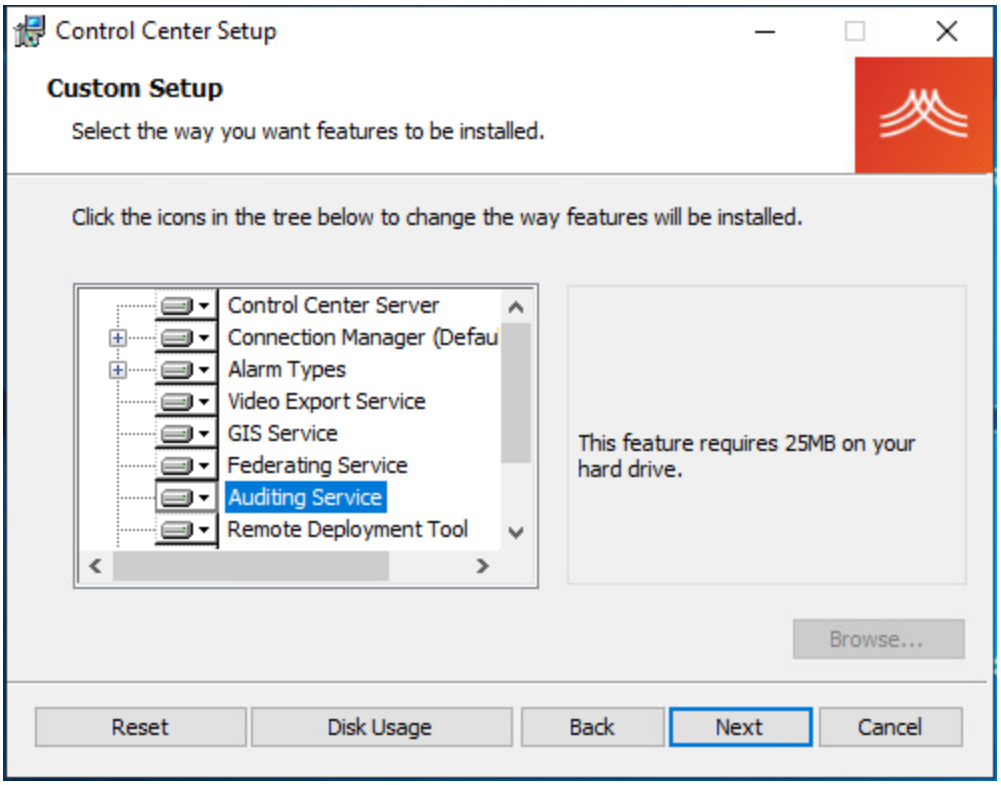


The **Choose Setup Type** page offers the following options:

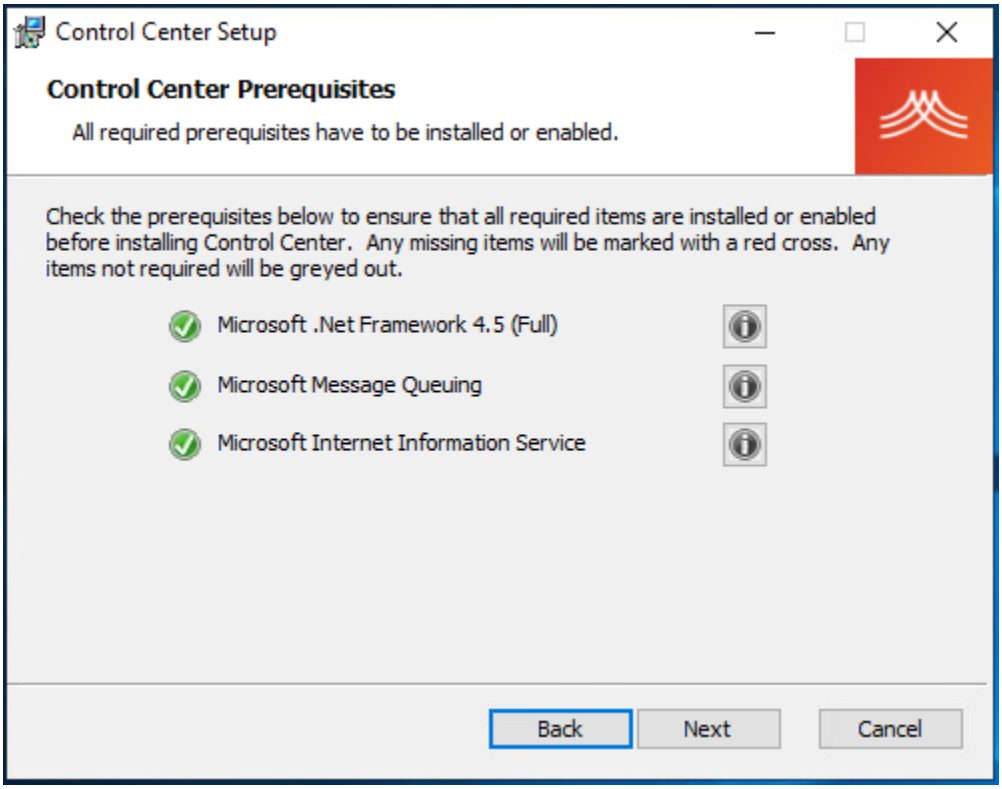
- **Typical** – Provides options to pre-select the Server, Connection Manager, Alarm Types components, GIS Service, Remote Deployment Tool and Location Importer Tool.
- **Custom** – Provides individual component selection.
- **Complete** – Installs all components.



4. Click **Custom** to continue.
5. On the **Custom Setup** page, select the required components for the installation. Typically, you only need the Server, Connection Manager, Reporting, GIS, and Alarm Types components. Select all the required components and click **Next** to continue.

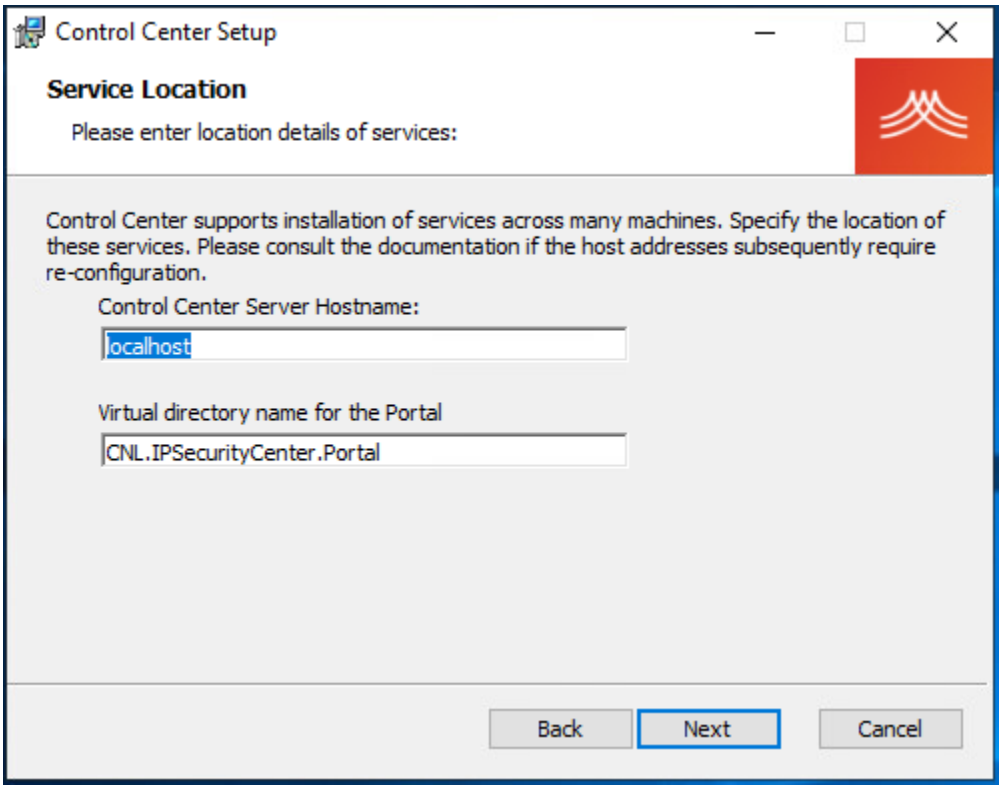


6. Ensure that all prerequisites are met. The **Control Center Prerequisites** page will check for all the required components and report the ones that are missing. Click **Next** to continue.



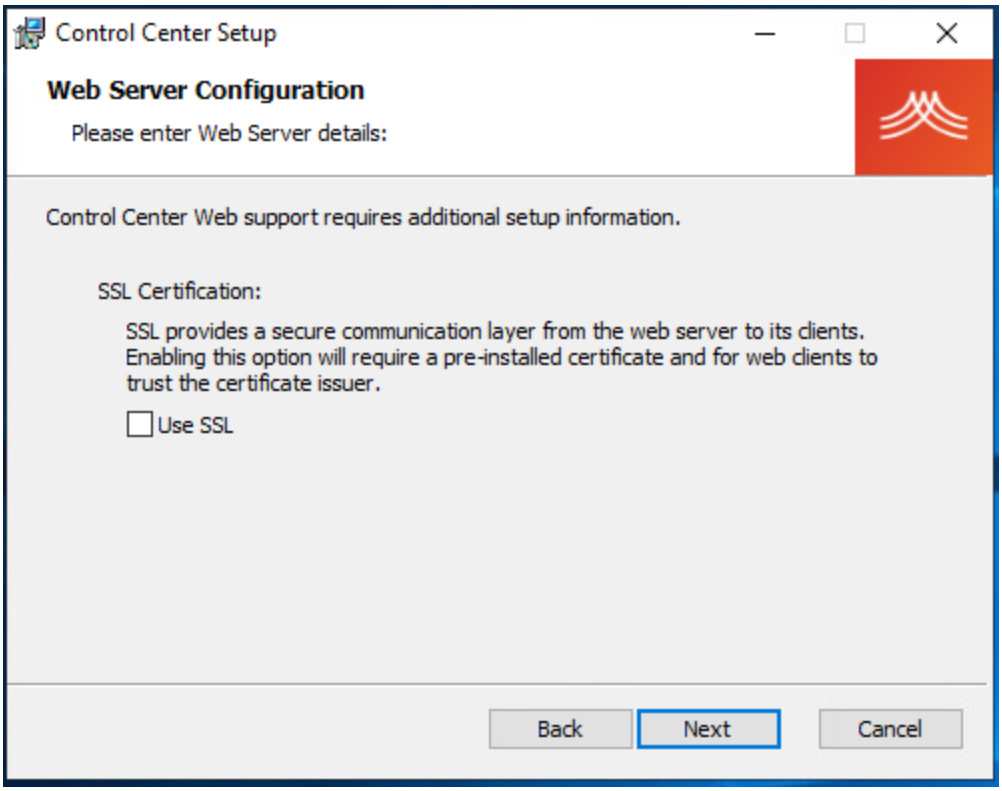
7. On the **Service Location** page, specify the hostnames for the different services in the solution.

Note: This page is only shown when installing either the Web Server or the Data Web Service components. See Streaming Server Installation Guide.

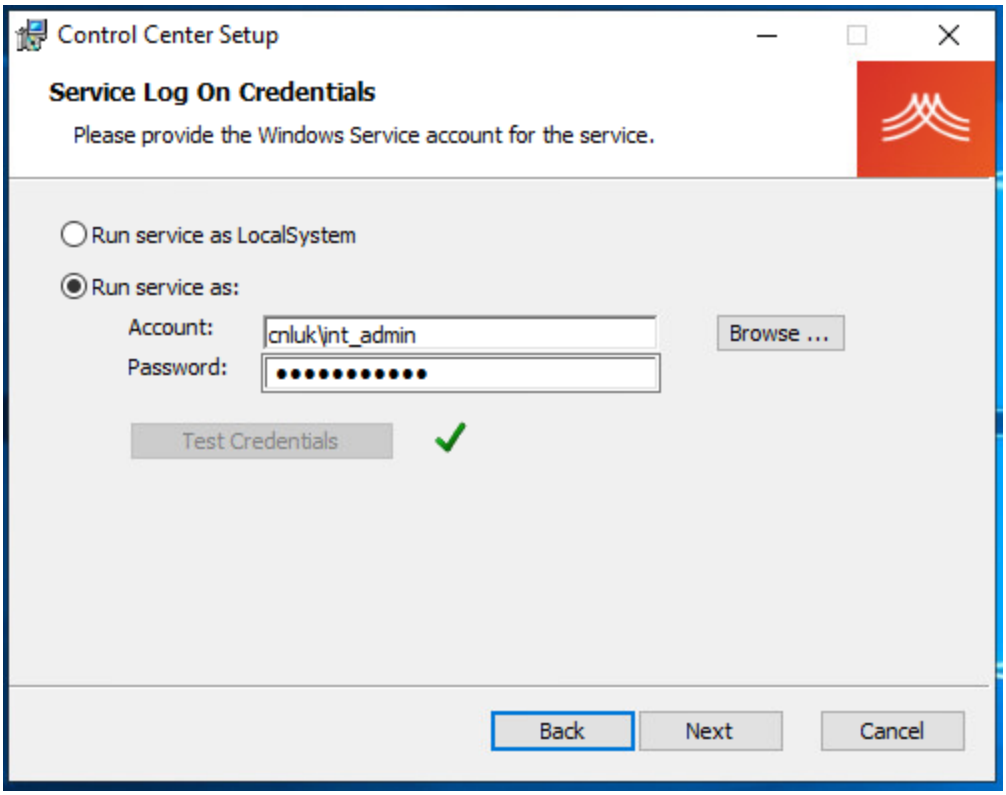


The **Web Server Configuration** page includes an option for specifying an SSL certificate for secure communications.

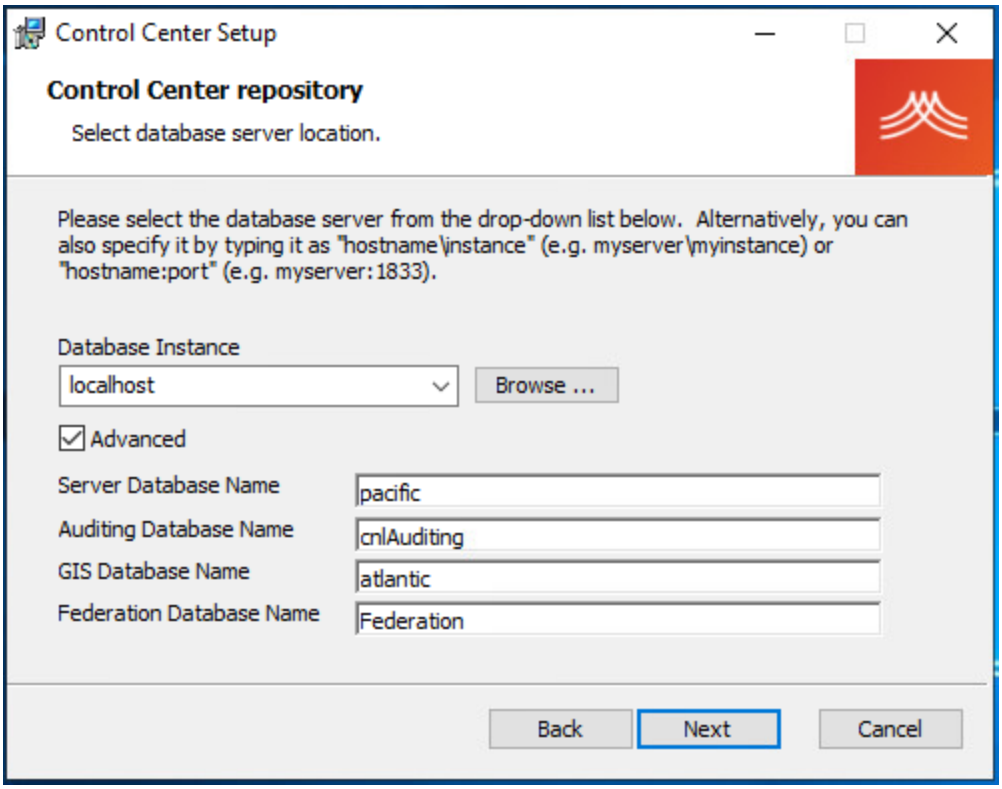
Note: This page only appears if you selected the Web Server or the Data Web Service components on the Custom Setup page. See Streaming Server Installation Guide.



8. Specify the log on credentials for all the different Windows services and IIS applications. It is recommended to use a dedicated account to run the various services instead of the **LocalSystem** account for greater control over security. Additionally, **LocalSystem** typically does not have the required access levels for SQL Server.
9. Enter a valid service account and password, and click **Test Credentials**. Everbridge recommends that you use Windows authentication. If you use SQL authentication, you must enter a password that is then stored as plain text in other areas of Control Center, for example, dashboards. For security reasons, therefore, it is better to use Windows authentication.



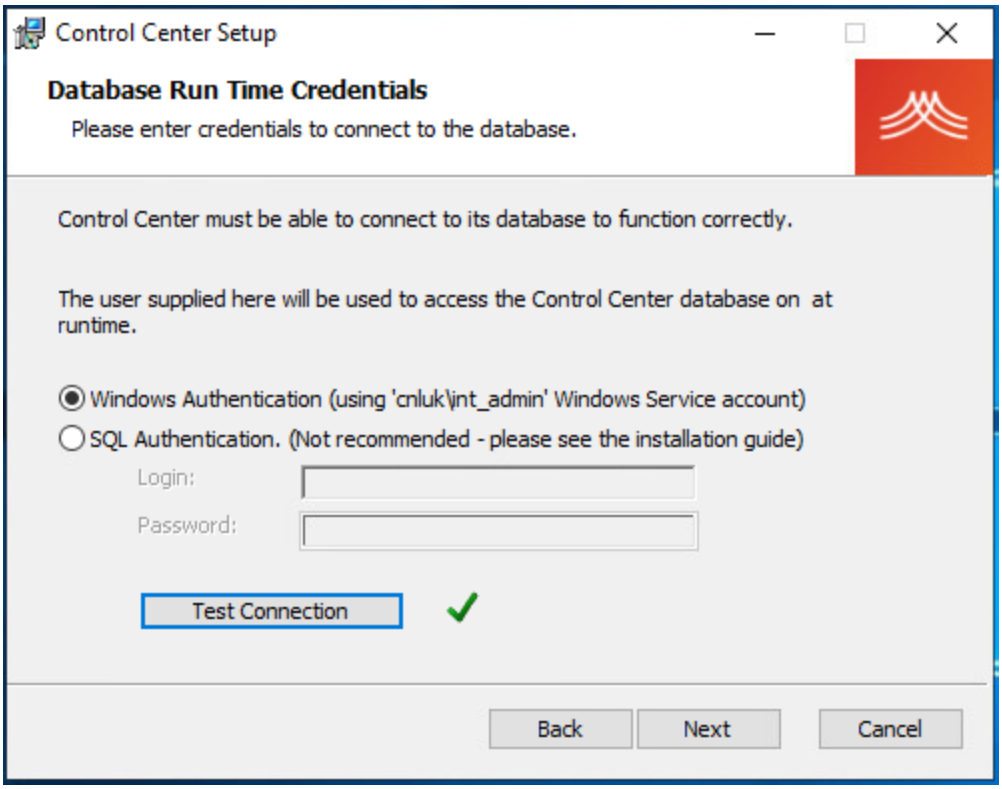
10. Click **Next**. The installer will create or update the necessary databases during installation. Specify the database instance to contain the Control Center databases. The installer will use a set of pre-defined database names. It is particularly useful if multiple Control Center instances are using the same database server. To change the pre-defined database names, select the **Advanced** checkbox.



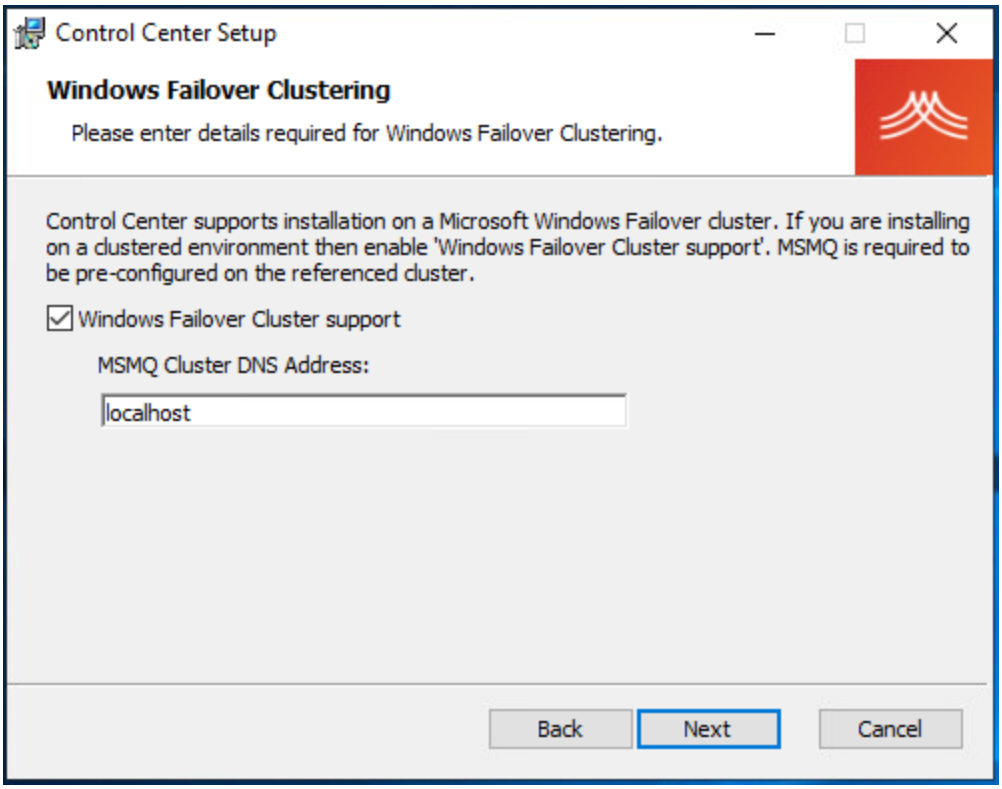
Note:

- The installer will update the database content if you select an existing database.
- The Control Center Installer assumes that your SQL server database is using the default port number, **1433**. If your SQL server database is not using the default port number, then you can install to another SQL server database temporarily, restore the database to the correct SQL server and manually update your `connectionstings.config` file.

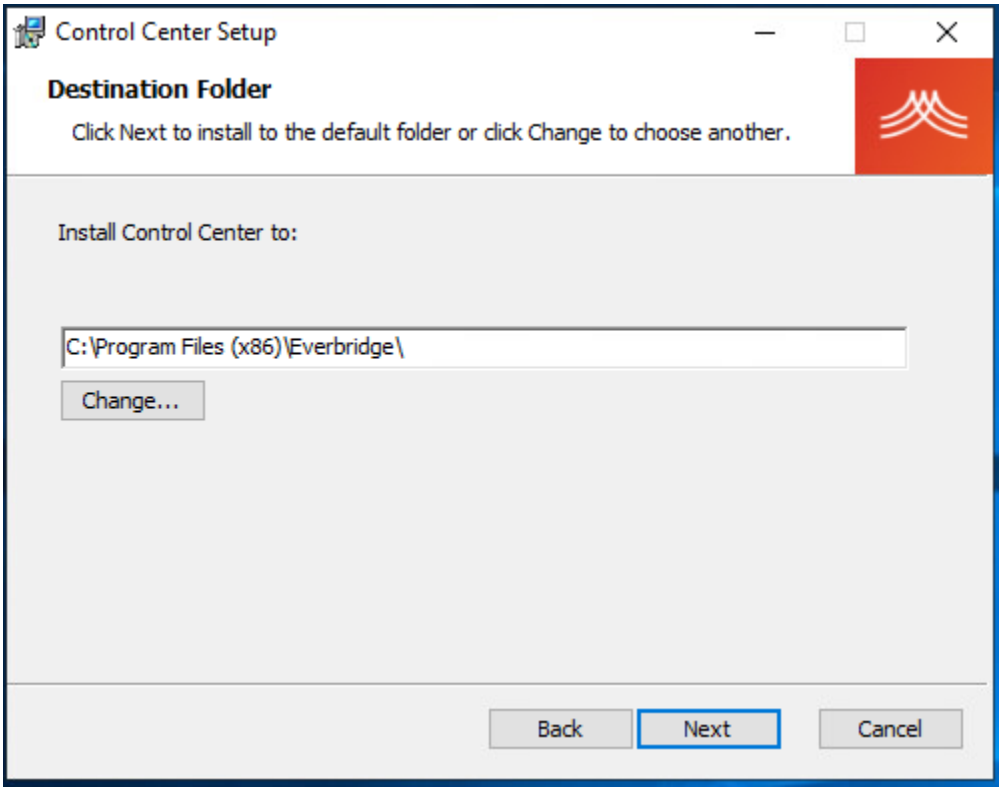
11. Click **Next** to continue. Once the correct SQL instance is specified, you must specify the credentials that you want to connect with. If you want to use Dashboards in Control Center, then you must use Windows Authentication.



12. Click **Test Connection** and then click **Next**.
13. On the **Windows Failover Clustering** page, leave the option deselected and then click **Next** to continue.

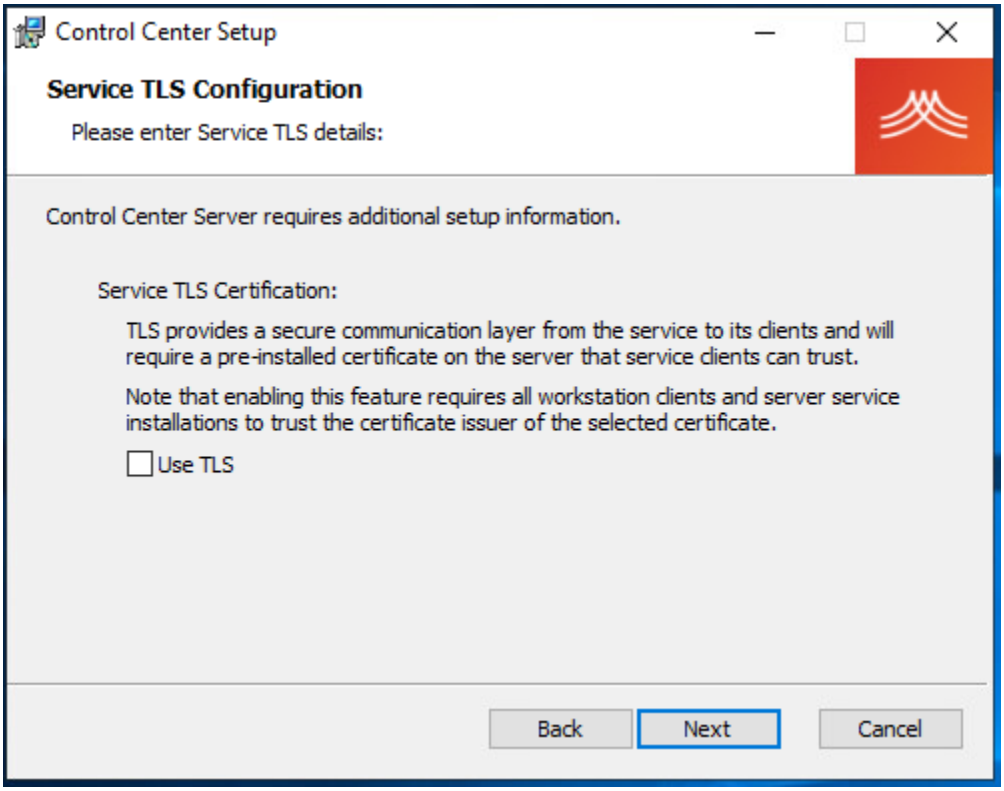


14. Select the destination folder in which the components need to be saved. By default, all components are stored under `c:\Program Files(x86)\Everbridge` folder.



Note: When Server and Client are saved in different custom locations, the addons installed on the Control Center will be saved on both server and client side.

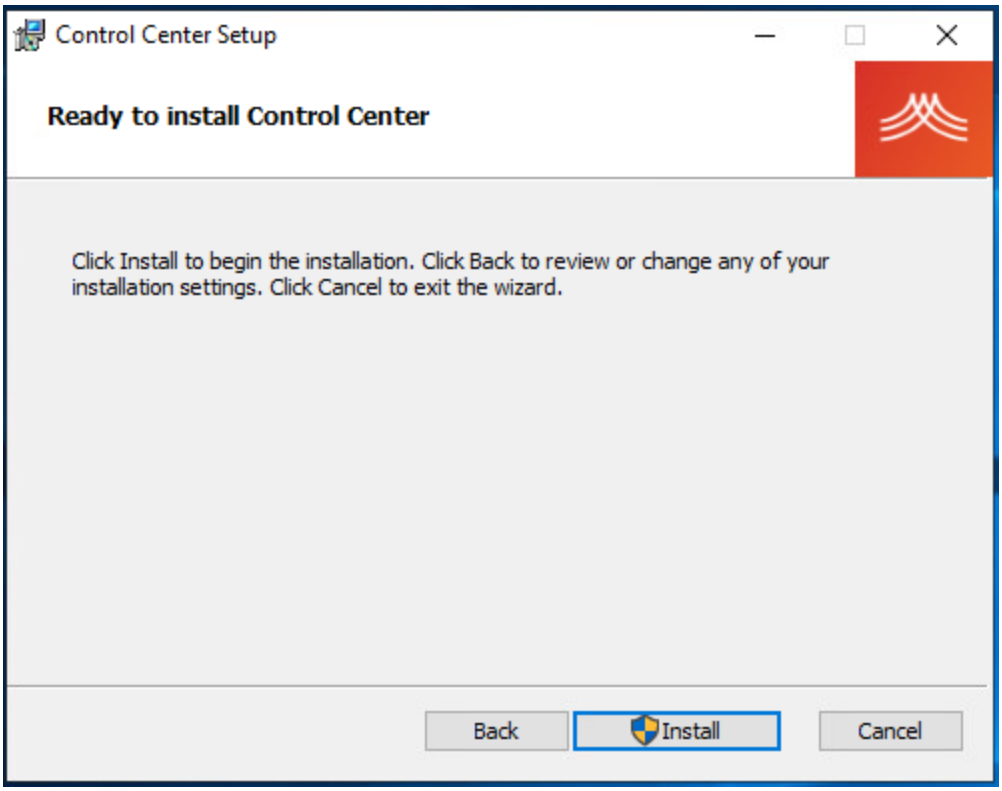
15. Enable the use of **Service TLS** if you wish to have a more secure communication channel between Server and Clients. This requires a pre-installed certificate on the server that the service clients can trust.



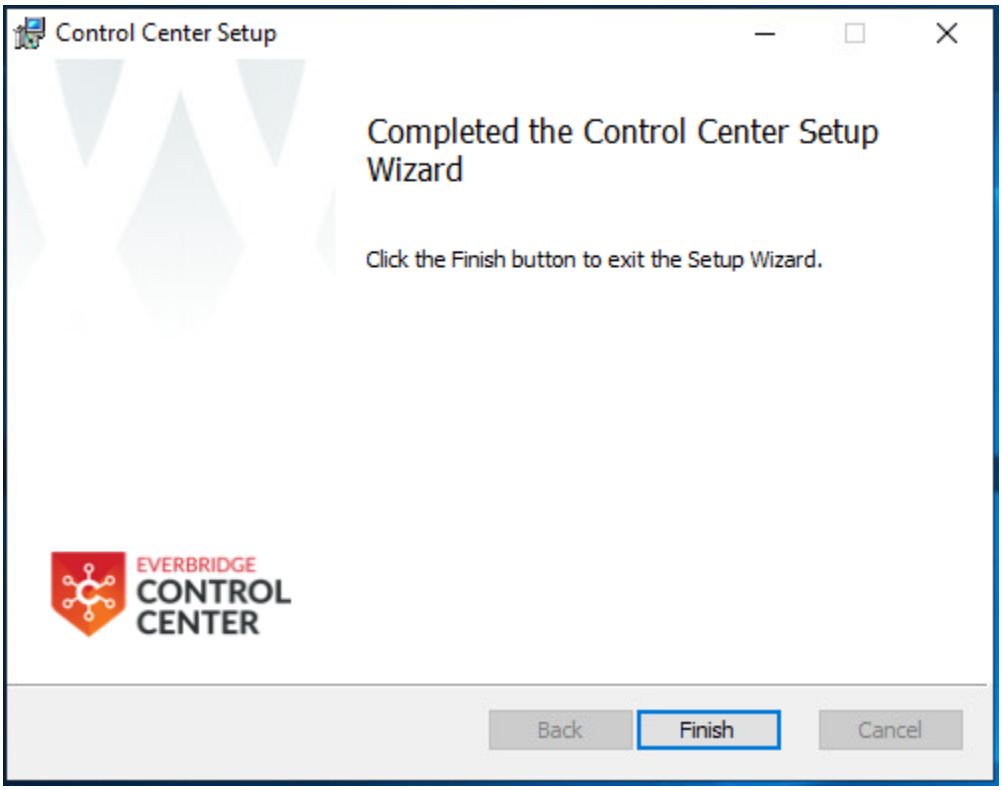
16. By enabling the **Use TLS** option, a drop down menu of all the certificates installed on your local machine is displayed. Choose the one you want to use and click **Next**.

Note: Ensure that the certificate is installed in the Windows Local Machine Personal Certificates store. For more details on TLS, see TLS for secured connection between Server and Client.

17. Click **Install**.



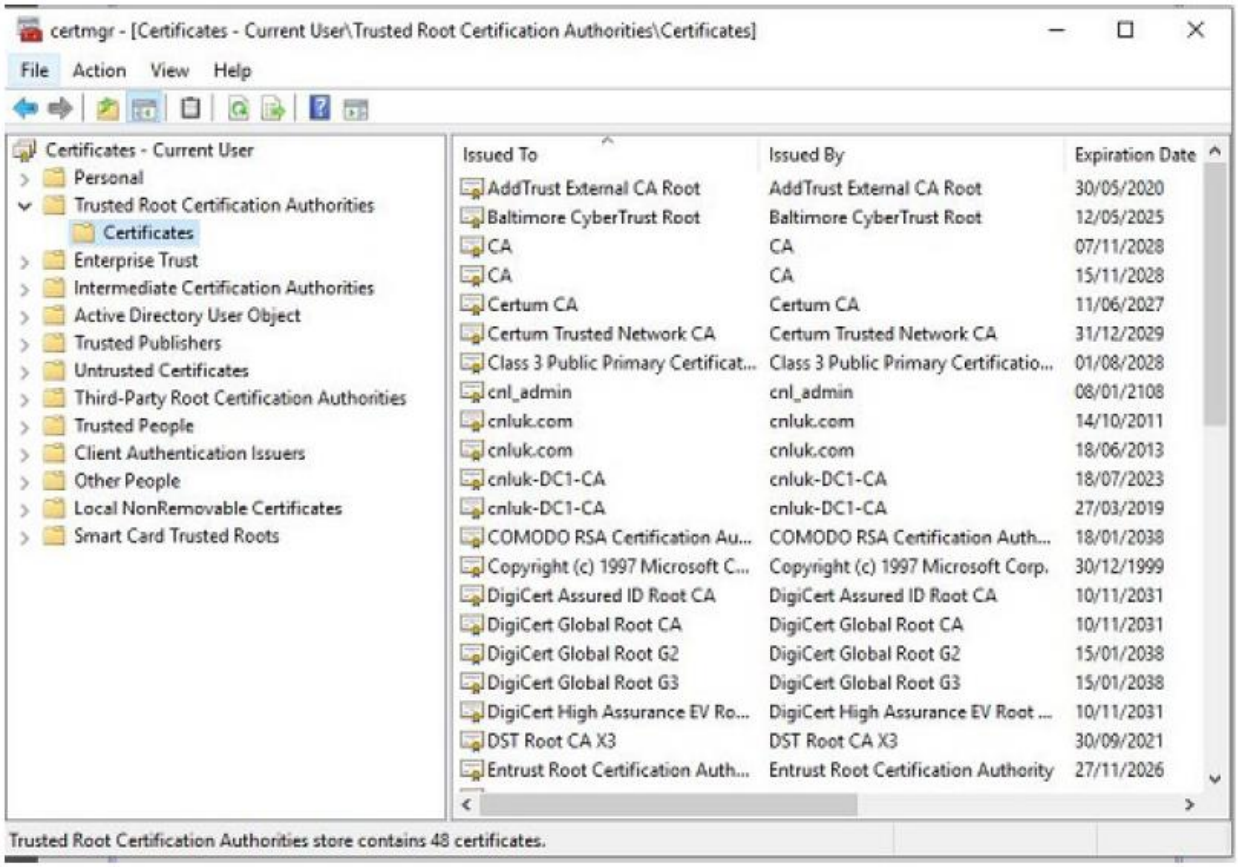
18. Click **Finish** to complete the Control Center Setup Wizard.



TLS for Secured Connection Between Server and Client

Server TLS authentication or certificate based authentication refers to the client verifying the server it needs to connect to, through the digital certificate provided by the Certificate Authorities. In real terms, the client requests the server for authentication and the server presents its certificate issued by the trusted Certificate Authorities (CAs) which the client validates and a secure communication channel is established.

As authentication relies purely on authorizing the digital certificate, certification authorities such as Verisign or Microsoft Certificate Server are an important part of the Server authentication process. You need to purchase the certificate from any trusted authorities or host your own trusted certificates and install them on your Windows Local Machine Personal Certificates store. You could verify the certificates installed on your machine by going to Manage Local Machine Certificates window. You could either navigate through Control Panel or type `certmgr.msc` on the command prompt to open the window for you.



From a high-level point of view, the process of authenticating and establishing an encrypted channel using certificate-based authentication involves the following steps:

A client requests access to a protected resource on the Server

1. The server presents its certificate to the client for authentication
2. The client verifies the certificate presented by the Server
3. If successful, a handshake is initiated between the server and the client.
4. The server grants access to the protected resource requested by the client

Everbridge suggests the following recommendation listed below for a seamless secure communication channel between Server and client.

- Always configure Windows Operating System and .NET Frameworks to only use TLS 1.2 and above
- Always Enable Security provider's 'Use Strong Crypto flag'
- For more information on Server setup, we recommend you read Microsoft .Net TLS Best Practices and Microsoft Channel TLS1.2 Reference

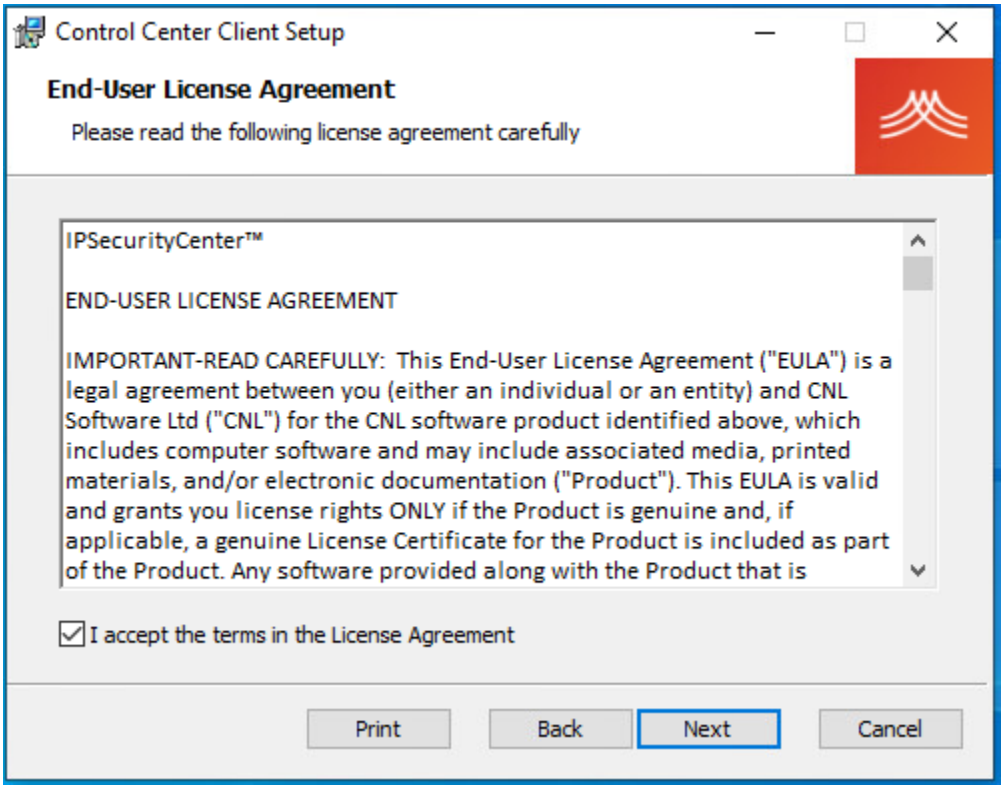
Installing Control Center Client

The Control Center Windows Client provides a front-end to the Users. It is where you perform most configuration of a Control Center solution.

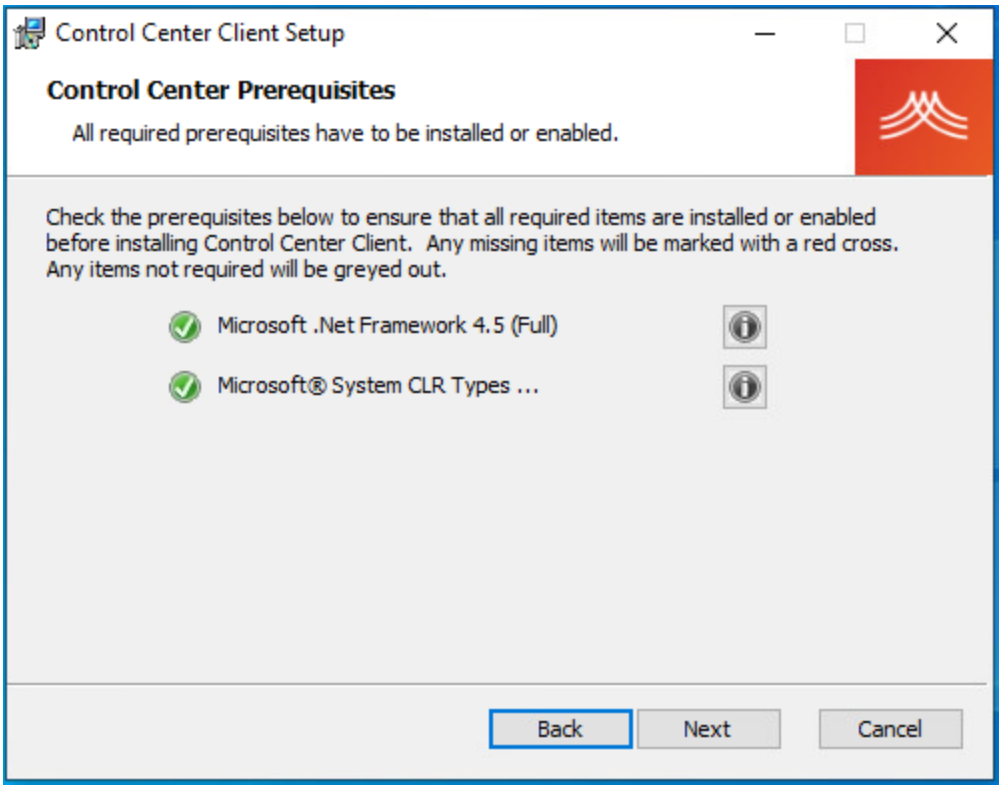
Note: If using TLS, the client certificate needs to be installed before doing the Client installation

To install Windows Client:

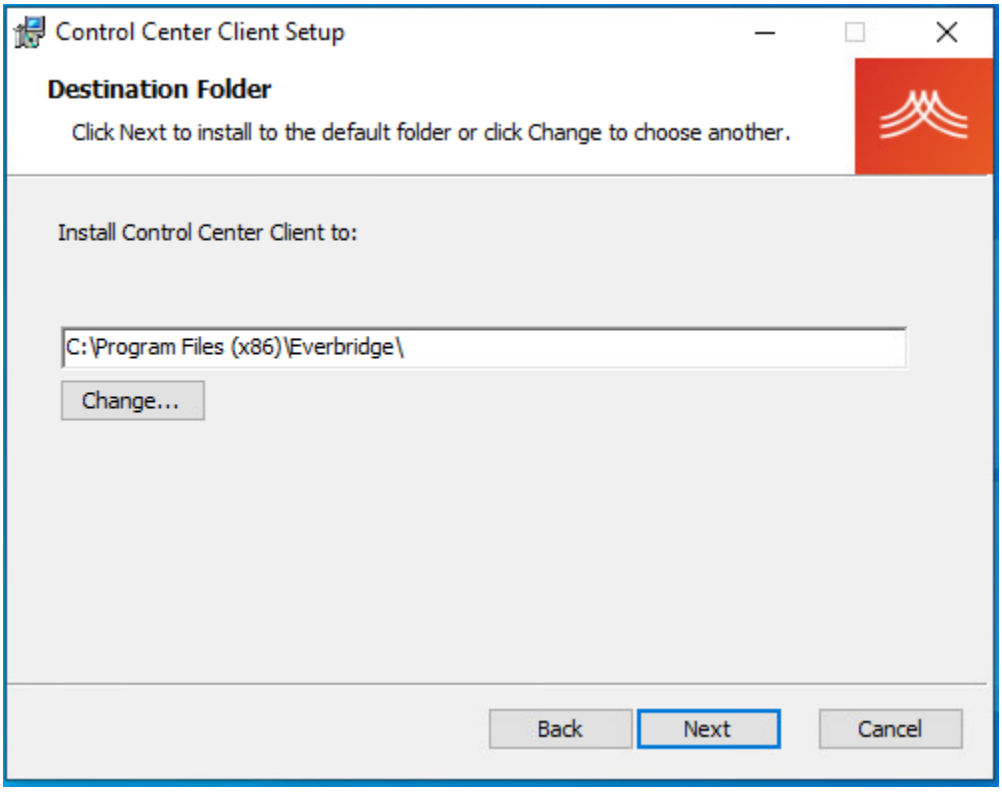
1. Run the **Everbridge.ControlCenter.WindowsClient.Installer.msi** Windows installer package.
2. Read the introduction page and click **Next**.



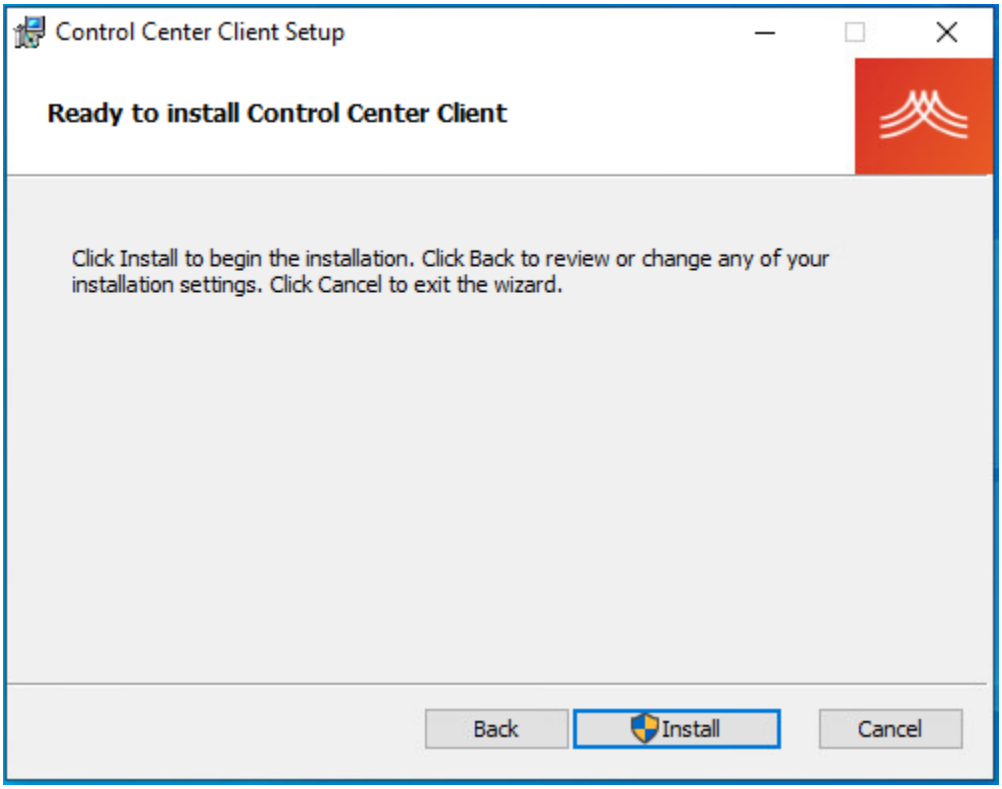
3. Ensure that all prerequisites are met. The **Control Center Prerequisites** page will check for all required components and report any that are missing. Once all prerequisites are met, click **Next**.



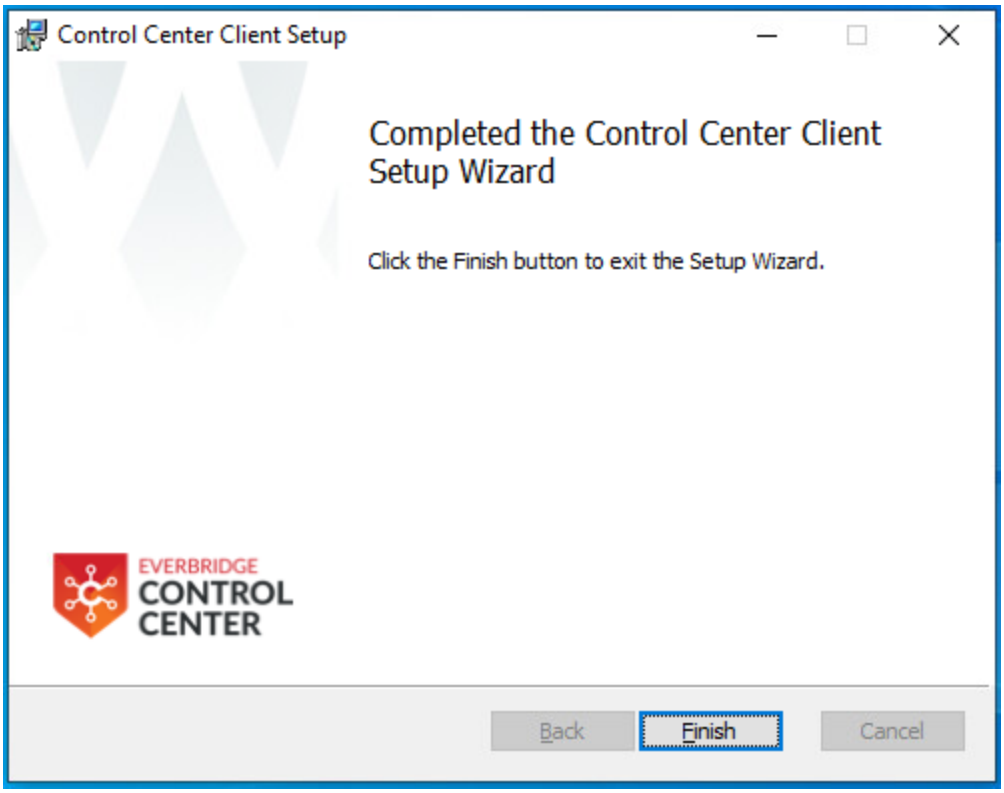
4. Select the destination folder in which the components need to be saved. By default, all components are stored under `c:\Program Files(x86)\Everbridge` folder.



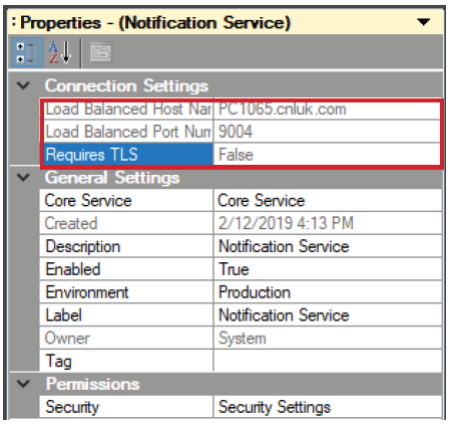
5. Click **Install**.



6. Click **Finish** to complete the **Control Center Windows Client Setup Wizard**.



Note: The TLS connection settings can be verified in the properties of the Notification Service. Go to **System Configuration** > **Services** and click on Notification Service. The properties window on the right displays the connection settings for TLS. The properties are read-only and is not allowed to be modified by the user.



Starting Windows Services

Once the Control Center Server installation is complete, you must start the Control Center Windows services. Make sure to start all services detailed in the Prerequisites section.

1. Click **Start** > **Run**, type **services.msc** and then press **Enter**.
2. Start all the Control Center windows services using a valid Log On As account.

Configuring No Domain Tool

When running Control Center outside of a domain, where the client is located on a different computer to the server, you must update certain configuration files to allow the client to successfully communicate with the server.

The default configuration is applicable for a domain and you must change it only if connecting to the server from a separate computer. If Control Center is setup for a domain, or if there is no domain and the client and server are running on the same computer, then you can use the default configuration.

A `Tools.SwitchFile` is included when installing Control Center to update the configuration files for the required type of configuration (domain or no domain). This tool copies the relevant configuration files to the appropriate folders based on the configuration selected. To locate the `Tools.SwitchFile`, go to **Program Files** > **ControlCenter Tools**. The Control Center installers will place the tool at the following location:
`\Everbridge\ControlCenter\ControlCenter Tools`

You must run the `Tools.SwitchFile` from the Command Prompt as an Administrator. The following steps will run the tool to toggle the configuration between domain and no domain.

1. Click **Start** > **Run**, then type **cmd** and click **Run as administrator**.
2. Go to **Program Files > Everbridge > ControlCenter > ControlCenter Tools**. For example, `C:\Program Files (x86)\Everbridge\ControlCenter\ControlCenter Tools`
3. Type **Everbridge.ControlCenter.Tools.SwitchFile.exe** and click **Enter**.

Running the `SwitchFile`tool will detect the current configuration and then switch the configuration files to the opposite setting. For example, if you run the tool repeatedly, it will toggle between domain and no domain configuration.

Alternatively, you can also specify the required configuration using parameters when running the `SwitchFile`tool. The available parameters are as follows:

- **/nodomain** – Move to a no domain configuration
- **/domain** – Move to a domain configuration

Connection Manager

The Control Center Connection Manager is designed to process all communications to and from devices. You can install multiple instances of the service in any of the solutions. In addition, the Connection Manager services can reside on computer that do not have the Control Center Server installed to accommodate load balancing.

You should install the Connection Manager from a dedicated application using the separate installer included with the installer.

Note: The application is limited to 16 instances of a connection manager on any one machine.

It is recommended to install separate connection managers for the following reasons:
High volume of assets - You can spread across many assets in a solution across multiple connection managers to spread the load. Alternatively, you can locate the connection managers on separate hardware to improve performance.

- **Reduce network traffic and processing** - The Connection Manager enables filtering of events before they are sent to other Control Center services. A filter can be applied to disregard events closer to the source, particularly if the connection manager is installed on or near to the sub system.
- **Increase stability** - Stability issues in a solution caused by a troublesome sub-system or device driver can be isolated to a separate Connection Manager to improve the stability of other devices.
- **Connection Manager Redundancy** - Multiple instances of the Connection Manager service can be configured to act as failover nodes for the other instances without the need for a Windows Cluster.

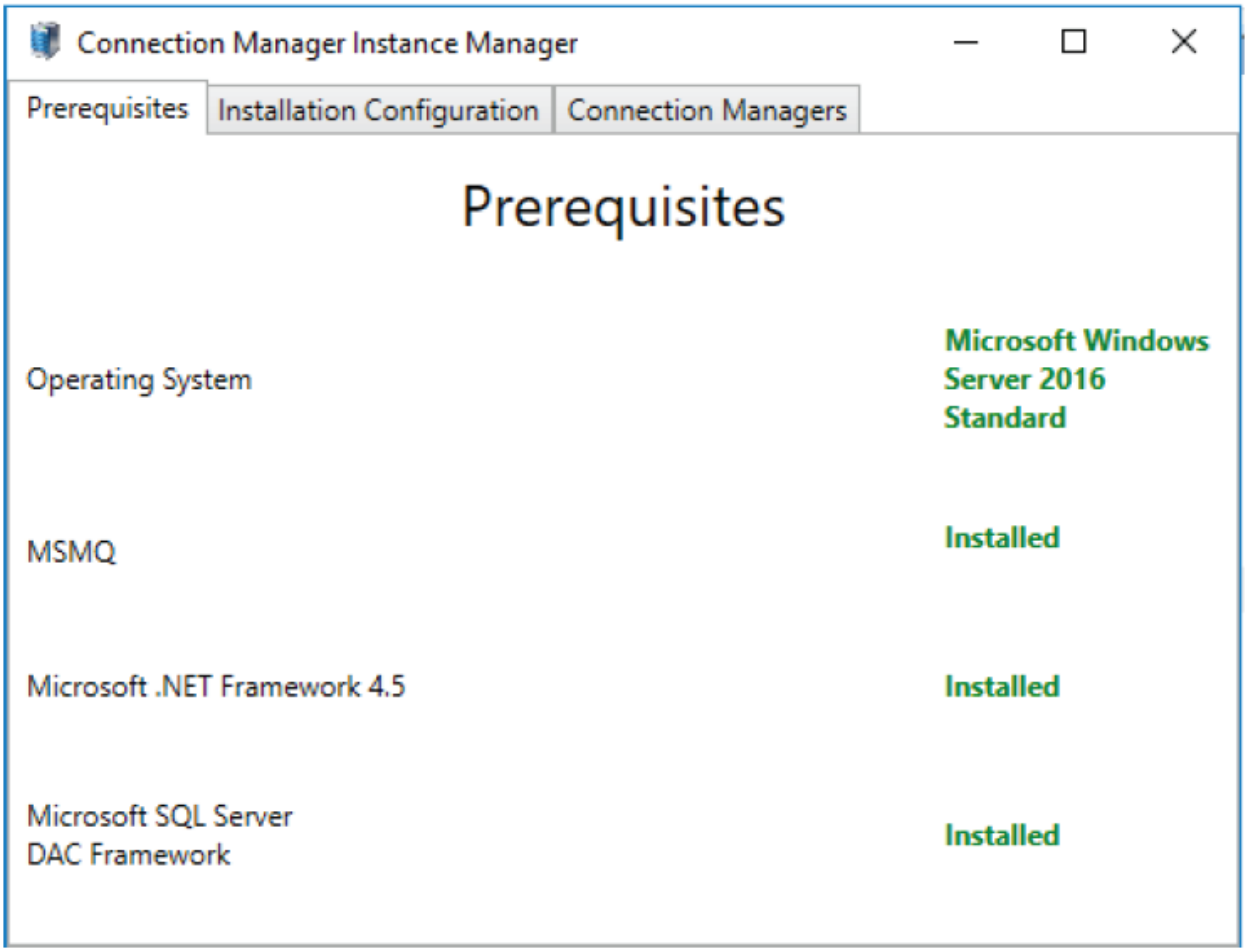
Installing Connection Manager

The Connection Manager installer comprises the following files:

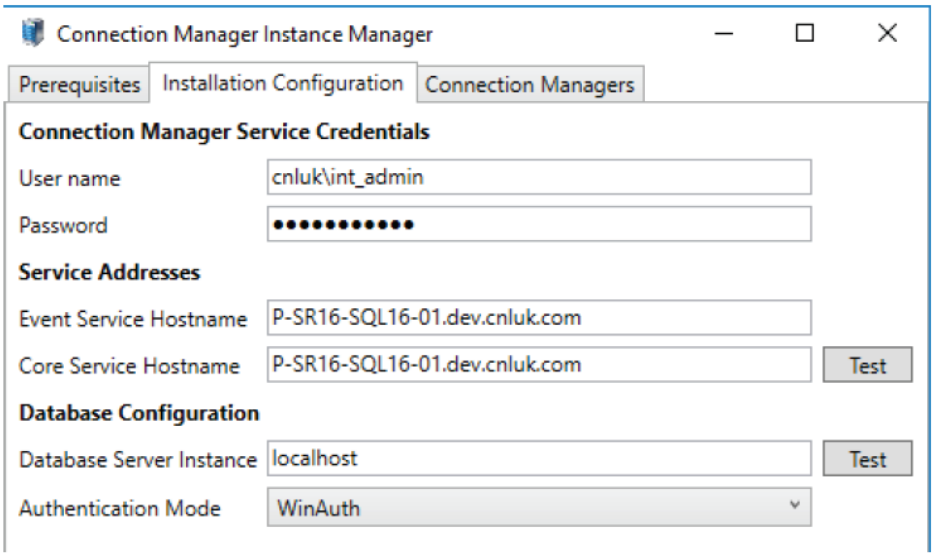
- **Everbridge.ControlCenter.ConnectionManager.Installer.exe** - Loads an application to manage the installed Connection Managers.
- **Everbridge.ControlCenter.ConnectionManager.Installer.exe** - Performs the installation using the application.

To install the Connection Manager:

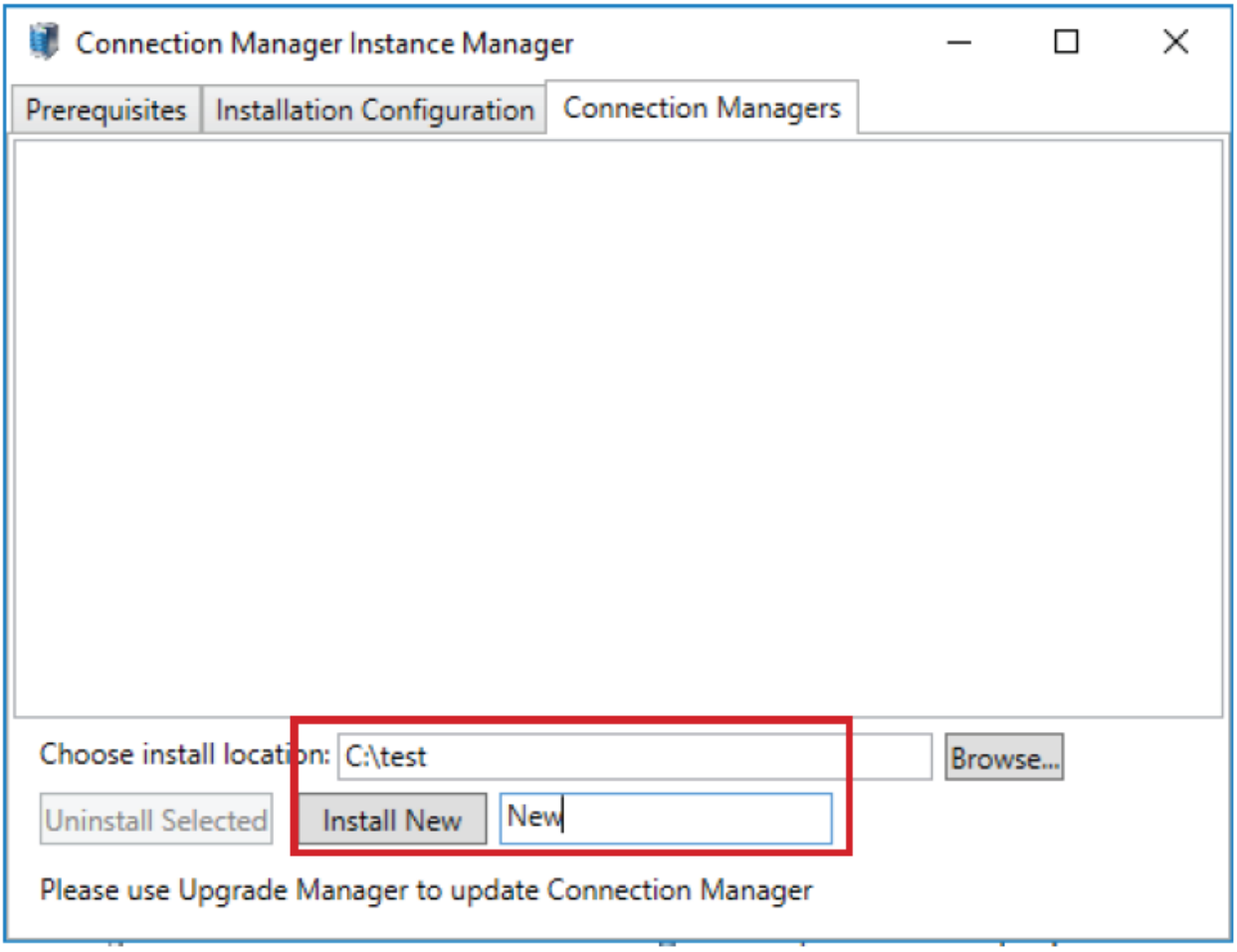
1. Double-click to run the **Everbridge.ControlCenter.ConnectionManager.Installer.exe**. The Connection Manager Instance Manager will appear.
2. Verify if all the prerequisites are met and install applications that appear with a warning message.



3. Before installing the Connection Manager service, navigate to the **Installation Configuration** tab and enter the following connection details:
    - **Connection Manager Service Credentials** – Domain\User credentials of the account being used.
    - **Service Addresses** – Hostnames for the Event and Core Control Center services.
    - **Database Configuration** – The Database Server instance name and authentication mode (Windows authorization or SQL Server authorization).

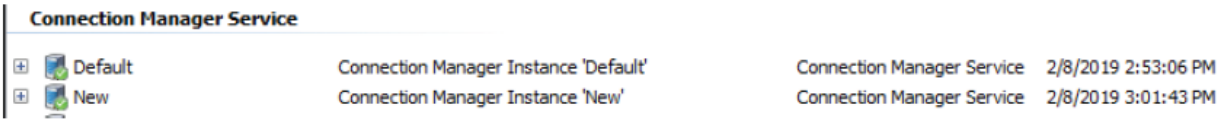


4. Navigate to the **Connection Managers** tab and enter a name for the new Connection Manager. Note that all instances are always stored in the default location. The user can also choose to save the instances in a custom location. Browse through to the location you want to save and then enter the instance name.
5. Click **Install New** to install the connection manager service with the specified name.



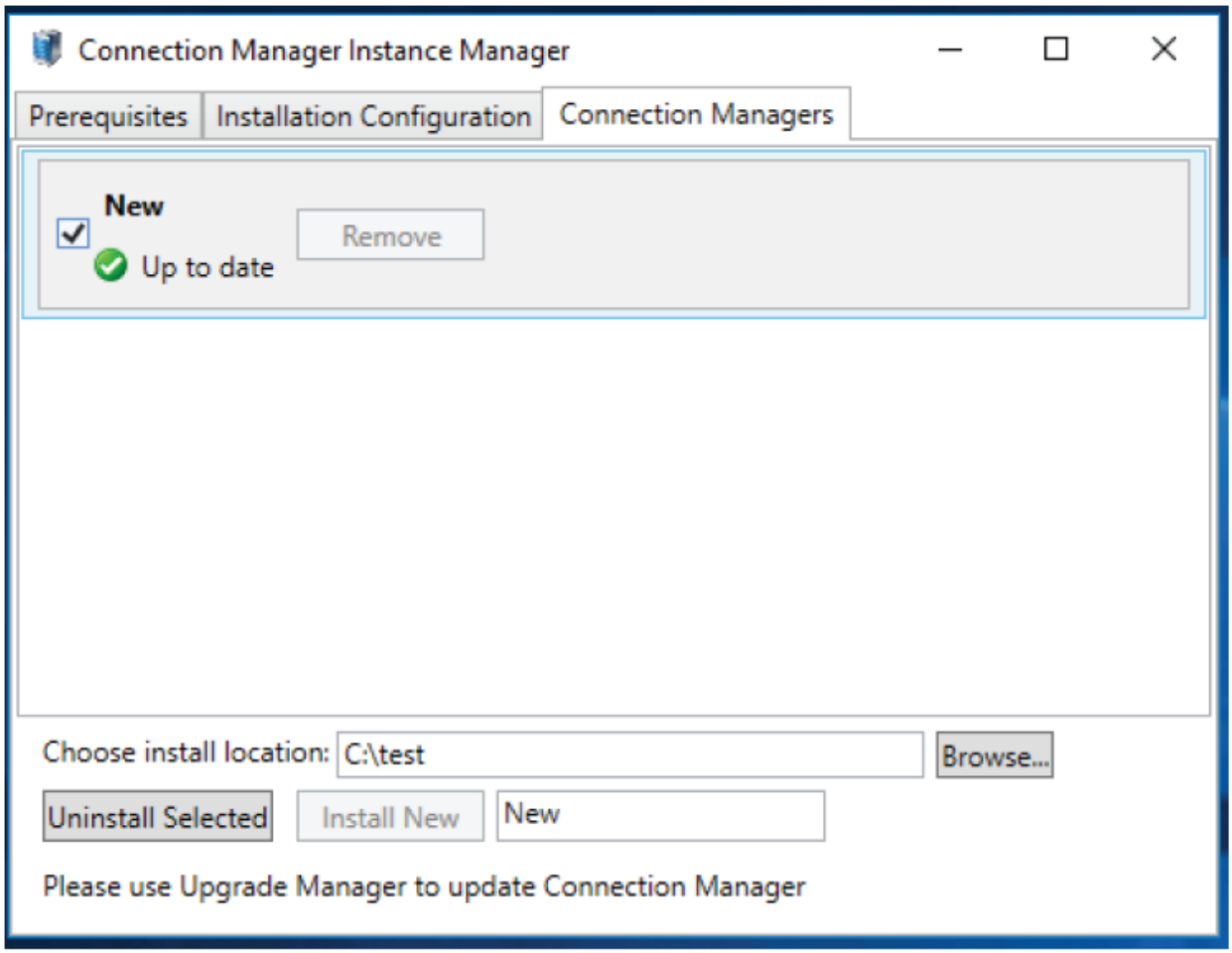
A new connection manager service is installed.

6. Once the installation of the service is complete, start the Control Center Connection Manager Service (New).
7. The Connection Manager creates a corresponding record for itself in the specified Control Center solution as per the Installation Configuration tab of the Connection Manager Installation Manager. The new service object can be found in the **Services** folder.



To update an existing Connection Manager and database stored in the default location, you need to use the Upgrade Manager to make all necessary updates. However, if the Connection Manager instances are stored in the custom location it has to be uninstalled manually and a newer version installed in the desired location.

To uninstall a Connection Manager, select the Connection Manager instances that you want to remove, and then click **Uninstall Selected**.



Upgrading Control Center

The recommended upgrade process is to use Control Center Upgrade Manager. This automates the upgrade of Control Center. It can upgrade all server components (except web) as well as the client application and its plugins.

Important: Upgrade Manager requires Microsoft .NET Framework 4.7.2

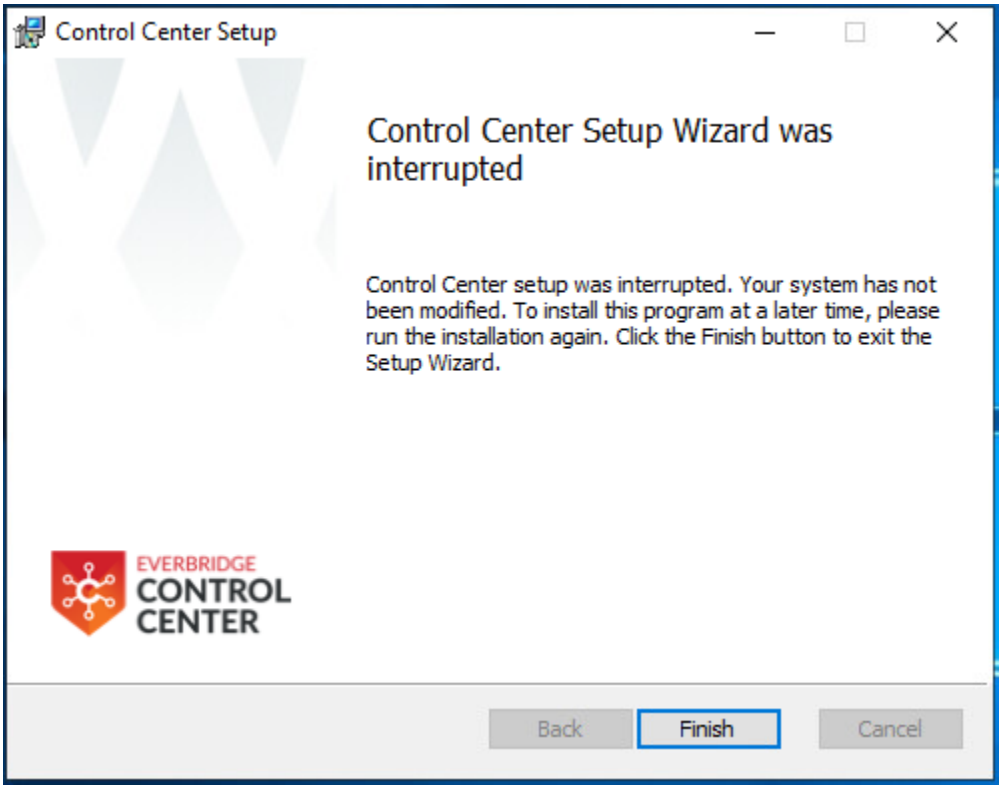
With the Upgrade Manager, you do not have to perform the following additional steps that are required when you install the software normally:

- Manually uninstall each product or add-on component one-by-one.
- Select the correct .msi packages to re-install based on the previous packages installed.
- Manually run multiple packages one-by-one.
- Provide the correct configuration for service accounts, database connection, and additional information collected by the installer UIs that was lost during uninstallation.
- Manually restart the same services that were running before installation.

Troubleshooting

Control Center Installation Fails to Complete

The Control Center Installation wizard ended prematurely.



This issue typically relates to permissions issues and must be investigated by creating a log file from the installation.

To resolve this, investigate the issue by generating a log file of the install. To generate a log file for the database installer:

1. Copy the installer to the root of the local disk, for example C:\.
2. Click **Start** > **Run**, type cmd, and then press Enter.
3. Navigate to the folder containing the installer.
4. Paste the following text and then press enter: `msiexec /i "Everbridge.ControlCenter.Server.Installer.msi" /l*v log.txt`
5. Run through the installer and then inspect the log file created on the desktop. For example, the following message appears in the log file when a database cannot be created, **CAQuietExec: *** SQL01268 C:\Program Files\Everbridge\pacific.sql(39,0) .Net SqlClient Data Provider: Msg 5170, Level 16, State 1, Line 1 Cannot create file 'C:\Databases\Current\pacific.mdf**, because it already exists.
6. Change the file path or the file name, and retry the operation.

Note:

- When looking for errors in the log file, search for return value 3 and the error should appear above an instance of this text near the end of the log file.
- Database needs to be NOT participating in an availability group.

Installer Error: Sqlpackage.exe has Stopped Working

The Control Center installer fails and reports that sqlpackage.exe has stopped working. The installer then rolls back and fails to install successfully.

This error message is related to the missing Microsoft SQL Server components that are required for the SQL Server 2012 Data-Tier Application Framework.

To resolve this, install the following files:

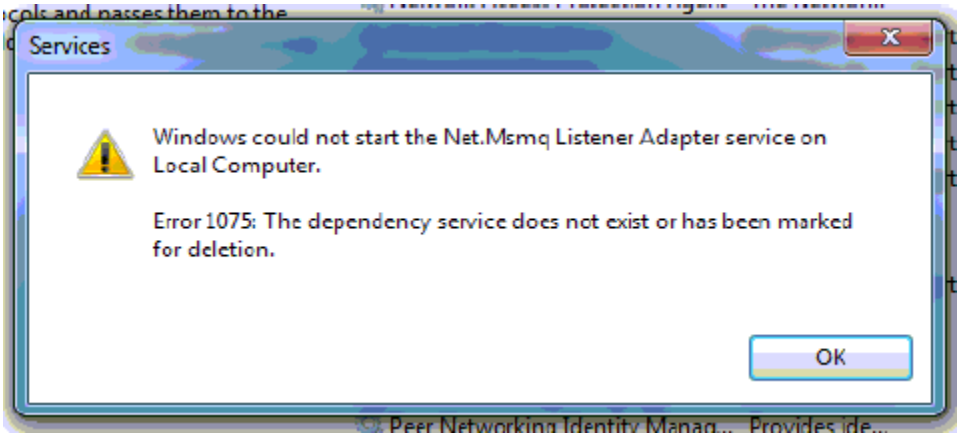
- SqlDom.msi
- SQLSysClrTypes.msi

https://www.microsoft.com/en-gb/download/details.aspx?id=52676

Note: If installing on an x64 machine, install both x64 and x86 versions of the MSIs.

Unable to Start Net.Msmq Listener Adapter Service

The Control Center installation wizard displays an error message on the **Select Components** page, when you click **Next** after selecting the **Install the Auditing service** option.



This is because Microsoft MSMQ has not been correctly installed.

To resolve this:

1. Configure MSMQ correctly and restart the machine.
2. Run through the prerequisites, see Configuing Prerequisites for Control Center.

General Service Failure

Control Center services fail to start or do not function as expected. The Windows event log and Debug View do not report any exceptions.

To resolve this, run the suspected services, including the Server service, in debug mode. Follow these steps to run the Alarm Types service in debug mode. The server service is also included as this can also indicate the cause of the issue.

1. Ensure that all services run in debug mode are stopped. For example, stop the Control Center AlarmTypes Service and the Control Center Server Service.
2. Open a Command Prompt dialog for each service to be run in debug mode. In this case, 2 command prompt dialogs are required.
3. Navigate to the **Program Files** > **Service** folder, filter the folder contents by type **Application**, and then drag the file into a command prompt window.
4. Type `debug` after the file path and then press **Enter**.
5. Repeat this for all services to be run in debug mode and then check the command prompt dialog for any debug information to identify the issue.

Waiting for Security Service to Start

The Windows Client is successfully communicating with the Control Center Server, but the following message is displayed.

The Client application cannot reach the Control Center Server workstation using its hostname.

To resolve this, edit the hosts file on the workstation where Control Center Client is installed. To edit the hosts file:

1. Locate the hosts file in the following path: `c:\windows\system32\drivers\etc\hosts`
2. Edit the hosts file to add the hostname of the Control Center Server workstation and the IP address at which it can be reached. For example:

```
#      102.54.94.97     rhino.acme.com          # source server
#       38.25.63.10     x.acme.com              # x client host
```

Note: Refer to the documentation for your version of Windows for the steps required to update the hosts file.

Appendix

Port Description

The following tables describe the executables, ports, protocol and direction that has to be considered when configuring network security.

Some services, such as the Rules Engine and Alarm Types Service communicate within the same host (not via the firewall) and are thus excluded from the tables.

Furthermore, certain windows services shall be configured and enabled and able to talk to each of the components in the architecture to ensure the system works correctly. These are:

- NTP
- Active Directory
- Ping - used to check availability of services between Control Center Client and all other components
- MSMQ – used to send messages between some subsystems and Control Center server (see https://support.microsoft.com/en-gb/help/183293/how-to-configure-a-firewall-for-msmq-access)

For the purposes of this document it is assumed that MS SQL Server is configured to use its default ports of TCP/UDP 1433 and 1434.

Control Center Server

Server Inbound

Name & Executable	Port	Protocol	Source	Notes
ipscserver.exe	9000	TCP	Client	Core Server
Everbridge.ControlCenter.AlarmTypes.WindowsService.exe	9003	TCP	Client	Alarm Types Service
System.exe	9004	TCP	Client	Notification Service - SignalR Listening on HTTP, and TCP level protocol is handled by OS, hence no executable.
Everbridge.ControlCenter..GIS.WindowsService.exe	9005	TCP	Client	GIS Service
System.exe	9006	TCP	Client	GIS Service - Listening on HTTP, and TCP level protocol is handled by OS, hence no executable.
System.exe	9007	TCP	Client	The Connection Manager is listening on HTTP, and TCP level protocol is handled by OS, hence no executable and running

				under the [System] process. 9007 is used by SignalR HTTP (Event viewer).
System.exe	9008	TCP	Client	Listening on HTTP, and TCP level protocol is handled by OS, hence no executable.
Everbridge.ControlCenter.Security.WindowsService.exe	9009	TCP	Client	Security Service
Data Web services	9010	TCP		Signair for Notification to Web Clients
SMSvcHost.exe SMSvcHost64.exe	9099	TCP	Client	Connection Manager use the Net.Tcp Port Sharing Service (SMSvcHost). 9099 is used by Device services.
Default\Everbridge.ControlCenter.Driver.ConnectionManager.Windows Service.exe	9100	TCP	Client	The instance name is specific to the installation. One rule has to be configured per instance. Used by Connection Manager services to modify configuration.
Everbridge.ControlCenter.Federation.WindowsService.exe	9901 9902	TCP	Client	Federated Service
Everbridge.ControlCenter.WindowsClient.exe	7333	TCP	Client	Video Export Service
MS firewall default "Message Queuing TCP Inbound " Message Queuing UDP				%systemroot%\system32\mqsvc.exe; MS firewall rule added automatically when enabling MSMQ.
ICMP		ICMP		

Server Outbound

Name & Executable	Port	Protocol	Destination	Notes
Everbridge.ControlCenter.AlarmTypes.WindowsService.exe Default\Everbridge.ControlCenter.Driver.ConnectionManager.WindowsService.exe Everbridge.ControlCenter.GIS.WindowsService.exe Everbridge.ControlCenter.RulesEngine.WindowsService.exeipscserver.exe	1433 1434	TCP, UDP	MS SQL	MS SQLPort not specified by Control Center. Check SQL Server installation for specific ports.

Everbridge.ControlCenter.Security.WindowsService.exe	1434	UDP	MS SQL	
Federated Service	9901	TCP	Server	
Everbridge.ControlCenter.Windows Client.exe	7333	TCP	Client	Video Export Service
MS firewall default "Message Queuing TCP Inbound "Message Queuing UDP				%systemroot%\system32\ mqsvc.exe MS firewall rule added automatically when enableing MSMQ.
ICMP		ICMP		

Control Center Client

Client Inbound

Name & Executable	Port	Protocol	Source	Notes
Client Watchdog	8567	TCP	Loopback	
Heartbeat		ICMP	IPSC Server	Ping
MS firewall default "Message Queuing TCP Inbound "Message Queuing UDP		MSMQ		

Client Outbound

Name & Executable	Port	Protocol	Destination	Notes
Everbridge.ControlCenter.WindowsClient.exe Everbridge.ControlCenter.ClientWatchdog.WindowsService.exe Everbridge.ControlCenter.Driver.VideoControlManager.exe Everbridge.ControlCenter.Driver.VideoControlManager64.exe	9004	TCP	Server	Notification Service - SignalR
Everbridge.ControlCenter.WindowsClient.exe	9000, 9003, 9005,	TCP	Server	Core Server, Alarm Types Service, GIS Service, Connection Manager

	9006, 9007, 9008, 9009			service, Security Service
Everbridge.ControlCenter. WindowsClient.exe	7333	TCP	Server	Video Export Service
Everbridge.ControlCenter. WindowsClient.exe	7339	TCP	Server	Reporting
Everbridge.ControlCenter. WindowsClient.exe Everbridge.ControlCenter.Driver. VideoControlManager.exe Everbridge.ControlCenter.Driver. VideoControlManager64.exe	9099, 9100	TCP	CIPSC Server	Connection Manager service
Federated Service	9901, 9902	TCP	Server	
Everbridge.ControlCenter.Driver. VideoControlManager.exe Everbridge.ControlCenter.Driver. VideoControlManager64.exe	554	TCP	Video Edge NVR	VideoEdge - RTSP video streams
Heartbeat		ICMP	Server	Video Edge NVR, MS SQL, Cameras, Ping
MS firewall default "Message Queuing TCP Inbound "Message Queuing UDP				%systemroot%\system32\ mqsvc.exe MS firewall rule added automatically when enabling MSM