Configure the Windows Firewall to Allow SQL Server Access

SQL Server 2014

Firewall systems help prevent unauthorized access to computer resources. If a firewall is turned on but not correctly configured, attempts to connect to SQL Server might be blocked.

To access an instance of the SQL Server through a firewall, you must configure the firewall on the computer that is running SQL Server to allow access. The firewall is a component of Microsoft Windows. You can also install a firewall from another company. This topic discusses how to configure the Windows firewall, but the basic principles apply to other firewall programs.

Note

This topic provides an overview of firewall configuration and summarizes information of interest to a SQL Server administrator. For more information about the firewall and for authoritative firewall information, see the firewall documentation, such as Windows Firewall with Advanced Security and IPsec.

Users familiar with the **Windows Firewall** item in Control Panel and with the Windows Firewall with Advanced Security Microsoft Management Console (MMC) snap-in and who know which firewall settings they want to configure can move directly to the topics in the following list:

- Configure a Windows Firewall for Database Engine Access
- Configure the Windows Firewall to Allow Analysis Services Access
- Configure a Firewall for Report Server Access

In this Topic

This topic has the following sections:

Basic Firewall Information

Firewalls work by inspecting incoming packets, and comparing them against a set of rules. If the rules allow the packet, the firewall passes the packet to the TCP/IP protocol for additional processing. If the rules do not allow the packet, the firewall discards the packet and, if logging is enabled, creates an entry in the firewall logging file.

The list of allowed traffic is populated in one of the following ways:

- When the computer that has the firewall enabled initiates communication, the firewall creates an entry in the list so that the response is allowed. The incoming response is considered solicited traffic and you do not have to configure this.
- An administrator configures exceptions to the firewall. This allows either access to specified programs running on your computer, or access to specified connection ports on your computer. In this case, the computer accepts unsolicited incoming traffic when acting as a server, a listener, or a peer. This is the type of configuration that must be completed to connect to SQL Server.

Choosing a firewall strategy is more complex than just deciding if a given port should be open or closed. When designing a firewall strategy for your enterprise, make sure that you consider all the rules and configuration options available to you. This topic does not review all the possible firewall options. We recommend that you review the following documents:

Windows Firewall with Advanced Security Getting Started Guide

Windows Firewall with Advanced Security Design Guide

Introduction to Server and Domain Isolation

Default Firewall Settings

The first step in planning your firewall configuration is to determine the current status of the firewall for your operating system. If the operating system was upgraded from a previous version, the earlier firewall settings may have been preserved. Also, the firewall settings could have been changed by another administrator or by a Group Policy in your domain.

> **Note**
>
> Turning on the firewall will affect other programs that access this computer, such as file and print sharing, and remote desktop connections. Administrators should consider all applications that are running on the computer before adjusting the firewall settings.

Programs to Configure the Firewall

There are three ways to configure the Windows Firewall settings.

- **Windows Firewall item in Control Panel**

  The **Windows Firewall** item can be opened from Control Panel.

  > **Important**
  >
  > Changes made in the **Windows Firewall** item in Control Panel only affect the current profile. Mobile devices, for example a laptop, should not use the **Windows Firewall** item in Control Panel as the profile might change when it is connected in a different configuration. Then the previously-configured profile will not be in effect. For more information about profiles, see Windows Firewall with Advanced Security Getting Started Guide.

  The **Windows Firewall** item in Control Panel allows you to configure basic options. These include the following:

  - Turning the **Windows Firewall** item in Control Panel on or off
  - Enabling and disabling rules
  - Granting exceptions for ports and programs
  - Setting some scope restrictions

  The **Windows Firewall** item in Control Panel is most appropriate for users who are not experienced in firewall configuration, and who are configuring basic firewall options for computers that are not mobile. You can also open the **Windows Firewall** item in Control Panel from the **run** command by using the following procedure:

  ### To open the Windows Firewall item

  1. On the **Start** menu, click **Run**, and then enter `firewall.cpl`.
  2. Click OK.

- **Microsoft Management Console (MMC)**

  The Windows Firewall with Advanced Security MMC snap-in lets you configure more advanced firewall settings. This snap-in presents most of the firewall options in an easy-to-use manner, and presents all firewall profiles. For more information, see Using the Windows Firewall with Advanced Security Snap-in later in this topic.

- **netsh**

  The **netsh.exe** tool can be used by an administrator to configure and monitor Windows-based computers at a command prompt or using a batch file**.** By using the **netsh** tool, you can direct the context commands you enter to the appropriate helper, and the helper then performs the command. A helper is a Dynamic Link Library (.dll) file that extends the functionality of the **netsh** tool by providing configuration, monitoring, and support for one or more services, utilities, or protocols. All operating systems that support SQL Server have a firewall helper. Windows Server 2008 also has an advanced firewall helper called **advfirewall**. The details of using **netsh** are not discussed in this topic. However, many of the configuration options described can be configured by using **netsh**. For example, run the following script at a command prompt to open TCP port 1433:

  ```
  netsh firewall set portopening protocol = TCP port = 1433 name = SQLPort mode = ENABLE scope = SUBNET profile = CURRENT
  ```

  A similar example using the Windows Firewall for Advanced Security helper:

  ```
  netsh advfirewall firewall add rule name = SQLPort dir = in protocol = tcp action = allow localport = 1433 remoteip = localsubnet profile = DOM
  ```

  For more information about **netsh**, see the following links:

  - How to Use the Netsh.exe Tool and Command-Line Switches
  - How to use the “netsh advfirewall firewall” context instead of the “netsh firewall” context to control Windows Firewall behavior in Windows Server 2008 and in Windows Vista
  - The "netsh firewall" command together with the "profile=all" parameter does not configure the public profile on a Windows Vista-based computer

Ports Used By SQL Server

The following tables can help you identify the ports being used by SQL Server.

Ports Used By the Database Engine

The following table lists the ports that are frequently used by the Database Engine.

Scenario	Port	Comments

SQL Server default instance running over TCP	TCP port 1433	This is the most common port allowed through the firewall. It applies to routine connections to the default installation of the Database Engine, or a named instance that is the only instance running on the computer. (Named instances have special considerations. See Dynamic Ports later in this topic.)
SQL Server named instances in the default configuration	The TCP port is a dynamic port determined at the time the Database Engine starts.	See the discussion below in the section Dynamic Ports. UDP port 1434 might be required for the SQL Server Browser Service when you are using named instances.
SQL Server named instances when they are configured to use a fixed port	The port number configured by the administrator.	See the discussion below in the section Dynamic Ports.
Dedicated Admin Connection	TCP port 1434 for the default instance. Other ports are used for named instances. Check the error log for the port number.	By default, remote connections to the Dedicated Administrator Connection (DAC) are not enabled. To enable remote DAC, use the Surface Area Configuration facet. For more information, see Surface Area Configuration.
SQL Server Browser service	UDP port 1434	The SQL Server Browser service listens for incoming connections to a named instance and provides the client the TCP port number that corresponds to that named instance. Normally the SQL Server Browser service is started whenever named instances of the Database Engine are used. The SQL Server Browser service does not have to be started if the client is configured to connect to the specific port of the named instance.
SQL Server instance running over an HTTP endpoint.	Can be specified when an HTTP endpoint is created. The default is TCP port 80 for CLEAR_PORT traffic and 443 for SSL_PORT traffic.	Used for an HTTP connection through a URL.
SQL Server default instance running over an HTTPS endpoint.	TCP port 443	Used for an HTTPS connection through a URL. HTTPS is an HTTP connection that uses secure sockets layer (SSL).

Service Broker	TCP port 4022. To verify the port used, execute the following query:
	`SELECT name, protocol_desc, port, state_desc`
	`FROM sys.tcp_endpoints`
	`WHERE type_desc = 'SERVICE_BROKER'`
		There is no default port for SQL Server Service Broker, but this is the conventional configuration used in Books Online examples.

Database Mirroring	Administrator chosen port. To determine the port, execute the following query:
	`SELECT name, protocol_desc, port, state_desc FROM sys.tcp_endpoints`
	`WHERE type_desc = 'DATABASE_MIRRORING'`
		There is no default port for database mirroring however Books Online examples use TCP port 7022. It is very important to avoid interrupting an in-use mirroring endpoint, especially in high-safety mode with automatic failover. Your firewall configuration must avoid breaking quorum. For more information, see Specify a Server Network Address (Database Mirroring).

Replication	Replication connections to SQL Server use the typical regular Database Engine ports (TCP port 1433 for the default instance, etc.) Web synchronization and FTP/UNC access for replication snapshot require additional ports to be opened on the firewall. To transfer initial data and schema from one location to another, replication can use FTP (TCP port 21), or sync over HTTP (TCP port 80) or File Sharing. File sharing uses UDP port 137 and 138, and TCP port 139 if it using NetBIOS. File Sharing uses TCP port 445.	For sync over HTTP, replication uses the IIS endpoint (ports for which are configurable but is port 80 by default), but the IIS process connects to the backend SQL Server through the standard ports (1433 for the default instance. During Web synchronization using FTP, the FTP transfer is between IIS and the SQL Server publisher, not between subscriber and IIS.
Transact-SQL debugger	TCP port 135 See Special Considerations for Port 135 The IPsec exception might also be required.	If using Visual Studio, on the Visual Studio host computer, you must also add **Devenv.exe** to the Exceptions list and open TCP port 135. If using Management Studio, on the Management Studio host computer, you must also add **ssms.exe** to the Exceptions list and open TCP port 135. For more information, see Configure the Transact-SQL Debugger.

For step by step instructions to configure the Windows Firewall for the Database Engine, see Configure a Windows Firewall for Database Engine Access.

Dynamic Ports

By default, named instances (including SQL Server Express) use dynamic ports. That means that every time that the Database Engine starts, it identifies an available port and uses that port number. If the named instance is the only instance of the Database Engine installed, it will probably use TCP port 1433. If other instances of the Database Engine are installed, it will probably use a different TCP port. Because the port selected might change every time that the Database Engine is started, it is difficult to configure the firewall to enable access to the correct port number. Therefore, if a firewall is used, we recommend reconfiguring the Database Engine to use the same port number every time. This is called a fixed port or a static port. For more information, see Configure a Server to Listen on a Specific TCP Port (SQL Server Configuration Manager).

An alternative to configuring a named instance to listen on a fixed port is to create an exception in the firewall for a SQL Server program such as **sqlservr.exe** (for the Database Engine). This can be convenient, but the port number will not appear in the **Local Port** column of the **Inbound Rules** page when you are using the Windows Firewall with Advanced Security MMC snap-in. This can make it more difficult to audit which ports are open. Another consideration is that a service pack or cumulative update can change the path to the SQL Server executable which will invalidate the firewall rule.

Note

The following procedure uses the **Windows Firewall** item in Control Panel. The Windows Firewall with Advanced Security MMC snap-in can configure a more complex rule. This includes configuring a service exception which can be useful for providing defense in depth. See Using the Windows Firewall with Advanced Security Snap-in below.

To add a program exception to the firewall using the Windows Firewall item in Control Panel.

1. On the **Exceptions** tab of the **Windows Firewall** item in Control Panel, click **Add a program**.
2. Browse to the location of the instance of SQL Server that you want to allow through the firewall, for example **C:\Program Files\Microsoft SQL Server\MSSQL12.<instance_name>\MSSQL\Binn**, select **sqlservr.exe**, and then click **Open**.
3. Click OK.

For more information about endpoints, see Configure the Database Engine to Listen on Multiple TCP Ports and Endpoints Catalog Views (Transact-SQL).

Ports Used By Analysis Services

The following table lists the ports that are frequently used by Analysis Services.

Feature	Port	Comments
Analysis Services	TCP port 2383 for the default instance	The standard port for the default instance of Analysis Services.
SQL Server Browser service	TCP port 2382 only needed for an Analysis Services named instance	Client connection requests for a named instance of Analysis Services that do not specify a port number are directed to port 2382, the port on which SQL Server Browser listens. SQL Server Browser then redirects the request to the port that the named instance uses.
Analysis Services configured for use through IIS/HTTP (The PivotTable® Service uses HTTP or HTTPS)	TCP port 80	Used for an HTTP connection through a URL.
Analysis Services configured for use through IIS/HTTPS (The PivotTable® Service uses HTTP or HTTPS)	TCP port 443	Used for an HTTPS connection through a URL. HTTPS is an HTTP connection that uses secure sockets layer (SSL).

If users access Analysis Services through IIS and the Internet, you must open the port on which IIS is listening and specify that port in the client connection string. In this case, no ports have to be open for direct access to Analysis Services. The default port 2389, and port 2382, should be restricted together with all other ports that are not required.

For step by step instructions to configure the Windows Firewall for Analysis Services, see Configure the Windows Firewall to Allow Analysis Services Access.

Ports Used By Reporting Services

The following table lists the ports that are frequently used by Reporting Services.

Feature	Port	Comments
Reporting Services Web Services	TCP port 80	Used for an HTTP connection to Reporting Services through a URL. We recommend that you do not use the preconfigured rule **World Wide Web Services (HTTP)**. For more information, see the Interaction with Other Firewall Rules section below.
Reporting Services configured for use through HTTPS	TCP port 443	Used for an HTTPS connection through a URL. HTTPS is an HTTP connection that uses secure sockets layer (SSL). We recommend that you do not use the preconfigured rule **Secure World Wide Web Services (HTTPS)**. For more information, see the Interaction with Other Firewall Rules section below.

When Reporting Services connects to an instance of the Database Engine or Analysis Services, you must also open the appropriate ports for those services. For step-by-step instructions to configure the Windows Firewall for Reporting Services, Configure a Firewall for Report Server Access.

Ports Used By Integration Services

The following table lists the ports that are used by the Integration Services service.

Feature	Port	Comments
Microsoft remote procedure calls (MS RPC) Used by the Integration Services runtime.	TCP port 135 See Special Considerations for Port 135	The Integration Services service uses DCOM on port 135. The Service Control Manager uses port 135 to perform tasks such as starting and stopping the Integration Services service and transmitting control requests to the running service. The port number cannot be changed. This port is only required to be open if you are connecting to a remote instance of the Integration Services service from Management Studio or a custom application.

For step-by-step instructions to configure the Windows Firewall for Integration Services, see Configure a Windows Firewall for Access to the SSIS Service.

Additional Ports and Services

The following table lists ports and services that SQL Server might depend on.

Scenario	Port	Comments
Windows Management Instrumentation For more information about WMI, see WMI Provider for Configuration Management Concepts	WMI runs as part of a shared service host with ports assigned through DCOM. WMI might be using TCP port 135.	SQL Server Configuration Manager uses WMI to list and manage services. We recommend that you use the preconfigured rule group **Windows Management Instrumentation (WMI)**. For more information, see the Interaction with Other Firewall Rules section below.

	See Special Considerations for Port 135	
Microsoft Distributed Transaction Coordinator (MS DTC)	TCP port 135 See Special Considerations for Port 135	If your application uses distributed transactions, you might have to configure the firewall to allow Microsoft Distributed Transaction Coordinator (MS DTC) traffic to flow between separate MS DTC instances, and between the MS DTC and resource managers such as SQL Server. We recommend that you use the preconfigured **Distributed Transaction Coordinator** rule group. When a single shared MS DTC is configured for the entire cluster in a separate resource group you should add sqlservr.exe as an exception to the firewall.
The browse button in Management Studio uses UDP to connect to the SQL Server Browser Service. For more information, see SQL Server Browser Service (Database Engine and SSAS).	UDP port 1434	UDP is a connectionless protocol. The firewall has a setting, which is named UnicastResponsesToMulticastBroadcastDisabled Property of the INetFwProfile Interface which controls the behavior of the firewall with respect to unicast responses to a broadcast (or multicast) UDP request. It has two behaviors: • If the setting is TRUE, no unicast responses to a broadcast are permitted at all. Enumerating services will fail. • If the setting is FALSE (default), unicast responses are permitted for 3 seconds. The length of time is not configurable. in a congested or high-latency network, or for heavily loaded servers, tries to enumerate instances of SQL Server might return a partial list, which might mislead users.
IPsec traffic	UDP port 500 and UDP port 4500	If the domain policy requires network communications to be done through IPsec, you must also add UDP port 4500 and UDP port 500 to the exception list. IPsec is an option using the **New Inbound Rule Wizard** in the Windows Firewall snap-in. For more information, see Using the Windows Firewall with Advanced Security Snap-in below.
Using Windows Authentication with Trusted Domains	Firewalls must be configured to allow authentication requests.	For more information, see How to configure a firewall for domains and trusts.
SQL Server and Windows Clustering	Clustering requires additional ports that are not directly related to SQL Server.	For more information, see Enable a network for cluster use.
URL namespaces reserved in the HTTP Server API (HTTP.SYS)	Probably TCP port 80, but can be configured to other ports. For general information, see Configuring HTTP and HTTPS.	For SQL Server specific information about reserving an HTTP.SYS endpoint using HttpCfg.exe, see About URL Reservations and Registration (SSRS Configuration Manager).

Special Considerations for Port 135

When you use RPC with TCP/IP or with UDP/IP as the transport, inbound ports are frequently dynamically assigned to system services as required; TCP/IP and UDP/IP ports that are larger than port 1024 are used. These are frequently informally referred to as "random RPC ports." In these cases, RPC clients rely on the RPC endpoint mapper to tell them which dynamic ports were assigned to the server. For some RPC-based services, you can configure a specific port instead of letting RPC assign one dynamically. You can also restrict the range of ports that RPC dynamically assigns to a small range, regardless of the service. Because port 135 is used for many services it is frequently attacked by malicious users. When opening port 135, consider restricting the scope of the firewall rule.

For more information about port 135, see the following references:

- Service overview and network port requirements for the Windows Server system
- Troubleshooting RPC Endpoint Mapper errors using the Windows Server 2003 Support Tools from the product CD
- Remote procedure call (RPC)
- How to configure RPC dynamic port allocation to work with firewalls

Interaction with Other Firewall Rules

The Windows Firewall uses rules and rule groups to establish its configuration. Each rule or rule group is generally associated with a particular program or service, and that program or service might modify or delete that rule without your knowledge. For example, the rule groups **World Wide Web Services (HTTP)** and **World Wide Web Services (HTTPS)** are associated with IIS. Enabling those rules will open ports 80 and 443, and SQL Server features that depend on ports 80 and 443 will function if those rules are enabled. However, administrators configuring IIS might modify or disable those rules. Therefore, if you are using port 80 or port 443 for SQL Server, you should create your own rule or rule group that maintains your desired port configuration independently of the other IIS rules.

The Windows Firewall with Advanced Security MMC snap-in allows any traffic that matches any applicable allow rule. So if there are two rules that both apply to port 80 (with different parameters), traffic that matches either rule will be permitted. So if one rule allows traffic over port 80 from local subnet and one rule allows traffic from any address, the net effect is that all traffic to port 80 is permitted regardless of the source. To effectively manage access to SQL Server, administrators should periodically review all firewall rules enabled on the server.

Overview of Firewall Profiles

Firewall profiles are discussed in Windows Firewall with Advanced Security Getting Started Guide in the section **Network location-aware host firewall**. To summarize, the operating systems identify and remember each of the networks to which they connect with regard to connectivity, connections, and category.

There are three network location types in Windows Firewall with Advanced Security:

- Domain. Windows can authenticate access to the domain controller for the domain to which the computer is joined.

- Public. Other than domain networks, all networks are initially categorized as public. Networks that represent direct connections to the Internet or are in public locations, such as airports and coffee shops should be left public.
- Private. A network identified by a user or application as private. Only trusted networks should be identified as private networks. Users will likely want to identify home or small business networks as private.

The administrator can create a profile for each network location type, with each profile containing different firewall policies. Only one profile is applied at any time. Profile order is applied as follows:

1. If all interfaces are authenticated to the domain controller for the domain of which the computer is a member, the domain profile is applied.
2. If all interfaces are either authenticated to the domain controller or are connected to networks that are classified as private network locations, the private profile is applied.
3. Otherwise, the public profile is applied.

Use the Windows Firewall with Advanced Security MMC snap-in to view and configure all firewall profiles. The **Windows Firewall** item in Control Panel only configures the current profile.

Additional Firewall Settings Using the Windows Firewall Item in Control Panel

Exceptions that you add to the firewall can restrict the opening of the port to incoming connections from specific computers or the local subnet. This restriction of the scope of the port opening can reduce how much your computer is exposed to malicious users, and is recommended.

Note
Using the **Windows Firewall** item in Control Panel only configures the current firewall profile.

To change the scope of a firewall exception using the Windows Firewall item in Control Panel

1. In the **Windows Firewall** item in Control Panel, select a program or port on the **Exceptions** tab, and then click **Properties** or **Edit**.
2. In the **Edit a Program** or **Edit a Port** dialog box, click **Change Scope**.
3. Choose one of the following options:
    - **Any computer (including those on the Internet)**

      Not recommended. This will allow any computer that can address your computer to connect to the specified program or port. This setting might be necessary to allow information to be presented to anonymous users on the internet, but increases your exposure to malicious users. Your exposure can be further increased if you enable this setting and also allow Network Address Translation (NAT) traversal, such as the Allow edge traversal option.
    - **My network (subnet) only**

      This is a more secure setting than **Any computer**. Only computers on the local subnet of your network can connect to the program or port.
    - **Custom list:**

    Only computers that have the IP addresses you list can connect. This can be a more secure setting than **My network (subnet) only**, however, client computers using DHCP can occasionally change their IP address. Then the intended computer will not be able to connect. Another computer, which you had not intended to authorize, might accept the listed IP address and then be able to connect. The **Custom list** option might be appropriate for listing other servers which are configured to use a fixed IP address; however, IP addresses might be spoofed by an intruder. Restricting firewall rules are only as strong as your network infrastructure.

Using the Windows Firewall with Advanced Security Snap-in

Additional advanced firewall settings can be configured by using the Windows Firewall with Advanced Security MMC snap-in. The snap-in includes a rule wizard and exposes additional settings that are not available in the **Windows Firewall** item in Control Panel. These settings include the following:

- Encryption settings
- Services restrictions
- Restricting connections for computers by name
- Restricting connections to specific users or profiles
- Edge traversal allowing traffic to bypass Network Address Translation (NAT) routers
- Configuring outbound rules
- Configuring security rules
- Requiring IPsec for incoming connections

To create a new firewall rule using the New Rule wizard

1. On the Start menu, click **Run**, type **WF.msc**, and then click **OK**.
2. In the **Windows Firewall with Advanced Security**, in the left pane, right-click **Inbound Rules**, and then click **New Rule**.
3. Complete the **New Inbound Rule Wizard** using the settings that you want.

Troubleshooting Firewall Settings

The following tools and techniques can be useful in troubleshooting firewall issues:

- The effective port status is the union of all rules related to the port. When trying to block access through a port, it can be helpful to review all the rules which cite the port number. To do this, use the Windows Firewall with Advanced Security MMC snap-in and sort the inbound and outbound rules by port number.
- Review the ports that are active on the computer on which SQL Server is running. This review process includes verifying which TCP/IP ports are listening and also verifying the status of the ports.

  To verify which ports are listening, use the **netstat** command-line utility. In addition to displaying active TCP connections, the **netstat** utility also displays a variety of IP statistics and information.

  ### To list which TCP/IP ports are listening

  1. Open the Command Prompt window.
  2. At the command prompt, type **netstat -n -a**.

     The **-n** switch instructs **netstat** to numerically display the address and port number of active TCP connections. The **-a** switch instructs **netstat** to display the TCP and UDP ports on which the computer is listening.
- The **PortQry** utility can be used to report the status of TCP/IP ports as listening, not listening, or filtered. (With a filtered status, the port might or might not be listening; this status indicates that the utility did not receive a response from the port.) The **PortQry** utility is available for download from the Microsoft Download Center.

See Also

Other Resources
Service overview and network port requirements for the Windows Server system

Community Additions