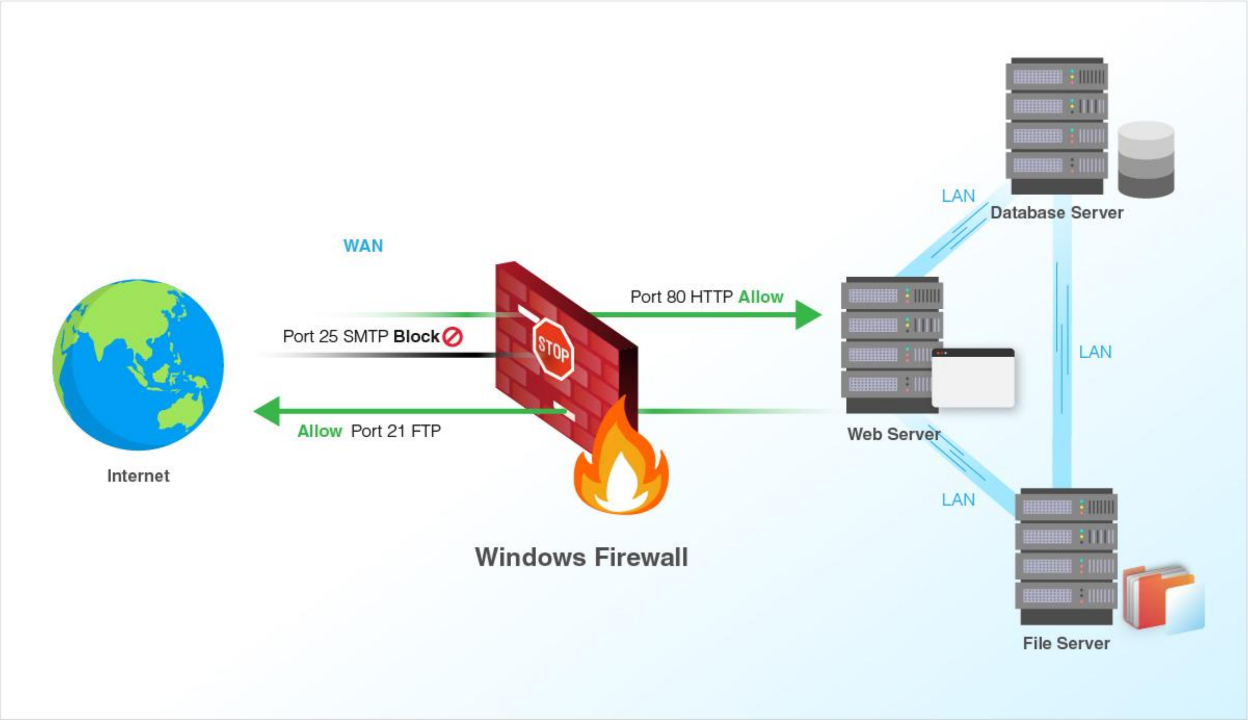
Configuring Windows Server Firewall



Best Practices and general Firewall information helpful links

https://www.csoonline.com/article/3562743/how-to-optimize-windows-firewall-security.html

https://www.alibabacloud.com/help/en/doc-detail/51403.htm

https://community.spiceworks.com/topic/2135198-domain-client-firewall-best-practice

https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-41r1.pdf

https://docs.microsoft.com/en-US/troubleshoot/windows-server/identity/config-firewall-for-ad-domains-and-trusts

Host Based Firewalls and Personal Firewalls Host-based firewalls for servers and personal firewalls for desktop and laptop personal computers (PC) provide an additional layer of security against network-based attacks. These firewalls are software-based, residing on the hosts they are protecting—each monitor and control the incoming and outgoing network traffic for a single host. They can provide more granular protection than network firewalls to meet the needs of specific hosts. Host based firewalls are available as part of server operating systems such as Linux, Windows, Solaris, BSD, and Mac OS X Server, and they can also be installed as third-party add-ons. Configuring a host based firewall to allow only necessary traffic to the server provides protection against malicious activity from all hosts, including those on the same subnet or on other internal subnets not separated by a network firewall. Limiting outgoing traffic from a server may also be helpful in preventing certain malware that infects a host from spreading to other hosts. Host based firewalls usually perform logging and can often be configured to perform address-based and application-based access controls. Many host based firewalls can also act as intrusion prevention systems (IPS) that, after detecting an attack in progress, take actions to thwart the attacker and prevent damage to the targeted host.

Services to restart if wrong firewall profile is active

- Network Location Awareness
- Network List Service (will be prompted to automatically restart if you restart Network Location Awareness first, from services console)

Recommended Inbound Rules for Domain Controller

Client Port(s)	Server Port	Service
49152-65535/UDP	123/UDP	W32Time
49152-65535/TCP	135/TCP	RPC Endpoint Mapper
49152-65535/TCP	464/TCP/UDP	Kerberos password change
49152-65535/TCP	49152-65535/TCP	RPC for LSA, SAM, NetLogon
49152-65535/TCP/UDP	389/TCP/UDP	LDAP
49152-65535/TCP	636/TCP	LDAP SSL
49152-65535/TCP	3268/TCP	LDAP GC
49152-65535/TCP	3269/TCP	LDAP GC SSL
53, 49152-65535/TCP/UDP	53/TCP/UDP	DNS
49152-65535/TCP	49152-65535/TCP	FRS RPC
49152-65535/TCP/UDP	88/TCP/UDP	Kerberos
49152-65535/TCP/UDP	445/TCP	SMB
49152-65535/TCP	49152-65535/TCP	DFSR RPC

Best practices for Windows Firewall with Advanced Security

If attackers can scan for and discover open instance ports, such as port 3389 on a Windows instance and port 22 on a Linux instance, they can initiate attacks on these ports. You can prevent the attacks by modifying the default port or restricting access sources. In this topic, an instance that runs Windows Server is used to describe how to use Windows Firewall with Advanced Security (WFAS) to restrict access from specific IP addresses.

Use MMC to configure Windows Firewall

1. Enable the firewall.

    . Press the shortcut keys `Win+R` to open the **Run** dialog box.
    i. Enter *firewall.cpl* and press the Enter key.
    ii. Click **Turn Windows Firewall on or off** to view the firewall status.
    By default, the firewall is disabled.

2. Check Remote Desktop Protocol (RDP) port 3389.

    . Press the shortcut keys `Win+R` to open the **Run** dialog box.
    i. Enter *wf.msc* and press the Enter key.
    ii. Click **Inbound Rules**. Sort by column **Local Port** to see if any 3389 rule is enabled.

3. Add RDP port 3389 to **Windows Firewall with Advanced Security**.

    . In the Actions section, click **New Rule...**. The **New Inbound Rule Wizard** dialog box appears.
    i. In the **Rule Type** step, select **Port** and click **Next**.
    ii. In the Protocol and Ports step, select `TCP` as the protocol, select Specific local ports and enter **3389** in the field, and then click **Next**.
    iii. Select **Allow the connection** and click **Next**.
    iv. Use the default settings and click **Next**.
    v. Enter a rule name. In this example, RemoteDesktop is used. Click **Finish**.

4. Configure the scope.

    . Right-click the created RemoteDesktop inbound rule and click **Properties**
    i. On the **Scope** tab, select **These IP addresses:** in the **Remote IP address** section, add one or more IP addresses or CIDR blocks, and then click **OK**.
    Notice After the parameters on the Scope tab are configured, only IP addresses that you have specified in the Remote IP address section can access to the Windows instance.

5. Add an IP address to the Remote IP address section. In this example, 192.168.144.10 is added. Then, click **OK**.

Using CLI to configure Windows Firewall

You can also run the **netsh** command in CLI to configure WFAS. The following section provides examples of the **netsh** command:

- Export the firewall configuration file.

```
netsh advfirewall export c:\adv.pol
```

- Import the firewall configuration file.

```
netsh advfirewall import c:\adv.pol
```

- Restore the default settings of the firewall.

```
netsh advfirewall reset
```

- Disable the firewall.

```
netsh advfirewall set allprofiles state off
```

- Enable the firewall.

```
netsh advfirewall set allprofiles state on
```

- Delete the ftp rule.

```
netsh advfirewall firewall delete rule name=ftp
```

- Delete all inbound rules for the local port 80.

```
netsh advfirewall firewall delete rule name=all protocol=tcp localport=80
```

- Add an inbound rule for the remote desktop to allow traffic from port 3389.

```
netsh advfirewall firewall add rule name=”Remote Desktop (Added VIA netsh)” protocol=TCP dir=in localport=3389 action=allow RemoteIP=192.168.144.10
```

Using PowerShell to create/modify Windows Firewall

- Create an inbound firewall rule to only allow RDP access from a specified IP address

```
New-NetFirewallRule -DisplayName "Remote Desktop (Added via PowerShell)" -Direction Inbound -Action Allow -Protocol TCP -LocalPort 3389 -RemoteAddress 192.168.144.10
```

- Create an inbound firewall rule to allow TestNav Proctor Cache server access

```
New-NetFirewallRule -DisplayName "TestNav ProctorCache" -Direction Inbound -Action Allow -Protocol TCP -LocalPort 4480-4481
```

- Create an outbound firewall rule to block all of the traffic from the local computer that originates on TCP port 80.

```
New-NetFirewallRule -DisplayName "Block Outbound Port 80" -Direction Outbound -LocalPort 80 -Protocol TCP -Action Block
```

- Create a firewall rule that blocks all inbound traffic from all WINS servers.

```
New-NetFirewallRule -DisplayName "Block WINS" -Direction Inbound -Action Block -RemoteAddress WINS
```

- Create an inbound firewall rule that allows traffic for the Windows Messenger program only from computers on the same subnet as the local computer.

```
New-NetFirewallRule -DisplayName "Allow Messenger" -Direction Inbound -Program "C:\Program Files (x86)\Messenger\msmsgs.exe" -RemoteAddress LocalSubnet -Action Allow
```

- Create a firewall rule that allows inbound Windows Messenger network traffic only if the connection from the remote computer is authenticated by using a separate IPsec rule

```
New-NetFirewallRule -DisplayName "Allow Authenticated Messenger" -Direction Inbound -Program "C:\Program Files (x86)\Messenger\msmsgs.exe" -Authentication Required -Action Allow
```

- Create a firewall rule that allows all of the network traffic from computers that are members of a specific computer group, and only from users that are members of a specific user group. Both memberships must be confirmed by authentication using a separate connection security rule.

```
New-NetFirewallRule -DisplayName "Allow Only Specific Computers and Users" -Direction Inbound -RemoteMachine "D:(A;;CC;;;SIDforMachineGroupAccount)"  -RemoteUser "D:(A;;CC;;;SIDforUserGroupAccount)"  -Action Allow -Authentication Required
```

- Create firewall rules that block all of the wireless network traffic.

```
New-NetFirewallRule -DisplayName "Block Wireless In" -Direction Inbound -InterfaceType Wireless -Action Block
```

```
New-NetFirewallRule -DisplayName "Block Wireless Out" -Direction Outbound -InterfaceType Wireless -Action Block
```

- Create a firewall rule to allow TCP traffic addressed to port 12345 and the range of ports 5000-5020 to a specific application from the computers on the remote side of an edge (NAT) device, using the Teredo IPv6 interface.

```
New-NetFirewallRule -DisplayName "Allow TCP 12345 and 5000-5020 over Teredo" -Direction Inbound -Action Allow -EdgeTraversalPolicy Allow -Protocol TCP -LocalPort 12345,5000-5020 -Program "C:\Program Files (x86)\TestIPv6App.exe"
```