Dr. D. Y. Patil Pratishthan’s

Institute for Advanced Computing &Software Development

IACSD

Operating System

INDEX

1. Windows Deployment... ………… …..1

2. Overview of Disk Management 5

3. Introduction of DNS Terminology, Components, and Concepts.....................7

4. Dynamic Host Configuration Protocol (DHCP) 24

5. Introduction of Active Directory Domain Services...................................36

6. Active Directory Certificate Services... 57

7. Active Directory Federation Service (ADFS)... 61

8. Microsoft System Center Configuration Manager (SCCM)........................66

9. Windows and containers.. 75

10. Network Load Balancing. 80

11. VPN 85

12. DFS 96

13. NPS ………………………………………………………………………………100

14. Powershell ……………………………………………………………………103

Windows Deployment

Windows Deployment Services (WDS) is a service role that allows you to deploy the Windows operating system to a PXE-enabled client. You can install a Windows operating system on a PXE-capable client without the need for physical media. Besides, it allows you to deploy a Windows operating system to multiple clients at the same time. In this way, reducing the entire installation time. WDS makes your life easier. You do not have to go to another room in the same room to deploy the operating system with the necessary software and drivers. Administrators can schedule schedules for deployment. The next day when you come in and log in to your computer, you will be installing a new operating system. In addition, drivers and software have been installed.

WDS Functions:

• Allows you to install a network-based operating system.
• Facilitate the process of deployment.
• Supports deployment of computers without any installed operating system.
• Provide end-to-end deployment solutions for clients and server computers.
• Use existing technologies like Windows PE, Windows Image File (.wim) and virtual hard disk (.vhd and .vhdx) image files, and image-based deployment.

Hardware Requirements:

- RAM: minimum of 4GB
- Processor: 64-bit processor
- Hard Drive Space: At least 10GB Or Depend On OS ISO and Software’s Size.

Prerequisites to Configure WDS Server:

Before configuring the WDS server, your server must be sure that the following requirements are met.

1) Changed Windows Server Hostname.
2) Turn off Windows Server Firewall.
3) Set Static IP on Windows Server.

Follow the below Steps To Install and Configure WDS Server (Windows Deployment Services) On Windows Server 2016:

Installation of WDS:

Step 1. Open **Server Manager**.

Step 2. Click **Add Role and Features**.

Step 3. Read the important requirements and try to meet them (this includes a strong password for the administrator account, stable IP configuration, and installation of the latest security updates). Click next to continue.

Step 4. Choose **Roll-based or Feature-based installation** and click **Next**.

Step 5. Select the destination server for **WDS** from the server pool and click Next.

Step 6. Select Windows Deployment Services from the server role. When you check that option, a new window will pop up. Click Add Features.

Step 7. Click Next two times.

Step 8. Click Next (Leave default selection of both deployment and transport server)

Step 9. Click **Install**

Step 10. Wait for the installation to complete. It may take several minutes to **close**.

Configuration of WDS:

Step1. Open **Server Manager** Dashboard. **Tools** ->click **Windows Deployment Services.**

Step 2.Right-click on your **WDS server** and then click **Configure Server**.

Step 3. Read the pre-requisites and click **next**.

Step 4. Choose "**Integrate with Active Directory**", If you have configured AD on my network. If you do not have AD configured and you are configuring WDS in standalone mode, select the **standalone server**. Click **Next**

Step 5. Provide a route to your NTFS drive where you want to store a boot image, install images, PXE boot files and WDS management tools. Click **Next.**

Step 6. I have configured DHCP on the same WDS server. So check both boxes and click **next**.

Step 7. Choose '**Respond to all client computers**' (known and unknown).

Step 8. Wait for the wizard to finish (this may take a few minutes to complete).

Step 9. Click **Finish**.

Add Install Images to WDS:

When you configure Windows Deployment Services on your server, the next step is to add an image to your WDS to the client machine. There are two types of images that you need to add. There is an install.wim (actual Windows installation files) and the other is boot.wim (used to boot client machines).

Step 1. Open windows deployment services console.

Step 2. Expand your server.

Step 3.Right-click on **Install Images** and then click **Add Install Image**.

Step 4. Provide the image group name and then click **Next**.

Step 5. **Browse** to the source folder (located on your Windows installation CD/DVD or local hard drive).

Step 6. Choose the **install.wim** file and click **Next**.

Step 7. Click Next.

Step 8.Select Image and Click **Next**.

Step 9. Check **Summary** and Click **Next**.

Step 10. Wait for the file to be copied. (This can take several minutes to complete).

Steps 11. Click On **Finish**.

Steps 12. As you can see on the output above we have successfully added the **install.wim** Image File.

Add Boot Image to WDS:

Step 1.In the Windows deployment services console, right-click **boot image**.

Step 2. Click on the **Add boot image**.

Step 3. Browse to source folder of your CD/DVD and locate **boot.wim.**

Step 4. Click Next.

Step 5.Rename Image File and Click **Next**.

Step 6. Check **Summary** and Click **Next**.

Step 7. Wait for the file to be copied. (This can take several minutes to complete).

Steps 8. Click On Finish.

Steps 9. As you can see on the output above we have successfully added the **boot.wim** Image File.

Configure DHCP Scope Options:

Enabling bare-metal client systems on Windows Deployment Services (WDS) PXE-boot to kickstart the process of deploying Windows PE on a client system and running the Windows system. In some WDS environments, you might want to configure the following DHCP options to direct your PXE clients to the correct network boot file.

Steps 1. Now go to **Server Manager** and **right-click** on the DHCP server and click on **DHCP Manager**.

Steps 2. Right, click on **Scope Options** and click **Configure Options.**

Steps 3. Add Boot **Server Hostname or IP** in cope Options.**66 = DNS name of the WDS server**

Steps 4. Option 67 Where can you get the boot file name to configure? This name will be something similar to **boot\x64\wdsnbp.com** where this route corresponds to the Remount folder on the WDS server.

67 = boot file name

Steps 5. Finally, we can start the WDS server. The WWS server has not been configured only. **Right-click** on the WDS server.hover your mouse over **all the tasks**, then click **Start**. Soon after the WDS server starts. You can see on the output above we have successfully started the WDS server.

Deploy Windows with WDS:

Steps 1. The client will want to install the Windowsos of the system. First of all, **connect** the first **WDS server** to the **client computer** with **LAN cable**. To confirm the same you can use the **DHCP Address Lease**.

Steps 2. Boot Client System With Network.

Steps 3. Check WDS IP and Hostname and press F12 to Deploy Windows.The machine is booting via network and windows os files loading starts.

Steps 4. Select your location and Keyboard Language.

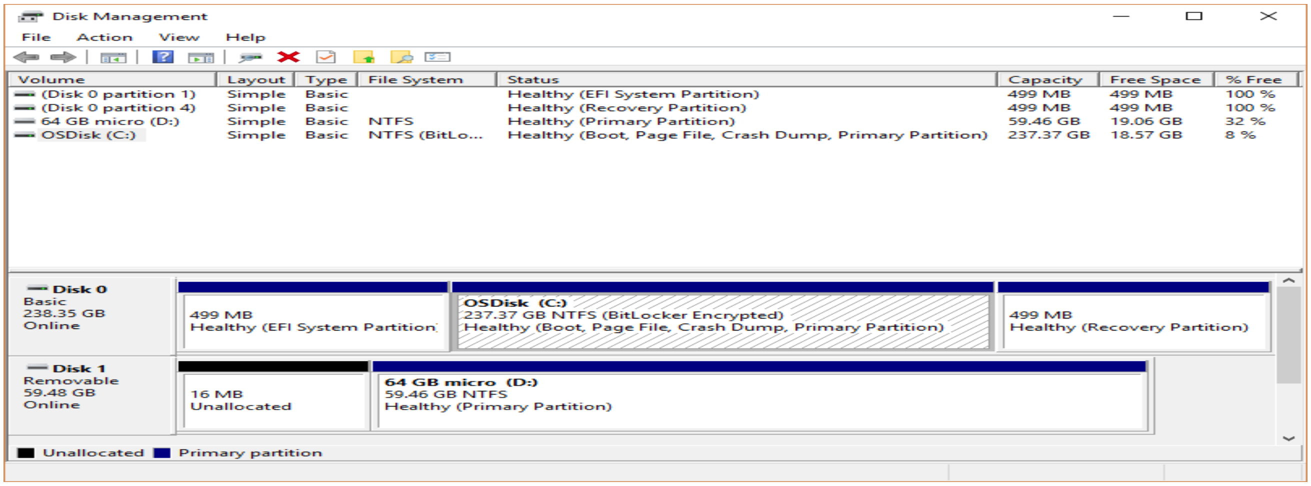
Steps 5. Enter WDS Credentials and connect WDS.

Overview of Disk Management

Applies To: Windows 10, Windows 8.1, Windows 7, Windows Server (Semi-Annual Channel), Windows Server 2019, Windows Server 2016, Windows Server 2012 R2, Windows Server 2012

Disk Management is a system utility in Windows that enables you to perform advanced storage tasks. Here are some of the things Disk Management is good for:

- To setup a new drive, see Initializing a new drive.
- To extend a volume into space that's not already part of a volume on the same drive, see Extend a basic volume.
- To shrink a partition, usually so that you can extend a neighboring partition, see Shrink a basic volume.
- To change a drive letter or assign a new drive letter, see Change a drive letter.



Volume	Layout	Type	File System	Status	Capacity	Free Space	% Free
(Disk 0 partition 1)	Simple	Basic		Healthy (EFI System Partition)	499 MB	499 MB	100 %
(Disk 0 partition 4)	Simple	Basic		Healthy (Recovery Partition)	499 MB	499 MB	100 %
64 GB micro (D:)	Simple	Basic	NTFS	Healthy (Primary Partition)	59.46 GB	19.06 GB	32 %
OSDisk (C:)	Simple	Basic	NTFS (BitLo...	Healthy (Boot, Page File, Crash Dump, Primary Partition)	237.37 GB	18.57 GB	8 %

Disk 0
Basic
238.35 GB
Online

499 MB
Healthy (EFI System Partition)

OSDisk (C:)
237.37 GB NTFS (BitLocker Encrypted)
Healthy (Boot, Page File, Crash Dump, Primary Partition)

499 MB
Healthy (Recovery Partition)

Disk 1
Removable
59.48 GB
Online

16 MB
Unallocated

64 GB micro (D:)
59.46 GB NTFS
Healthy (Primary Partition)

■ Unallocated ■ Primary partition

Tip

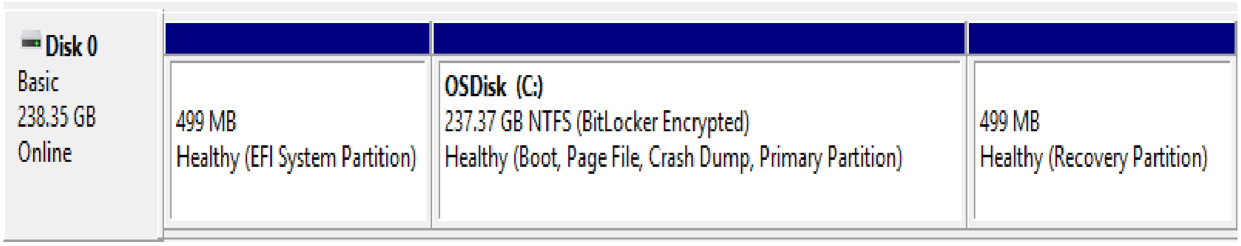
If you get an error or something doesn't work when following these procedures, take a peek at the **Troubleshooting Disk Management** topic. If that doesn't help - don't panic! There's a ton of info on the **Microsoft community** site - try searching the **Files, folders, and storage** section, and if you still need help, post a question there and Microsoft or other members of the community will try to help.

Here are some common tasks you might want to do but that use other tools in Windows:

- To free up disk space, see Free up drive space in Windows 10.
- To defragment your drives, see Defragment your Windows 10 PC.
- To take multiple hard drives and pool them together, similar to a RAID, see Storage Spaces.

About those extra recovery partitions

In case you're curious (we've read your comments!), Windows typically includes three partitions on your main drive (usually the C:\ drive):



Disk 0
Basic
238.35 GB
Online

499 MB
Healthy (EFI System Partition)

OSDisk (C:)
237.37 GB NTFS (BitLocker Encrypted)
Healthy (Boot, Page File, Crash Dump, Primary Partition)

499 MB
Healthy (Recovery Partition)

- **EFI system partition** - This is used by modern PCs to start (boot) your PC and your operating system.

- **Windows operating system drive (C:)** - This is where Windows is installed, and usually where you put the rest of your apps and files.

- **Recovery partition** - This is where special tools are stored to help you recover Windows in case it has trouble starting or runs into other serious issues.

Although Disk Management might show the EFI system partition and the recovery partition as 100% free, it's lying. These partitions are generally pretty full with really important files your PC needs to operate properly. It's best to just leave them alone to do their jobs starting your PC and helping you recover from problems.

Introduction of DNS Terminology, Components, and Concepts

Domain Name System

The domain name system, more commonly known as "DNS" is the networking system in place that allows us to resolve human-friendly names (Fully Qualified Domain Name) to unique IP addresses.

Domain Name

A domain name is the human-friendly name that we are used to associating with an internet resource. For instance, "google.com" is a domain name. Some people will say that the "google" portion is the domain, but we can generally refer to the combined form as the domain name.
The URL "google.com" is associated with the servers owned by Google Inc. The domain name system allows us to reach the Google servers when we type "google.com" into our browsers.

Top Level Domain (TLD)

TLD refers to the last part of a domain name. For example, the .com in amazon.com is the Top Level Domain. The most common TLDs include .com, .net, org, and .info. Country code TLDs represent specific geographic locations. For example: .in represents India. Here are some more examples:

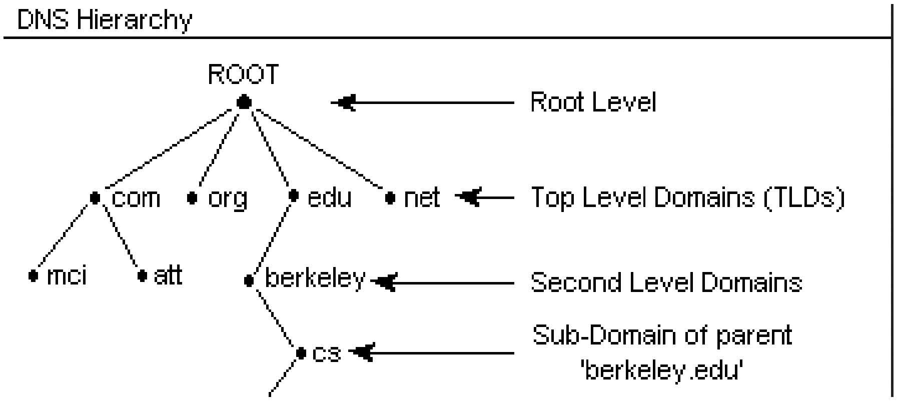
- **com** – Commercial businesses.
- **gov** – U.S. government agencies.
- **edu** – Educational institutions such as universities.
- **org** – Organizations (mostly non-profit).
- **mil** – Military.
- **net** – Network organizations.
- **eu** – European Union.

Second Level Domain

This is the part of a domain name which comes right before the TLD—amazon.com—for example.

Sub Domain

A subdomain can be created to identify unique content areas of a web site. For example, the aws of aws.amazon.com.



Hosts

Within a domain, the domain owner can define individual hosts, which refer to separate computers or services accessible through a domain. For instance, most domain owners make their web servers accessible through the bare domain (example.com) and also through the “host” definition “www” (www.example.com).

Fully Qualified Domain Name

A fully qualified domain name, often called FQDN, is what we call an absolute domain name. Domains in the DNS system can be given relative to one another, and as such, can be somewhat ambiguous. A FQDN is an absolute name that specifies its location in relation to the absolute root of the domain name system. This means that it specifies each parent domain including the TLD. A proper FQDN ends with a dot, indicating the root of the DNS hierarchy. An example of a FQDN is “mail.google.com.”. Sometimes

Software that calls for FQDN does not require the ending dot, but the trailing dot is required to conform to ICANN standards.

Name Server

A name server is a computer designated to translate domain names into IP addresses. These servers do most of the work in the DNS system. Since the total number of domain translations is too much for any one server, each server may redirect request to other name servers or delegate responsibility for a subset of subdomains they are responsible for.

Name servers can be “authoritative”, meaning that they give answers to queries about domains under their control. Otherwise, they may point to other servers, or serve cached copies of other name servers’ data.

Zone File

A zone file is a simple text file that contains the mappings between domain names and IP addresses. This is how the DNS system finally finds out which IP address should be contacted when a user requests a certain domain name.

Zone files reside in name servers and generally define the resources available under a specific domain, or the place that one can go to get that information.

Types of Zones

Active Directory Integrated Zones

Active Directory Integrated Zones stores its zone data in Active Directory. Integrated zones can be replicated to all domain controllers in the domain and forest. Active Directory integrated zones use multi-master replication, this means any domain controller running the DNS server service can write updates to the zone for which they are authoritative.

Advantages of Active Directory integrated Zones

- Replication is faster, more secure and efficient.
- Better redundancy due to zone data being copied to all Domain Controllers
- Improved Security if secure dynamic update is enabled
- No need to schedule or manage zone transfers

Primary Zone

This is the main zone and has a read/write copy of the zone data. All changes to the zone are made in the primary zone and are replicated to the secondary zones.
The zone data is stored in a text file located in this folder c:\windows\system32\DNS on the Windows server running DNS.

Secondary Zone

A secondary Zone is a read-only copy of the primary zone. This zone cannot process updates and can only retrieve updates from the primary zone. This zone can answer DNS name resolution queries from clients nodes, this helps reduce the workload on the primary zone. Secondary zones cannot be active directory integrated.

Stub Zone

Stub zones are like a secondary zone but only stores partial zone data. These zones are useful to help reduce zone transfers by passing the requests to authoritative servers. These zones only contain the SOA, NS and A records.

Forward Lookup Zone

A forward lookup zone provides hostname to IP address resolution.
When you access a system or website by its hostname such as mcirosoft.com DNS checks the forward lookup zone for the IP information related to the hostname.

Reverse Lookup Zone

Reverse lookup zones resolve IP addresses into hostnames.

For example, when you look up the IP 8.8.8.8 it resolves to google-public-dns-a.google.com. A reverse DNS record had to be created for the IP to resolve to the hostname.

Reverse lookup zones are not as common as forwarding lookups and in most cases are not needed.

Records

Within a zone file, records are kept. In its simplest form, a record is basically a single mapping between a resource and a name. These can map a domain name to an IP address, define the name servers for the domain, define the mail servers for the domain, etc.

How DNS Works

Now that you are familiar with some of the terminology involved with DNS, how does the system actually work?

Root Servers

As we said above, DNS is, at its core, a hierarchical system. At the top of this system is what are known as “root servers”. These servers are controlled by various organizations and are delegated authority by ICANN (Internet Corporation for Assigned Names and Numbers).

There are currently 13 root servers in operation. However, as there are an incredible number of names to resolve every minute, each of these servers is actually mirrored. The interesting thing about this set up is that each of the mirrors for a single root server share the same IP address. When requests are made for a certain root server, the request will be routed to the nearest mirror of that root server.What do these root servers do? Root servers handle requests for information about Top-level domains. So if a request comes in for something a lower-level name server cannot resolve, a query is made to the root server for the domain.

The root servers won't actually know where the domain is hosted. They will, however, be able to direct the requester to the name servers that handle the specifically requested top-level domain.
So if a request for "www.wikipedia.org" is made to the root server, the root server will not find the result in its records. It will check its zone files for a listing that matches "www.wikipedia.org". It will not find one.
It will instead find a record for the "org" TLD and give the requesting entity the address of the name server responsible for "org" addresses.

TLD Servers

The requester then sends a new request to the IP address (given to it by the root server) that is responsible for the top-level domain of the request.
So, to continue our example, it would send a request to the name server responsible for knowing about "org" domains to see if it knows where "www.wikipedia.org" is located.
Once again, the requester will look for "www.wikipdia.org" in its zone files. It will not find this record in its files.
However, it will find a record listing the IP address of the name server responsible for "wikipedia.org". This is getting much closer to the answer we want.

Domain-Level Name Servers

At this point, the requester has the IP address of the name server that is responsible for knowing the actual IP address of the resource. It sends a new request to the name server asking, once again, if it can resolve "www.wikipedia.org".

The name server checks its zone files and it finds that it has a zone file associated with "wikipedia.org". Inside of this file, there is a record for the "www" host. This record tells the IP address where this host is located. The name server returns the final answer to the requester.

What is a Resolving Name Server?

In the above scenario, we referred to a “requester”. What is the requester in this situation?

In almost all cases, the requester will be what we call a “resolving name server” A resolving name server is one configured to ask other servers questions. It is basically an intermediary for a user which caches previous query results to improve speed and knows the addresses of the root servers to be able to “resolve” requests made for things it doesn’t already know about.

Basically, a user will usually have a few resolving name servers configured on their computer system. The resolving name servers are usually provided by an ISP or other organizations. For instance Google provides resolving DNS servers that you can query. These can be either configured in your computer automatically or manually.

When you type a URL in the address bar of your browser, your computer first looks to see if it can find out locally where the resource is located. It checks the “hosts” file on the computer and a few other locations. It then sends the request to the resolving name server and waits back to receive the IP address of the resource.

The resolving name server then checks its cache for the answer. If it doesn’t find it, it goes through the steps outlined above.

Resolving name servers basically compress the requesting process for the end user. The clients simply have to know to ask the resolving name servers where a resource is located and be confident that they will investigate and return the final answer.

Zone Files

We mentioned in the above process the idea of “zone files” and “records”.

Zone files are the way that name servers store information about the domains they know about. Every domain that a name server knows about is stored in a zone file. Most requests coming to the average name server are not something that the server will have zone files for.

If it is configured to handle recursive queries, like a resolving name server, it will find out the answer and return it. Otherwise, it will tell the requesting party where to look next.

The more zone files that a name server has, the more requests it will be able to answer authoritatively.

A zone file describes a DNS “zone”, which is basically a subset of the entire DNS naming system. It generally is used to configure just a single domain. It can contain a number of records which define where resources are for the domain in question.

The zone’s $ORIGIN is a parameter equal to the zone’s highest level of authority by default.So if a zone file is used to configure the “example.com.” domain, the $ORIGIN would be set to example.com..

This is either configured at the top of the zone file or it can be defined in the DNS server’s configuration file that references the zone file. Either way, this parameter describes what the zone is going to be authoritative for.Similarly, the $TTL configures the “time to live” of the information it provides. It is basically a timer. A caching name server can use previously queried results to answer questions until the TTL value runs out.

Record Types

Within the zone file, we can have many different record types. We will go over some of the more common (or mandatory types) here.

SOA Records

The Start of Authority, or SOA, record is a mandatory record in all zone files. It must be the first real record in a file (although $ORIGIN or $TTL specifications may appear above). It is also one of the most complex to understand. The start of authority record looks something like this:

```
domain.com.  IN SOA ns1.domain.com. admin.domain.com. (
                              12083 ; serial number
                              3h      ; refresh interval
                              30m     ; retry interval
                              3w      ; expiry period
                              1h      ; negative TTL
)
```

Let’s explain what each part is for:

- **domain.com.**: This is the root of the zone. This specifies that the zone file is for the domain.com. domain. Often, you’ll see this replaced with @, which is just a placeholder that substitutes the contents of the $ORIGIN variable we learned about above.
- **IN SOA**: The “IN” portion means internet (and will be present in many records). The SOA is the indicator that this is a Start of Authority record.
- **ns1.domain.com.**: This defines the primary name server for this domain. Name servers can either be primary or secondary, and if dynamic DNS is configured one server needs to be a “primary”, which goes here. If you haven’t configured dynamic DNS, then this is just one of your primary name servers.
- **admin.domain.com.**: This is the email address of the administrator for this zone. The “@” is replaced with a dot in the email address. If the name portion of the email address normally has a dot in it, this is replace with a “" in this part (your.name@domain.com becomes your\name.domain.com).
- **12083**: This is the serial number for the zone file. Every time you edit a zone file, you must increment this number for the zone file to propagate correctly. Secondary servers will check if the primary server’s serial number for a zone is larger than the one they have on their system. If it is, it requests the new zone file, if not, it continues serving the original file.
- **3h**: This is the refresh interval for the zone. This is the amount of time that the secondary will wait before polling the primary for zone file changes.
- **30m**: This is the retry interval for this zone. If the secondary cannot connect to the primary when the refresh period is up, it will wait this amount of time and retry to poll the primary.
- **3w**: This is the expiry period. If a secondary name server has not been able to contact the primary for this amount of time, it no longer returns responses as an authoritative source for this zone.
- **1h**: This is the amount of time that the name server will cache a name error if it cannot find the requested name in this file.

A and AAAA Records

Both of these records map a host to an IP address. The "A” record is used to map a host to an IPv4 IP address, while “AAAA” records are used to map a host to an IPv6 address.

The general format of these records is this:

```
host    IN    A      IPv4_address
host    IN    AAAA   IPv6_address
```

So since our SOA record called out a primary server at “ns1.domain.com”, we would have to map this to an address to an IP address since “ns1.domain.com” is within the “domain.com” zone that this file is defining.

CNAME Records

CNAME records define an alias for canonical name for your server (one defined by an A or AAAA record).

For instance, we could have an A name record defining the “server1” host and then use the “www” as an alias for this host:

```
server1    IN  A      111.111.111.111
www        IN  CNAME  server1
```

Be aware that these aliases come with some performance losses because they require an additional query to the server. Most of the time, the same result could be achieved by using additional A or AAAA records.

One case when a CNAME is recommended is to provide an alias for a resource outside of the current zone.

MX Records

MX records are used to define the mail exchanges that are used for the domain. This helps email messages arrive at your mail server correctly.

Unlike many other record types, mail records generally don’t map a host to something, because they apply to the entire zone. As such, they usually look like this:

```
IN  MX  10  mail.domain.com.
```

Also note that there is an extra number in there. This is the preference number that helps computers decide which server to send mail to if there are multiple mail servers defined. Lower numbers have a higher priority.

The MX record should generally point to a host defined by an A or AAAA record, and not one defined by a CNAME.So, let’s say that we have two mail servers. There would have to be records that look something like this:

```
IN MX 10 mail1.domain.com.
IN  MX  50  mail2.domain.com.
            mail1   IN  A      111.111.111.111
            mail2   IN  A      222.222.222.222
```

NS Records

This record type defines the name servers that are used for this zone.

You may be wondering, “if the zone file resides on the name server, why does it need to reference itself? Part of what makes DNS so successful is its multiple levels of caching. One reason for defining name servers within the zone file is that the zone file may be actually being served from a cached copy on another name server. There are other reasons for needing the name servers defined on the name server itself, but we won’t go into that here.

Like the MX records, these are zone-wide parameters, so they do not take hosts either. In general, they look like this:

```
IN  NS     ns1.domain.com.
IN  NS     ns2.domain.com.
```

You should have at least two name servers defined in each zone file in order to operate correctly if there is a problem with one server. Most DNS server software considers a zone file to be invalid if there is only a single name server.

As always, include the mapping for the hosts with A or AAAA records:

```
IN  NS     ns1.domain.com.
IN  NS     ns2.domain.com.
            ns1     IN A      111.222.111.111
            ns2     IN A      123.211.111.233
```

There are quite a few other record types you can use, but these are probably the most common types that you will come across.

PTR Records

The PTR records are used define a name associated with an IP address. PTR records are the inverse of an A or AAAA record. PTR records are unique in that they begin at the .arpa root and are delegated to the owners of the IP addresses. The Regional Internet Registries (RIRs) manage the IP address delegation to organization and service providers. The Regional Internet Registries include APNIC, ARIN, RIPE NCC, LACNIC, and AFRINIC.

Here is an example of a PTR record for 111.222.333.444 would look like:

444.333.222.111.in-addr.arpa. 33692 IN PTR host.example.com.

This example of a PTR record for an IPv6 address shows the *nibble* format of the reverse of Google's IPv6 DNS Server 2001:4860:4860::8888.

8.8.8.8.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.6.8.4.0.6.8.4.1.0.0.2.ip6.arpa. 86400IN PTR google-public-dns-a.google.com.

CAA Records

CAA records are used to specify which Certificate Authorities (CAs) are allowed to issue SSL/TLS certificates for your domain. As of September 8, 2017 all CAs are required to check for these records before issuing a certificate. If no record is present, any CA may issue a certificate. Otherwise, only the specified CAs may issue certificates. CAA records can be applied to single hosts, or entire domains. Example CAA record follows:

example.com. IN CAA 0 issue "letsencrypt.org"

The host, IN, and record type (CAA) are common DNS fields. The CAA-specific information above is the 0 issue "letsencrypt.org" portion. It is made up of three parts: flags (0), tags (issue), and values ("letsencrypt.org").

- **Flags** are an integer which indicates how a CA should handle tags it doesn't understand. If the flag is 0, the record will be ignored. If 1, the CA must refuse to issue the certificate.
- **Tags** are strings that denote the purpose of a CAA record. Currently they can be issue to authorize a CA to create certificates for a specific hostname, issue wild to authorize wildcard certificates, or iodef to define a URL where CAs can report policy violations.

- **Values** are a string associated with the record's **tag**. For issue and issuewild this will typically be the domain of the CA you're granting the permission to. For iodef this may be the URL of a contact form, or a mailto: link for email feedback.

DNS Queries

The major task carried out by a DNS server is to respond to queries (questions) from a local or remote resolver or other DNS acting on behalf of a resolver. A query would be something like 'what is the IP address of fred.example.com'.

A DNS server may receive such a query for any domain. DNS servers may be configured to be authoritative for some domains, slaves for others, forward queries or other combinations. Most of the queries that a DNS server will receive will be for domains for which it has no knowledge, that is, for which it has no local zone files. DNS software typically allows the name server to respond in different ways to queries about which it has no knowledge. There are three types of queries defined for DNS:

1. **A recursive query-** the complete answer to the question is always returned. DNS servers are not required to support recursive queries.
2. **An Iterative (or non-recursive) query-** where the complete answer MAY be returned or a **referral** provided to another DNS. All DNS servers must support Iterative queries.
3. **An Inverse query -** where the user wants to know the domain name given a resource record. Reverse queries were poorly supported, very infrequent and are now obsolete.

Note: The process called Reverse Mapping (returns a host name given an IP address) does not use Inverse queries but instead uses Recursive and Iterative (non-recursive) queries using the special domain name IN-ADDR.ARPA.

Historically reverse IPv4 mapping was not mandatory. Many systems however now use reverse mapping for security and simple authentication schemes (especially mail servers) so proper implementation and maintenance is now practically essential. IPv6 originally mandated reverse mapping but, like a lot of the original IPv6 mandates, has now been rolled-back.

1. A recursive query

A recursive query is one where the DNS server will fully answer the query (or give an error). DNS servers are not required to support recursive queries and both the resolver (or another DNS acting recursively on behalf of another resolver) negotiate use of recursive service using a bit (RD) in the query header.

There are three possible responses to a recursive query:

1. The answer to the query accompanied by any CNAME records (aliases) that may be useful. The response will indicate whether the data is authoritative or cached.
2. An error indicating the domain or host does not exist (NXDOMAIN). This response may also contain CNAME records that pointed to the non-existing host.
3. An temporary error indication - for instance, can't access other DNS's due to network error etc..

In a recursive query a DNS Resolver will, on behalf of the client (stub-resolver), chase the trail of DNS system across the universe to get the real answer to the question. The journey of a simple query such as 'what is the IP address of www.example.com' to a DNS Resolver which supports recursive queries but is not authoritative for example.com is shown in Diagram 1-3 below:

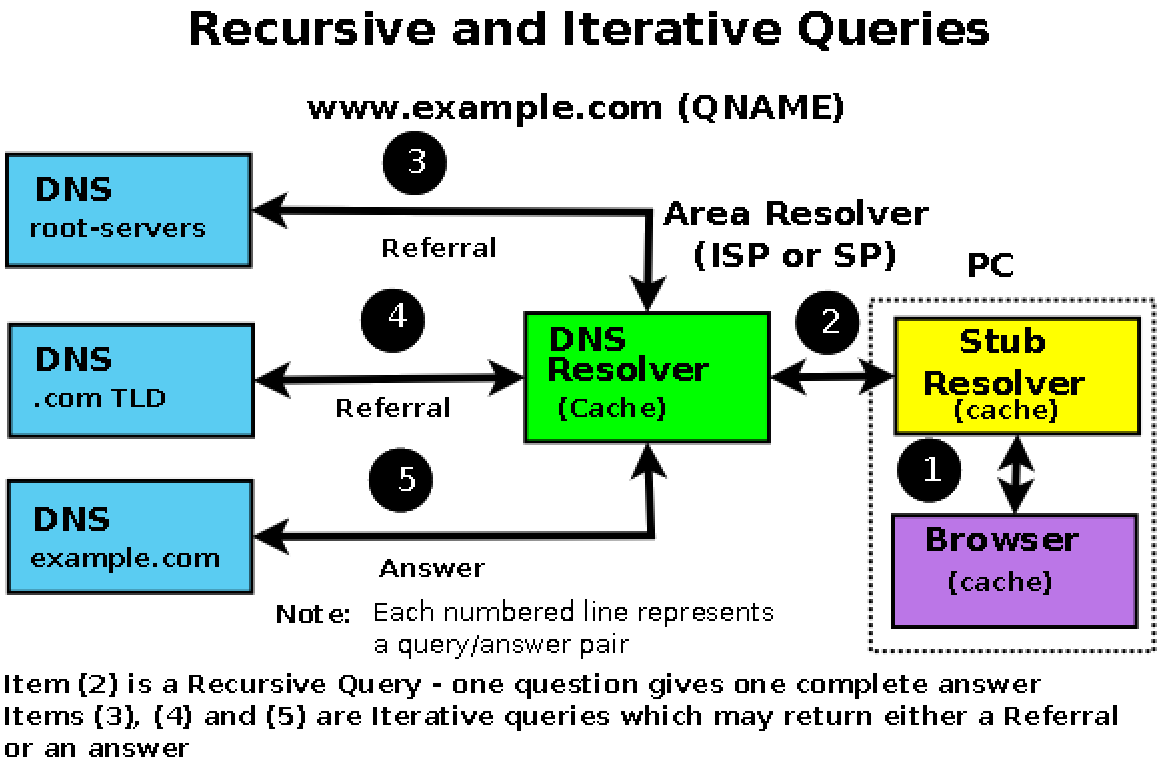

Recursive and Iterative Queries
www.example.com (QNAME)
3
DNS root-servers
Referral
Area Resolver (ISP or SP)
PC
4
DNS .com TLD
Referral
DNS Resolver (Cache)
2
Stub Resolver (cache)
1
5
DNS example.com
Answer
Browser (cache)
Note: Each numbered line represents a query/answer pair
Item (2) is a Recursive Query - one question gives one complete answer
Items (3), (4) and (5) are Iterative queries which may return either a Referral or an answer

Diagram 1-3 Recursive Query Processing

1. The user types www.example.com into their browser address bar. The browser issues a standard function library call (1) to the local stub-resolver.
2. The stub-resolver sends a query (2) 'what is the IP address of www.example.com' to locally configured DNS resolver (aka recursive name server). This is a standard DNS query requesting recursive services (RD (Recursion Desired) = 1).
3. The DNS Resolver looks up the address of www.example.com in its local tables (its **cache**) and does not find it. (If it were found it would be returned immediately to the Stub-resolver in an answer message and the transaction would be complete.)
4. The DNS resolver sends a query (3) to a root-server (every DNS resolver is configured with a file that tells it the names and IP addresses of the root servers) for the IP of www.example.com. (Root-servers, TLD servers and correctly configured user name servers do not, a matter of policy, support recursive queries so the Resolver will, typically, not set Recursion Desired (RD = 0) - this query is, in fact, an Iterative query.)
5. The root-server knows nothing about example.com, let alone the www part, but it does know about the next level in the hierarchy, in this case, the .com part so it replies (answers) with a referral (3) pointing at the TLD servers for .com.
6. The DNS Resolver sends a new query (4) 'what is the IP address of www.example.com' to one of the .com TLD servers. Again it will use, typically, an Iterative query.
7. The TLD server knows about example.com, but knows nothing about www so, since it cannot supply a complete response to the query, it replies (4) with a referral to the name servers for example.com.
8. The DNS Resolver sends yet another query (5) 'what is the IP address www.example.com' to one of the name servers for example.com. Once again it will use, typically, an Iterative query.
9. The example.com zone file defines a A (IPv4 address) record so the authoritative server for example.com returns (5) the A record for www.example.com (it fully answers the question).

10. The DNS Resolver sends the response (answer) www.example.com=x.x.x.x to the client's stub-resolver (2) and then places this information in its cache.
11. The stub-resolver places the information www.example.com=x.x.x.x in its cache (since around 2003 most stub-resolvers have been caching stub-resolvers) and responds to the original standard library function call (1) with www.example.com = x.x.x.x.
12. The browser receives the response to its standard function call, places the information in its cache (really) and initiates an HTTP session to the address x.x.x.x. DNS transaction complete. Quite simple really, not much could possibly go wrong.

In summary, the stub-resolver demands recursive services from the DNS Resolver. The DNS Resolver provides a recursive service but uses, typically, Iterative queries to achieve it.

Note: The resolver on Windows and most *nix systems is a **stub-resolver** (in point of fact, in most modern systems it is a **Caching stub-Resolver**) - which is defined in the standards to be a minimal resolver which cannot follow **referrals**. If you reconfigure your local PC or Workstation to point to a DNS server that only supports Iterative queries - it will not work. Period.

A Iterative (or non-recursive) query

A Iterative (or non-recursive) query is one where the DNS server may provide an answer or a partial answer (a referral) to the query (or give an error). All DNS servers must support non-recursive (Iterative) queries. An Iterative query is technically simply a normal DNS query that does not request Recursive Services.

There are four possible responses to a non-recursive query:

1. The answer to the query accompanied by any CNAME records (aliases) that may be useful (in a Iterative Query this will ONLY occur if the requested data is already available in the cache). The response will indicate whether the data is authoritative or cached.
2. An error indicating the domain or host does not exist (NXDOMAIN). This response may also contain CNAME records that pointed to the non-existing host.
3. An temporary error indication, for instance, can't access other DNS's due to network error etc..

4. A **referral**: If the requested data is not available in the cache then the name and IP addess(es) of one or more name server(s) that are closer to the requested domain name (in all cases this is the next lower level in the DNS hierarchy) will be returned. This referral may, or may not be, to the authoritative name server for the target domain.

In Diagram 1-3 above the transactions (3), (4) and (5) are normally all Iterative queries. Even if the DNS server requested Recursion (RD=1) it would be denied and a normal referral (or answer) returned. Why use Iterative queries? They are much faster, the DNS server receiving the query either already has the answer in its cache, in which case it sends it, or not, in which case it sends a referral. No messing around. Iterative queries give the requestor greater control. A referral typically contains a list of name servers for the next level in the DNS hierarchy. The requestor may have additional information about one or more of these name servers in its cache (including which is the fastest) from which it can make a better decision about which name server to use. Iterative queries are also extremely useful in diagnostic situations.

Inverse Queries

Historically, an Inverse query mapped a resource record to a domain. An example Inverse query would be 'what is the domain name for this MX record'. Inverse query support was optional and it was permitted for the DNS server to return a response **Not Implemented**.

Inverse queries are NOT used to find a host name given an IP address. This process is called Reverse Mapping (Look-up) uses recursive and Iterative (non-recursive) queries with the special domain name IN-ADDR.ARPA.

Inverse queries went the way of all "seemed like a good idea at the time" concepts when they were finally obsoleted by RFC 3425.

Dynamic Host Configuration Protocol (DHCP)

DHCP allows individual computers on a TCP/IP network to obtain their configuration information - in particular, their IP address - from a server. The DHCP server keeps track of which IP addresses have already been assigned so that when a computer requests an IP address, the DHCP server will offer it an IP address that is not already in use.

Configuration information provided by DHCP

Although the primary job of DHCP is to dole out IP addresses and subnet masks, DHCP actually provides more configuration information than just the IP address to its clients. The additional configuration information is referred to as DHCP options. The following is a list of some common DHCP options that can be configured by the server:

- The router address, also known as the Default Gateway address
- The expiration time for the configuration information
- Domain name
- DNS server address
- WINS server address

DHCP server may have three methods of allocating IP-addresses:

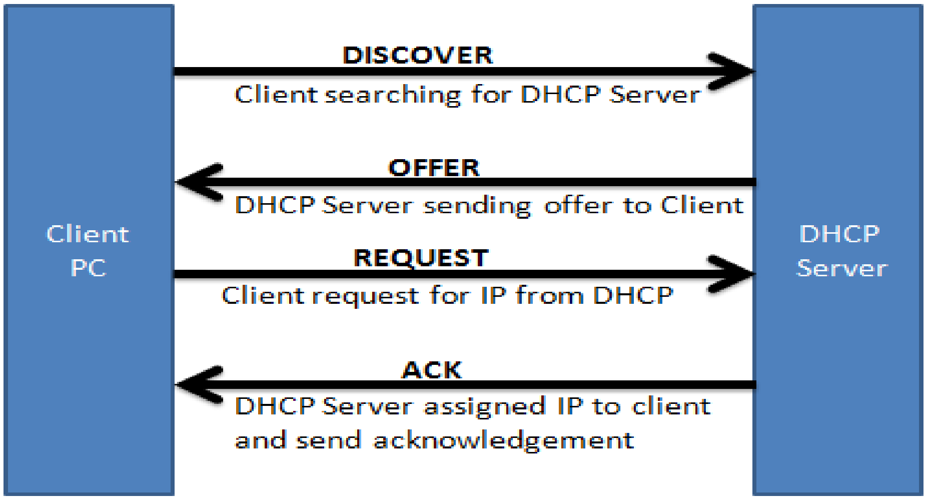
Static allocation: The DHCP server allocates an IP address based on a table with MAC address/IP address pairs, which are manually filled Only requesting clients with a MAC address listed in this table will be allocated an IP address.

Dynamic allocation: A network administrator assigns a range of IP addresses to DHCP, and each client computer on the LAN is configured to request an IP address from the DHCP server during network initialization.

Automatic allocation: The DHCP server permanently assigns a free IP address to a requesting client from the range defined by the administrator. This is like dynamic allocation, but the DHCP server keeps a table of past IP address assignments, so that it can preferentially assign to a client the same IP address that the client previously had.

How DHCP work

DHCP provides an automated way to distribute and update IP addresses and other configuration information on a network. A DHCP server provides this information to a DHCP client through the exchange of a series of messages, known as the DHCP conversation or the DHCP transaction.


Client PC
DISCOVER
Client searching for DHCP Server
OFFER
DHCP Server sending offer to Client
REQUEST
Client request for IP from DHCP
ACK
DHCP Server assigned IP to client and send acknowledgement
DHCP Server

STEP 1:DHCP client sends out a DHCP Discover message to find out the DHCP server. DHCP discover message is a layer 2 broadcast as well as layer 3 broadcast.

Fields in DHCP Discover Message:

Src IP: 0.0.0.0

Dst IP: 255.255.255.255

SrcMAC : DHCP clients MAC address

Dst MAC: FF:FF:FF:FF:FF:FF

Hence from the above fields it is clear DHCP Discover message is a Network Layer and Data Link Layer Broadcast.

STEP 2:DHCP server receives the DHCP discover message from client and sends back the DHCP offer message with field information as below:

Src IP: DHCP Server IP Address

Dst IP: 255.255.255.255 #Still Broadcast as Client still has no IP Address#

SrcMAC : MAC Address of DHCP Server

Dst MAC: DHCP clients MAC address

Hence from above field it is clear that DHCP offer message is a layer 2 unicast but still as layer 3 broadcast.

STEP 3: DHCP client receives the DHCP offer from DHCP server and sends back a DHCP Request message with following fields:

Src IP: 0.0.0.0 #As still the IP address hasn't been assigned to Client#

Dst IP: 255.255.255.255 #Still Broadcast as Client must have received Offer from more than one DHCP server in their domain and the DHCP client accepts the Offer that its receives the earliest and by doing a broadcast it intimates the other DHCP server to release the Offered IP address to their available pool again #

SrcMAC : DHCP clients MAC address

Dst MAC: FF:FF:FF:FF:FF:FF

Above fields concludes that DHCP request message is also a layer 2 unicast and a layer 3 broadcast.

STEP4:

Once the DHCP client sends the request to get the Offered IP address, DHCP server responds with an acknowledge message towards DHCP client with below fields:

Src IP: DHCP Server IP Address
Dst IP: 255.255.255.255
SrcMAC : MAC Address of DHCP Server
Dst MAC: DHCP clients MAC address

From above fields substantiates that DHCP Acknowledge is a layer 2 unicast but still a layer 3 broadcast.

BOOTP Vs DHCP

BASIS FOR COMPARISON	BOOTP	DHCP
Autoconfiguration	Not possible only supports manual configuration.	It automatically obtains and assigns IP addresses.
Temporary IP addressing	Not provided	Provided for a limited amount of time.
Compatibility	Not compatible with DHCP clients.	Interoperable with the BOOTP clients.
Mobile machines	IP Configuration and information access are not possible.	Supports mobility of machines.
Error occurance	Mannual configuration is prone to errors.	Autoconfiguration is immune to errors.
Usage	Provides the information to the diskless computer or workstation.	It requires disks to store and forward the information.

IP address Management

IP Address Management (IPAM) refers to a method of planning, tracking and managing the information associated with a network's Internet Protocol address space. With IPAM software, administrators can ensure that the inventory of assignable IP addresses remains current and sufficient. IPAM simplifies and automates the administration of many tasks involved in IP space management, including writing DNS records and configuring DHCP settings. Additional functionality, such as controlling reservations in DHCP as well as other data aggregation and reporting capability, is also common.

Remote access

Remote access is the ability to access a computer or a network remotely through a network connection. Remote access enables users to access the systems they need when they are not physically able to connect directly; in other words, users access systems remotely by using a telecommunications or internet connection. People at branch offices, telecommuters and people who are traveling may need access to their companies' networks.

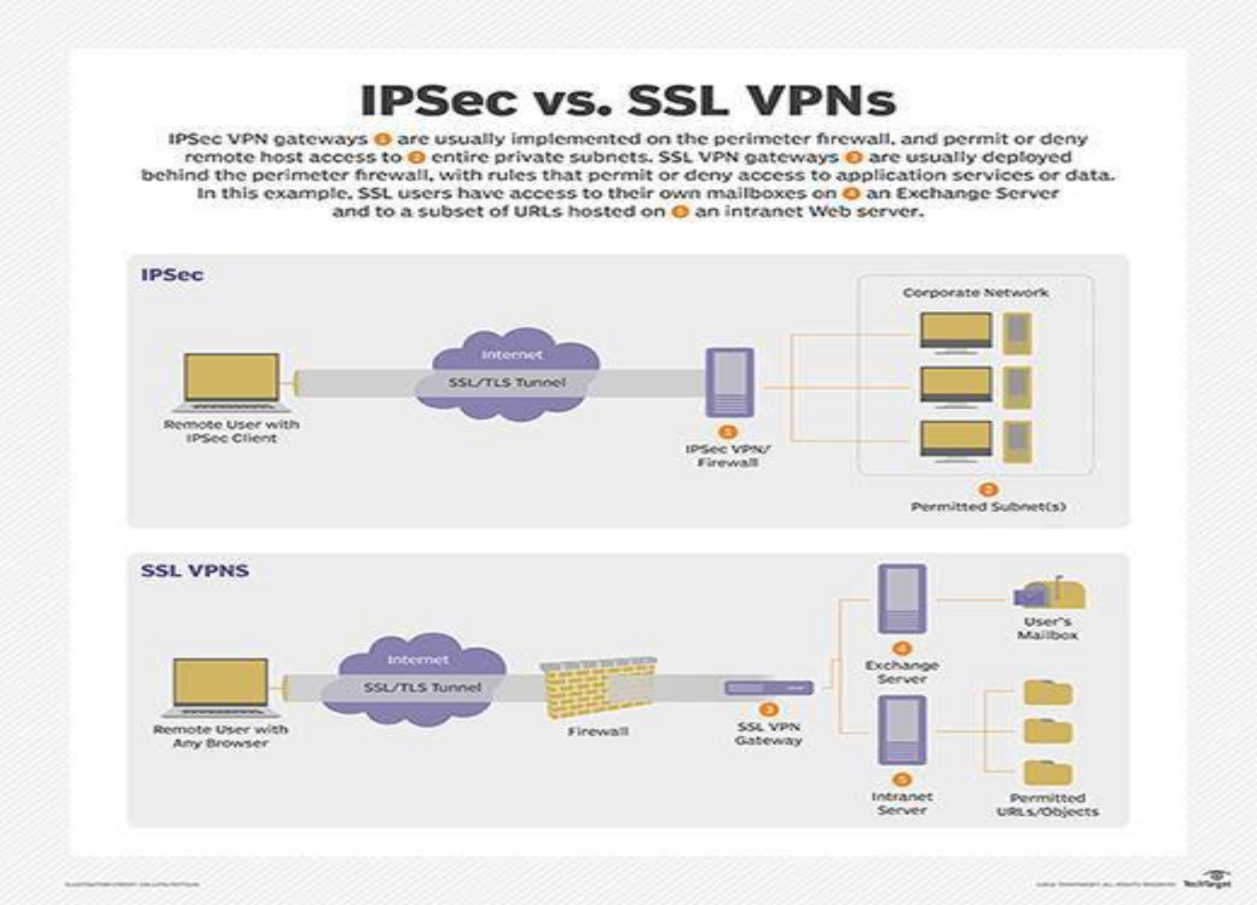
- Remote access enables remote users to access files and other system resources on any devices or servers that are connected to the network at any time, increasing employee productivity and enabling them to better collaborate with colleagues around the world.
- A remote access strategy also gives organizations the flexibility to hire the best talent regardless of location, remove silos and promote collaboration between teams, offices and locations.
- Technical support professionals also use remote access to connect to users' computers from remote locations to help them resolve issues with their systems or software.

One common method of providing remote access is via a remote access VPN connection. A VPN creates a safe and encrypted connection over a less secure network, such as the internet. VPN technology was developed as a way to enable remote users and branch offices to securely log into corporate applications and other resources.

How remote access works

Remote access is usually accomplished with a combination of software, hardware and network connectivity. For example, traditional remote access before the wide availability of internet connectivity was accomplished using terminal emulation software that controlled access over a hardware modem connected to a telephone network. Now, remote access is more commonly accomplished using a secure software solution like a VPN -- software -- by connecting hosts through a hard-wired network interface or Wi-Fi network interface -- hardware -- or by connecting via the internet -- network.

- Remote access VPNs are used to connect individual users to private networks. With a remote access VPN, each user needs a VPN client capable of connecting to the private network's VPN server.
- When a user is connected to the network via a VPN client, the software encrypts the traffic before it delivers it over the internet. The VPN server, or gateway, is located at the edge of the targeted network and decrypts the data and sends it to the appropriate host inside the private network.
- A computer must have software that enables it to connect and communicate with a system or resource hosted by the organization's remote access service. Once the user's computer is connected to the remote host, it can display a window with the target computer's desktop.



IPSec vs. SSL VPNs

IPSec VPN gateways 1 are usually implemented on the perimeter firewall, and permit or deny remote host access to 2 entire private subnets. SSL VPN gateways 3 are usually deployed behind the perimeter firewall, with rules that permit or deny access to application services or data. In this example, SSL users have access to their own mailboxes on 4 an Exchange Server and to a subset of URLs hosted on 5 an intranet Web server.

Using IPsec vs. SSL to power remote access through a VPN

Enterprises can also use remote desktops to enable users to connect to their applications and networks remotely. Remote desktops use application software -- sometimes incorporated into the remote host's operating system -- that enables apps to run remotely on a network server and be displayed locally at the same time.

Users can securely access on-premises and cloud applications and servers from anywhere, on any device with a variety of authentication methods, including remote single sign-on, which gives users easy and secure access to the apps they need without configuring VPNs or modifying firewall policies.

In addition, organizations can use multifactor authentication to verify a user's identity by combining multiple credentials unique to one person.

Types of remote access

Traditionally, enterprises use modems and dial-up technologies to allow employees to connect to office networks via telephone networks connected to remote access servers. Devices connected to dial-up networks use analog modems to call assigned telephone numbers to make connections and send or receive messages. Broadband provides remote users with high-speed connection options to business networks and to the internet. There are several types of broadband, including the following:

- Cable broadband shares bandwidth across many users and, as a result, upstream data rates can be slow during high-usage hours in areas with many subscribers.
- DSL (Digital Subscriber Line) broadband provides high-speed networking over a telephone network using broadband modem tech. However, DSL only works over a limited physical distance and may not be available in some areas if the local telephone infrastructure doesn't support DSL technology.
- Cellular internet services can be accessed by mobile devices via a wireless connection from any location where a cellular network is available.
- Satellite internet services use telecommunications satellites to provide users with internet access in areas where land-based internet access isn't available, as well as for temporary mobile installations.
- Fiber optics broadband technology enables users to transfer large amounts of data quickly and seamlessly.

Remote access protocols

Common remote access and VPN protocols include the following:

- **Point-to-Point Protocol (PPP)** enables hosts to set up a direct connection between two endpoints.
- **IPsec -- Internet Protocol Security** -- is a set of security protocols used to enable authentication and encryption services to secure the transfer of IP packets over the internet.

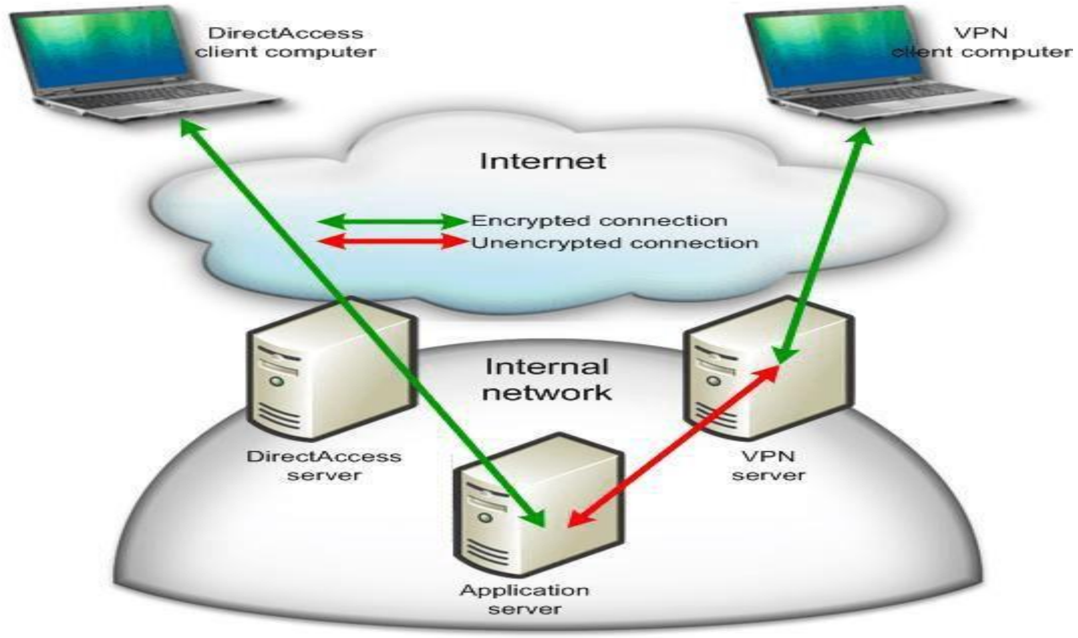
- **Point-to-Point Tunneling (PPTP)** is one of the oldest protocols for implementing virtual private networks. However, over the years, it has proven to be vulnerable to many types of attack. Although PPTP is not very secure, it persists in some cases

- **Layer Two Tunneling Protocol (L2TP)** is a VPN protocol that does not offer encryption or cryptographic authentication for the traffic that passes through the connection. As a result, it is usually paired with IPsec, which provides those services.
- **Remote Authentication Dial-In User Service (RADIUS)** is a protocol developed in 1991 and published as an Internet Standard track specification in 2000 to enable remote access servers to communicate with a central server to authenticate dial-in users and authorize their access to the requested system or service.
- **Terminal Access Controller Access Control System (TACACS)** is a remote authentication protocol that was originally common to Unix networks that enables a remote access server to forward a user's password to an authentication server to determine whether access to a given system should be allowed. TACACS+ is a separate protocol designed to handle authentication and authorization, and to account for administrator access to network devices, such as routers and switches.

Direct Access

Direct Access, also known as Unified Remote Access, is a product of Microsoft, designed exclusively for Windows. It was initially introduced in Windows Server 2008 and Windows 7 Enterprise edition to allow users to access private network resources remotely using the Internet. Direct Access is a more secure, convenient, and advanced alternative than the traditional VPN.

Direct Access primarily aims at providing a seamless intranet connectivity to its users. It offers a transparent always-on connection established by a machine and not by the user. Therefore, Direct Access starts securing the network channel as soon as a client gets on an active Internet connection. Direct Access also provides an authenticated, secure, and bidirectional connection in providing remote access to its users.



What makes Direct Access better than VPN?

- Direct Access overcomes some of the serious drawbacks in the implementation of VPN. As mentioned earlier, VPN connections are user-initiated, whereas in the case of Direct Access, the connection is machine-initiated. Additionally, all the clients are directly connected with management servers, ensuring security configuration compliance.
- Direct Access connections are considerably more secure than those offered by VPN because it is mandatory for all Direct Access clients to have a certificate issued by the organization itself.
- Direct Access is a firewall-friendly feature and is not restricted to any geographical area. It works anywhere, provided the user is connected to the Internet. Conversely, VPN networks face hurdles trying to handle some firewalls and they might sometimes fail to provide secure remote access to all the locations.
- In Direct Access, all clients are constantly monitored and managed by the host or management server, which minimizes the threat of intruders in a network. In the case of a VPN, a client can enter a network without the knowledge of centralized server, which could lead to a security problem with significant risks.

- DirectAccess is a bidirectional connection. Therefore, all the client systems in a DirectAccess network are always serviceable by the management server. A server in a DirectAccess network can easily troubleshoot an issue on a client system, which is not always possible in VPN.
- VPN involves a complex process of establishing a connection to the network, which reduces workers' productivity and efficiency. DirectAccess, on the other hand, is comparatively easy and hassle-free to set up, connect, and use.

Requirements

- One domain controller running Windows Server 2003 or above
- An Internal PKI (Public Key Infrastructure) designed by the organization to assign machine certificates to the clients and servers.
- DirectAccess server must be running on Windows server 2008 R2 and both clients and server must run on Windows 7 Enterprise/Ultimate editions or higher.
- IPv6 must be enabled on all the clients and servers as it is the cornerstone in the functioning of DirectAccess.
- All the DirectAccess clients must be a member of active directory domain.
- DirectAccess server must have two network interface adaptors to support its bidirectional communication.

Advantages of Direct Access

Increased security

- DirectAccess provides a fully encrypted and authenticated mode of connection. It gives employees an authenticated IPSec encryption for integrity and confidentiality. IPv6 is the cornerstone for all the DirectAccess connections, and thus, DirectAccess leverages on IPv6 for transport. Moreover, these IPv6 protocols are implemented ensuring a complete compatibility with all the IPv4 hosts.
- DirectAccess is secured in several stages in the entire remote connectivity process. It utilizes various digital certificates, Kerberos standards, and NTLM to maintain a reliable, secure, and an authenticated connection. Apart from all the aforementioned inbuilt security mechanisms of DirectAccess, organizations can

also integrate smart cards and dynamic one-time passwords for additional security and assurance that only authorized users can connect with the organization.

User experience

Since DirectAccess ships with an always-on connection by default, it doesn't require any specific action or setup from the user to establish a remote connection. DirectAccess provides a seamless user experience and allows a user to access the organizational resources remotely in the same way they do from the office.

Lower Support costs and ease of use

DirectAccess unarguably provides a better user experience to its users over a VPN or any other solution for remote connectivity. In DirectAccess, and entire remote access connection is established at the machine level, relieving the end users from a lengthy process of establishing a remote connection. Since most of the connection process is managed at the machine level, productivity of users increase. And perhaps even better, the work of the IT support staff is decreased.

Support for load balancing

DirectAccess comes integrated with load balancing solutions to provide higher scalability and availability. It uses either Windows network load balancing techniques or employees hardware load balancer, allowing a user to configure multiple DirectAccess servers in an organization so that the load is uniformly balanced across these multiple servers.

DirectAccess proves that mobility is no longer an unsurmountable challenge to the IT field, which is why it is the choice for many individuals and organizations.

Introduction of Active Directory Domain Services

A **directory** is a hierarchical structure that stores information about objects on the network. A directory, in the most generic sense, is a comprehensive listing of objects. A phone book is a type of directory that stores information about people, businesses, and government organizations. Phone books typically record names, addresses, and phone numbers.

Active Directory (AD) is a Microsoft technology used to manage computers and other devices on a network. It is a primary feature of Windows Server, an operating system that runs both local and Internet-based servers.

Benefits of Active Directory –

- Hierarchical organizational structure.
- Multimaster Authentication &Multimaster replication (the ability to access and modify AD DS from multiple points of administration)
- A single point of access to network resources.
- Ability to create trust relationships with external networks running previous versions of Active Directory and even Unix.

Directory Service –

A directory service is a hierarchical arrangement of objects which are structured in a way that makes access easy. However, functioning as a locator service is not AD's exclusive purpose. It also helps organizations have a central administration over all the activities carried out in their networks. Essentially a Network Directory Service:

- Provides information about the user objects, computers and services in the network.
- Stores this information in a secure database and provides tools to manage and search the directory.
- Allows to manage the user accounts and resources, apply policies consistently as needed by an organization.

Active Directory provides several different services, which fall under the umbrella of “Active Directory Domain Services, ” or AD DS. These services include:

1. **Domain Services –**
   Stores centralized data and manages communication between users and domains; includes login authentication and search functionality
2. **Certificate Services –**
   It generates, manages and shares certificates. A certificate uses encryption to enable a user to exchange information over the internet securely with a public key.
3. **Lightweight Directory Services –**
   Supports directory-enabled applications using the open (LDAP) protocol.
4. **Directory Federation Services –**
   Provides single-sign-on (SSO) to authenticate a user in multiple web applications in a single session.
5. **Rights Management –**
   It controls information rights and management. AD RMS encrypts content, such as email or Word documents, on a server to limit access.

Active Directory Domain Services (AD DS) is composed of both logical and physical components

Logical components	**Physical components**
• Partitions • Schema • Domains • Domain trees • Forests • Sites • Organizational units (OUs) • Containers	• Domain controllers • Data stores • Global catalogservers • Read-only domaincontrollers (RODC)

Physical components

Domain Controllers –

A server that is running AD DS is called a domain controller.Domain controllers host and replicate the directory service database inside the forest. The directory service also provides services for managing and authenticating resources in the forest.These servers host essential services in AD DS, including the following:

– Kerberos Key Distribution Center (kdc)
– NetLogon (Netlogon)
– Windows Time (W32time)
– Intersite Messaging (IsmServ)

Global catalog –

A global catalog that contains information about every object in the directory. This allows users and administrators to find directory information regardless of which domain in the directory actually contains the data. For more information about the global catalog, see The role of the global catalog.

Read Only Domain Controller (RODC)

A read only domain controller (RODC) is a type of domain controller that has read-only partitions of Active Directory Domain Services (AD DS) database.

Logical components

Schema –

A set of rules, the schema, that defines the classes of objects and attributes contained in the directory, the constraints and limits on instances of these objects, and the format of their names.

Active Directory organizational units

OUs can be used to form a hierarchy of containers within a domain. OUs are used to group objects for administrative purposes such as the application of Group Policy or delegation of authority. Control (over an OU and the objects within it) is determined by the access control lists (ACLs) on the OU and on the objects in the OU. To facilitate the management of large numbers of objects, AD DS supports the concept of delegation of authority. By means of delegation, owners can transfer full or limited administrative control over objects to other users or groups. Delegation is important because it helps to distribute the management of large numbers of objects across a number of people who are trusted to perform management tasks.

Domain –

It is a logical group of users and computers that share the characteristics of centralized security and administration. A domain is still a boundary for security – this means that an administrator of a domain is an administrator for only that domain, and no others, by default. A domain is also a boundary for replication – all domain controllers that are part of the same domain must replicate with one another. Domains in the same forest automatically have trust relationships configured.

Tree –

A tree is a collection of Active Directory domains that share a contiguous namespace. In this configuration, domains fall into a parent-child relationship, which the child domain taking on the name of the parent.

Forest –

A forest is the largest unit in Active Directory and is a collection of trees that share a common Schema, the definition of objects that can be created. In a forest all trees are connected by transitive two-way trust relationships, thus allowing users in any tree access to resources in another for which they have been given appropriate permissions and rights. By default the first domain created in a forest is referred to as the root

Domain . Amongst other things, this is where the Schema is stored by default.

There are two types of active directory forest :-

I) Single Forest

2) Multiple forest

Organizational Unit –

An organizational unit (OU) is a container object that helps to organize objects for the purpose of administration or group policy application. An OU exists within a domain and can only contain objects from that domain. OU can be nested, which allows for more flexibility in terms of administration. Different methods for designing OU structures exist including according to administration (most common), geography, or organizational structure. One popular use of OUs is to delegate administrative authority – this allows you to give a user a degree of administrative control over just the OU, and not the entire domain.

Sites

Active Directory Sites are the best solution for managing organizations that have branches in different geographical locations, but fall under the same domain. Sites are physical groupings of well-connected IP subnets that are used to efficiently replicate information among Domain Controllers (DCs). It can be thought of as a mapping that describes the best routes for carrying out replication in AD, thus making efficient use of the network bandwidth. Sites help to achieve cost-efficiency and speed. It also lets one exercise better control over the replication traffic and the authentication process. When there is more than one DC in the associated site that is capable of handling client logon, services, and directory searches, sites can locate the closest DC to perform these actions. Sites also play a role in the deployment and targeting of Group Policies.

In AD, the information about the topology is stored as site link objects. By default, the Default-First-Site-Name site container is created for the forest. Until another site is created, all DCs are automatically assigned to this site.

Subnets

Within sites, subnets help in grouping neighboring computers based on their IP address. So every subnet is identified by a range of IP associated addresses, and a site is the aggregate of all well connected subnets. Subnets could be based on either TCP/IPv4 or TCP/IPv6 addresses.

Site Links

As the name implies, site links are used to establish links between sites, the default site link being called Default-First-Site-Link. They give the flow of the replication that takes place between sites. By configuring site link properties such as site link schedule, replication cost and interval, intersite replication can be managed.

Sites and Replication

In AD, when a change is applied to a specific DC, all other DCs in the domain are informed about the change and updated. This happens through the process of replication.

Partitions in Active Directory

The active directory database is stored in a single NTDS.dit file which is logically separated into the following partitions:

- Schema Partition
- Configuration Partition
- Domain Partition
- Application Partition

Schema Partition

There is only one schema partition per forest and it is stored in all DCs of the forest. It contains the definition of objects and rules for their manipulation and creation in an active directory. It is replicated to all DCs of the forest.

Configuration Partition

Just like schema partition, there is just one master configuration partition per forest and a second one on all DCs in a forest. It contains the forest-wide active directory topology including DCs and sites and services. It is replicated to all DCs in a forest.

Domain Partition

Many domain partitions exist per forest and they are stored on all DCs in a domain. They contain information about users, groups, computers and OUs. It is replicated to all DCs in a given domain.

Application Partition

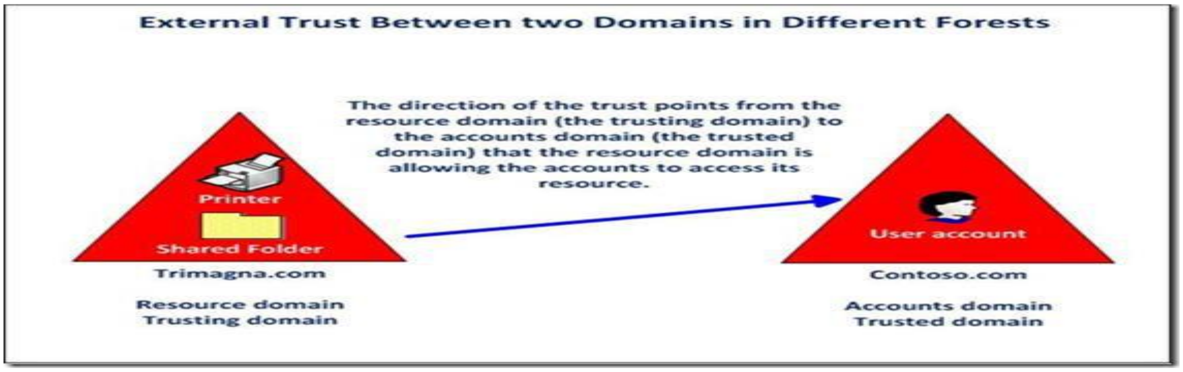
This partition stores information about applications in an AD. Suppose AD integrated DNS zones information is stored in this partition.

Active Directory Trusts

Active Directory domain to domain communications occur through a trust. An AD DS trust is a secured, authentication communication channel between entities, such as AD DS domains, forests, and UNIX realms. Trusts enable you to grant access to resources to users, groups and computers across entities.

The way a trust works is similar to allowing a trusted entity to access your own resources. It’s a two-step process. The first step is to establish the trust. The second step is to provide permissions.

For example, if users in the Contoso.com domain require access to a shared folder in the Trimagna.com domain, and the two domains are not in the same forest, you would establish the trust where Trimagna.com trusts Contoso.com, therefore the direction of the arrow would be Trimagna.com points to Contoso.com.

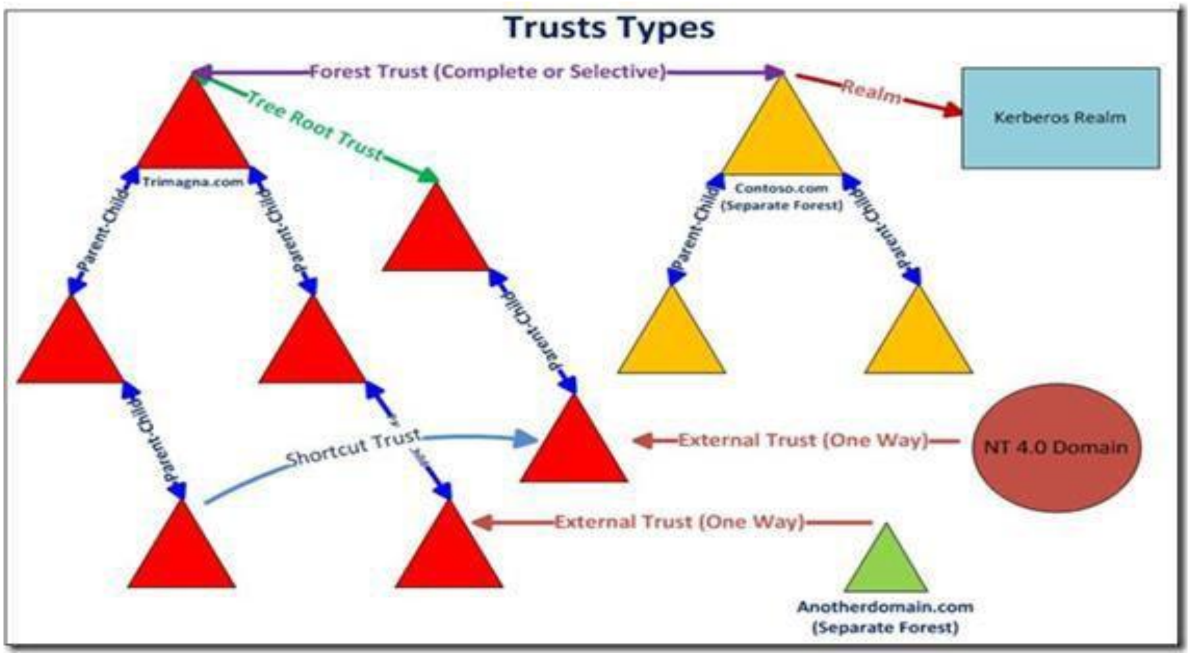

External Trust Between two Domains in Different Forests
The direction of the trust points from the resource domain (the trusting domain) to the accounts domain (the trusted domain) that the resource domain is allowing the accounts to access its resource.
Printer
Shared Folder
Trimagna.com
Resource domain
Trusting domain
User account
Contoso.com
Accounts domain
Trusted domain

For an analogy, if you were to give your car keys to a friend to allow him or her to use your car, you are establishing a trust between you and your friend. In this case, you are the trusting friend, or domain, and the friend is the trusted friend, or domain. Once the keys have been provided, then the next step is to allow access to your resource, or car, by providing permissions to use the car. However, this trust is only in one direction, you trust your friend. If you want your friend to trust you, your friend, or the other domain, must be initiated by your friend, or the other domain.

AD DS Trust Types

There are various trust types. The trust that you create must be appropriate for the design. Trusts can be transitive or non-transitive. The one that you choose to create depends on the scenario and requirements. Other trusts types can be created as required, depending on the scenario. The table below shows the various trust types you can create.

Trusts can be created using the New Trust Wizard found in the Active Directory Domains and Trusts console, or using the Netdom command line utility. If you choose to create one of the one-way trust types in both directions, it can be created simultaneously, or separately. If you create it separately, you must re-run the procedure to establish the trust in the other direction.



Trust Type	Characteristics	Direction	Authentication Mechanism	Notes
Parent-Child	Transitive	Two-way	Kerberos V5 or NTLM	Created automatically when a child domain is added.
Tree-Root	Transitive	Two-way	Kerberos V5 or NTLM	Created automatically when a new Tree is added to a forest.
Shortcut	Transitive	One-way or Two-way	Kerberos V5 or NTLM	Created Manually. Used in an AD DS forest to shorten the trust path to improve authentication times.
Forest	Transitive	One-way or Two-way	Kerberos V5 or NTLM	Created Manually. Used to share resources between AD DS forests.
External	Non-transitive	One-way	NTLM Only	Created Manually. Used to access resources in an NT 4.0 domain or a domain in another forest that does not have a forest trust established.
Realm	Transitive or non-transitive	One-way or Two-way	Kerberos V5 Only	Created Manually. Used to access resources between a non-Windows Kerberos V5 realm and an AD DS domain.

Trust Flow: Transitive vs. Non-Transitive

Trust communication flow is determined by the direction of the trust. The trust can be a one-way or a two-way trust. And the transitivity determines whether a trust can be extended beyond the two domains with which it was formed. A transitive trust can be used to extend trust relationships with other domains; a non-transitive trust can be used to deny trust relationships with other domains. Authentication requests follow a trust path. The transitivity of the trust will affect the trust path.

Transitive Trust

· Contoso.com trusts Trimagna.com.

· Contoso.com trusts Adatum.com.

· Therefore, Trimagna.com trusts Adatum.com.

One-way Trust

· Domain A trusts Domain B, but Domain B does not trust Domain A.

· Domain A trusts Domain C, but Domain C does not trust Domain A.

· Therefore, Domain B does not trust Domain C.

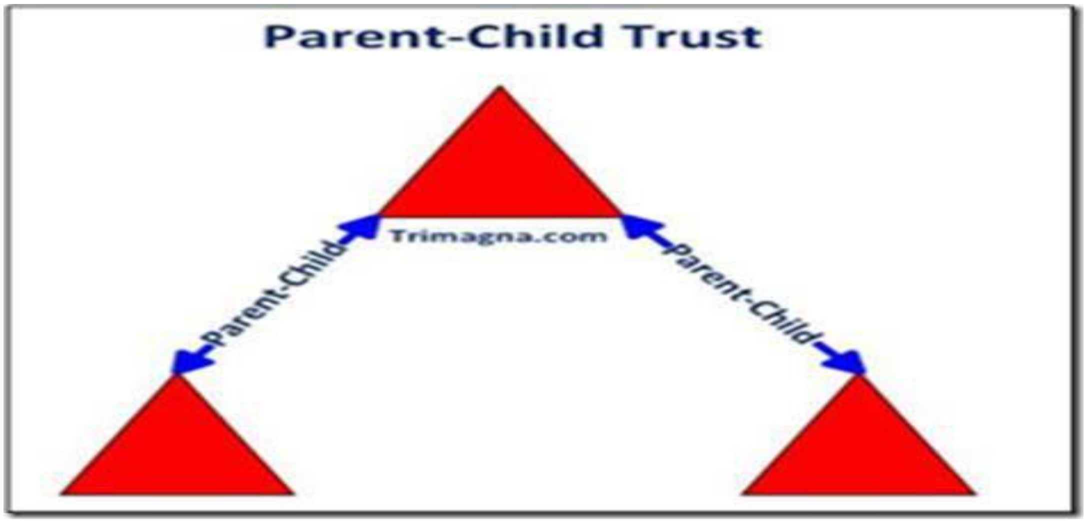
For these two domains to trust each other, you would need a one way trust created between each other.

Automatic Trusts: and Tree-Root Trusts

By default, two-way, transitive trusts are created automatically when a child domain is added or when a domain tree is added. The two default trust types are parent-child trusts and tree-root trusts.

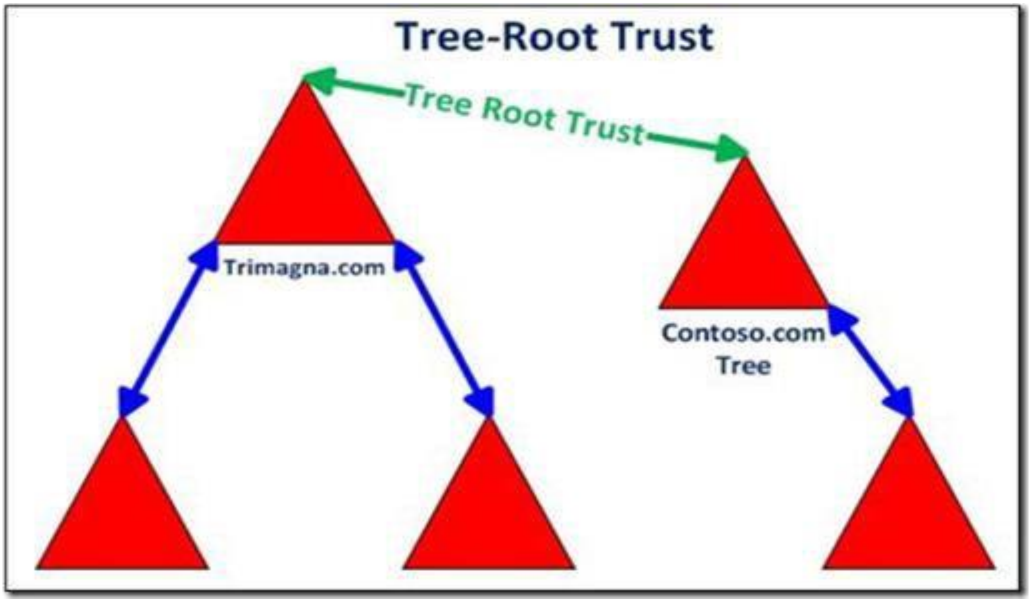
Parent-Child Trust

A transitive, two-way parent-child trust relationship automatically created and establishes a relationship between a parent domain and a child domain whenever a new child domain is created using the AD DS installation process process within a domain tree. They can only exist between two domains in the same tree with the same contiguous namespace. The parent domain is always trusted by the child domain. You cannot manually create a Parent-Child trust.



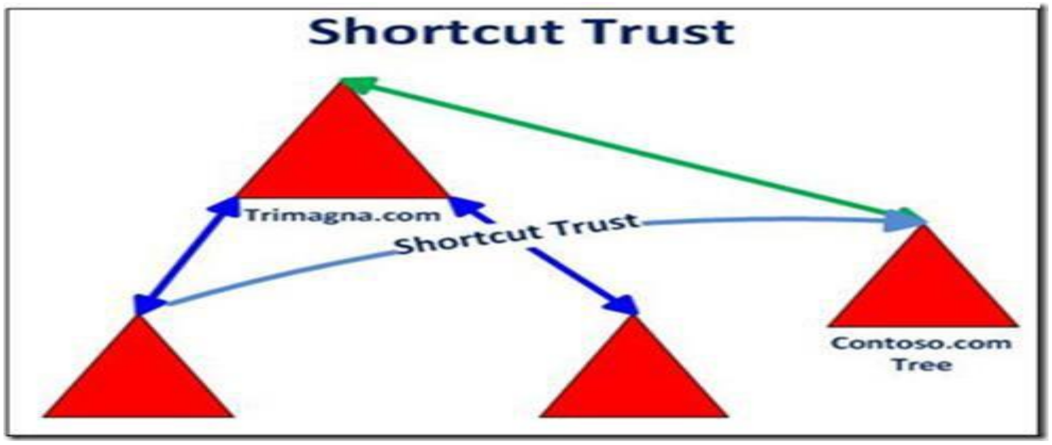
Tree-Root Trust

A transitive, two-way tree-root trust relationship automatically created and establishes a relationship between the forest root domain and a new tree, when you run the AD DS installation process to add a new tree to the forest. A tree-root trust can only be established between the roots of two trees in the same forest and are always transitive. You cannot manually create a tree-root trust.



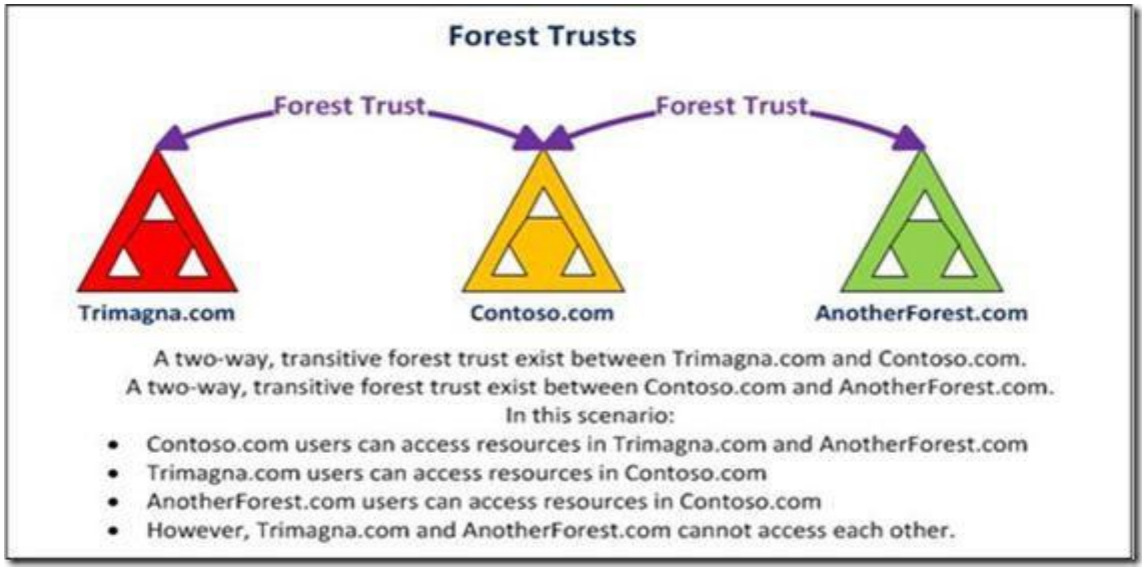
Shortcut Trust

Shortcut trusts are manually created, one-way, transitive trusts. They can only exist within a forest. They are created to optimize the authentication process shortening the trust path. The trust path is the series of domain trust relationships that the authentication process must traverse between two domains in a forest that are not directly trusted by each other. Shortcut trusts shorten the trust path.



Forest Trust

Forest trusts are manually created, one-way transitive, or two-way transitive trusts that allow you to provide access to resources between multiple forests. Forest trusts uses both Kerberos v5 and NTLM authentication across forests where users can use their Universal Principal Name (UPN) or their Pre-Windows 2000 method (domainName\username). Kerberos v5 is attempted first, and if that fails, it will then try NTLM.



A two-way, transitive forest trust exist between Trimagna.com and Contoso.com.
A two-way, transitive forest trust exist between Contoso.com and AnotherForest.com.
In this scenario:

- Contoso.com users can access resources in Trimagna.com and AnotherForest.com
- Trimagna.com users can access resources in Contoso.com
- AnotherForest.com users can access resources in Contoso.com
- However, Trimagna.com and AnotherForest.com cannot access each other.

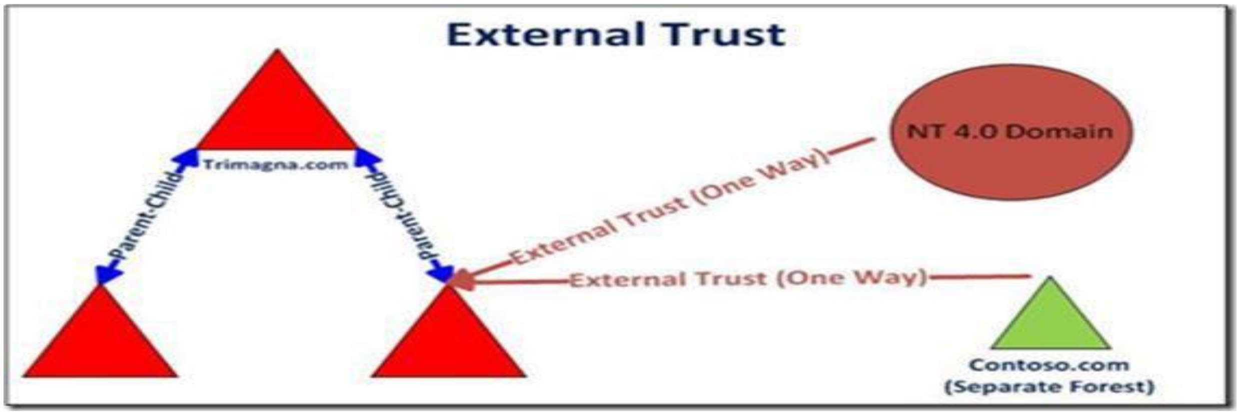
Forest trusts require DNS resolution to be established between forests, however to support NTLM failback, you must also provide NetBIOS name resolution support between the forests.

Forest trusts also provide SID filtering enforcement in Windows Server 2003 and newer. This ensures that any misuse of the SID history attribute on security principals (including the inetOrgPerson attribute) in the trusted forest cannot pose a threat to the integrity of the trusting forest.

Forest trusts cannot be extended to other forests, such as if Forest 1 trusts Forest 2, and another forest trust is created between Forest 2 and Forest 3, Forest 1 does not have an implied trust. If a trust is required, one must be manually created.

External Trust

An external trust is a one-way, non-transitive trust that is manually created to establish a trust relationship between AD DS domains that are in different forests, or between an AD DS domain and Windows NT 4.0 domain. External trusts allow you to provide users access to resources in a domain outside of the forest that is not already trusted by a Forest trust.



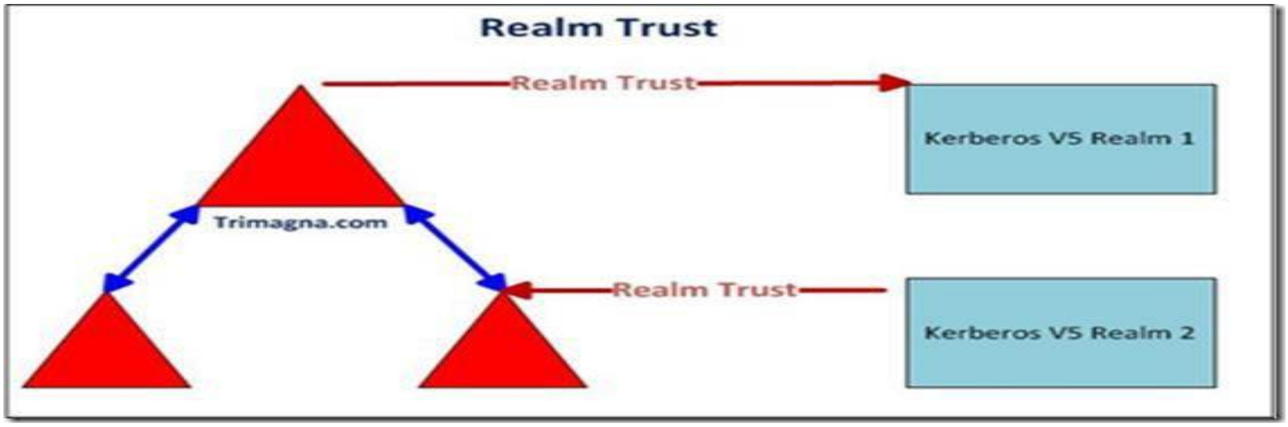
SID filter quarantining is enabled by default with Windows Server 2003 and newer AD DS domains. SID filtering verifies that incoming authentication requests made from security principals in the trusted domain contain only SIDs of security principals from the trusted domain.

External trusts are NTLM based, meaning users must authenticate using the Pre-Windows 2000 logon method (domain\username).NTLM requires NetBIOS name resolution support for functionality.

Realm Trust

A Realm trust can be established to provide resource access and cross-platform inter-operability between an AD DS domain and non-Windows Kerberos v5 Realm.

- A Realm trust only uses Kerberos V5 authentication. NTLM is not used.
- When the direction of the trust is from a non-Windows Kerberos Realm to an AD DS domain (Realm trusts AD DS domain), the non-Windows realm trusts all security principals in the AD DS domain.
- Realm trusts are one-way by default, but you can create a trust in the other direction to allow two-way access.
- Because non-Windows Kerberos tickets do not contain all the information AD DS requires, the AD DS domain only uses the account to which the proxy account (the non-Windows principal) is mapped to evaluate access requests and authorization. With Realm trusts, all AD DS domain proxy accounts can be used in an AD DS group in ACLs to control access for non-Windows accounts.



Additional reading:

- Understanding Trust Types

Trusted Domain Object (TDO)

To understand cross domain authentication, we must first understand Trusted Domain Objects (TDOs). Each domain within a forest is represented by a TDO that is stored in the System container within its domain. The information in the TDO varies depending on whether the TDO was created by a domain trust or by a forest trust.

When a domain trust is created, attributes such as the DNS domain name, domain SID, trust type, trust transitivity, and the reciprocal domain name are represented in the TDO. When a forest trust is first established, each forest collects all of the trusted namespaces in its partner forest and then stores the information in a TDO. The trusted namespaces and attributes that are stored in the TDO include domain tree names, child domain names, user principal name (UPN) suffixes, service principal name (SPN) suffixes, and security ID (SID) namespaces used in the other forest. TDO objects are stored in each domain, then replicated to the global catalog.

Therefore, because trusts are stored in Active Directory in the global catalog as TDOs, all domains in a forest have knowledge of the trust relationships that are in place throughout the forest. If there are two or more forests that are joined together through forest trusts, the forest root domains in each forest know of the trust relationships throughout all of the domains in the trusted forests.

The only exception to the rule is External trusts to a Windows NT 4.0 domain do not create TDOs in Active Directory because it is NTLM based, in which SPN and domain SIDs do not exist, therefore do not apply.

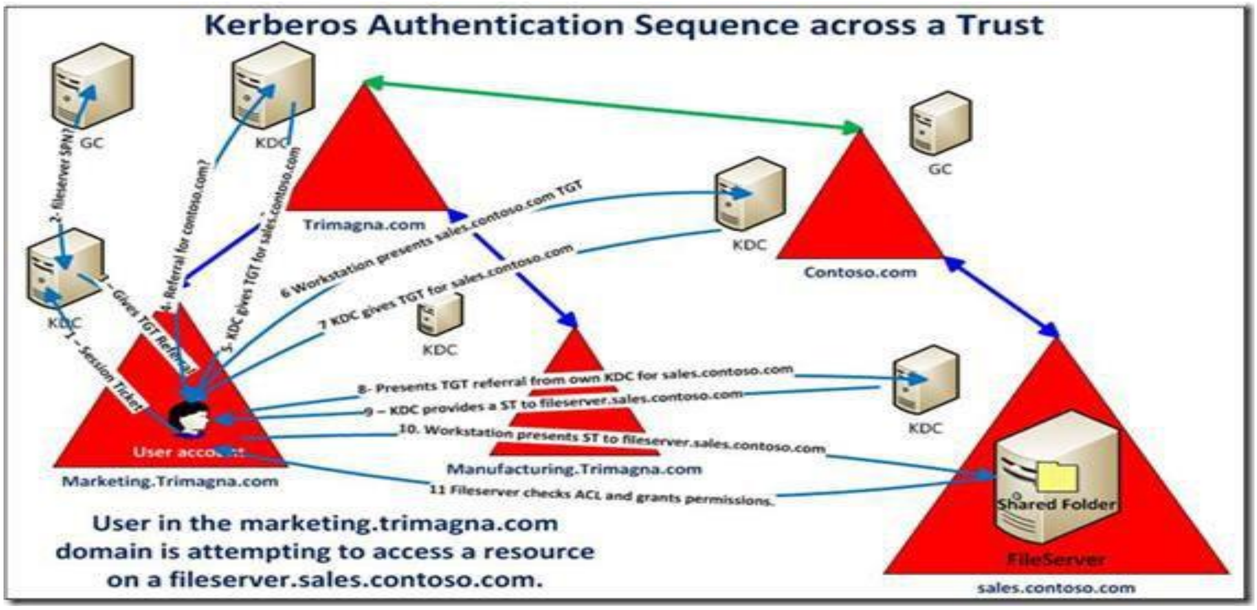
Trust Path between Domains

The trust path is the series of domain trust relationships that the authentication process must traverse between two domains in a forest that are not directly trusted by each other.

Before authentication for a user, computer or service can occur across trusts, Windows must determine if the domain being requested has a trust relationship with the requesting account’s logon domain. This is determined by quering the global catalog for TDO data. The Windows security system’s Netlgon service through an authenticated RPC (Remote Procedure Call) to the remote domain’s trusted domain authority, (the remote domain controller), computes a trust path between the domain controller for the server that receives the request and a domain controller in the domain of the requesting account.

The Windows security system extends a secured channel to other Active Directory domains through interdomain trust relationships. This secured channel is used to obtain and verify security information, including security identifiers (SIDs) for users and groups. The trust path is stored for authentication requests to the trusted domain.

Kerberos authentication Sequence between Domains in a Forest


Kerberos Authentication Sequence across a Trust
GC
KDC
GC
KDC
KDC
KDC
KDC
Trimagna.com
Contoso.com
Marketing.Trimagna.com
Manufacturing.Trimagna.com
sales.contoso.com
User acco...
Shared Folder
FileServer
1 – Session Ticket
2- fileserver SPN?
3 – Gives TGT Referral
4- Referral for contoso.com?
5- KDC gives TGT for sales.contoso.com
6 Workstation presents sales.contoso.com TGT
7 KDC gives TGT for sales.contoso.com
8- Presents TGT referral from own KDC for sales.contoso.com
9 – KDC provides a ST to fileserver.sales.contoso.com
10. Workstation presents ST to fileserver.sales.contoso.com
11 Fileserver checks ACL and grants permissions.
User in the marketing.trimagna.com domain is attempting to access a resource on a fileserver.sales.contoso.com.

A user in the marketing.trimagna.com domains needs to gain access to a file share on a server called fileserver.sales.contoso.com domain. This is assuming the User has already logged on to a workstation using credentials from the marketing.trimagna.com domain. As part of the logon process, the authenticating domain controller issues the User a ticket-granting ticket (TGT). This ticket is required for User1 to be authenticated to resources.

The User attempts to access a shared resource
on **\\FileServer.sales.contoso.com\share**.

The following Kerberos V5 authentication process occurs:

1. The User's workstation asks for a session ticket for the FileServer server in sales.contoso.com by contacting the Kerberos Key Distribution Center (KDC) on a domain controller in its domain (ChildDC1) and requests a service ticket for the FileServer.sales.contoso.com service principal name (SPN).

2. The KDC in the user's domain (marketing.trimagna.com) does not find the SPN for FileServer.sales.contoso.com in its domain database and queries the GC to see if any domains in the forest contain this SPN.

a. The GC checks its database about all forest trusts that exist in its forest. If a trust to the target domain is found, it compares the name suffixes listed in the forest trust trusted domain objects (TDOs) to the suffix of the target SPN to find a match.

b. Once a match is found, the global catalog sends the requested information as a referral back to the KDC in marketing.trimagna.com.

3. The KDC in the marketing.trimagna.com then issues the workstation a TGT for the contoso.com domain. This is known as a referral ticket.

4. The workstation then contacts the KDC in the trimagna.com tree root domain to request a referral to the KDC in the sales.contoso.com.

5. The KDC in the trimagna.com domain recognizes the user’s request to establish a session with a resource that exists in a foreign domain’s server.

a. The KDC then issues a TGT for the KDC in the contoso.com domain.

6. The workstation then presents the TGT for the sales.contoso.com domain to the KDC in the contoso.com domain.

7. The contoso.com KDC queries a GC to see if any domains in the forest contain this SPN. The GC checks its database about all forest trusts that exist in its forest. If a trust to the target domain is found, it compares the name suffixes listed in the forest trust trusted domain objects (TDOs) to the suffix of the target SPN to find a match.

a. Once a match is found, the global catalog sends the requested information as a referral back to the KDC in contoso.com.

8. The KDC issues a TGT for the sales.contoso.com domain.

9. The workstation then contacts the KDC of the sales.contoso.com domain and presents the referral ticket it received from its own KDC.

a. The referral ticket is encrypted with the interdomain key that is decrypted by the foreign domain’s TGS.

b. Note: When there is a trust established between two domains, an interdomain key based on the trust password becomes available for authenticating KDC functions, therefore it’s used to encrypt and decrypt tickets.

10. The workstation also presents the KDC in the sales.contoso.com the TGT it received from the KDC in contoso.com for the sales.contoso.com domain and is issued a ST (Session Ticket) for the sales.contoso.com domain.

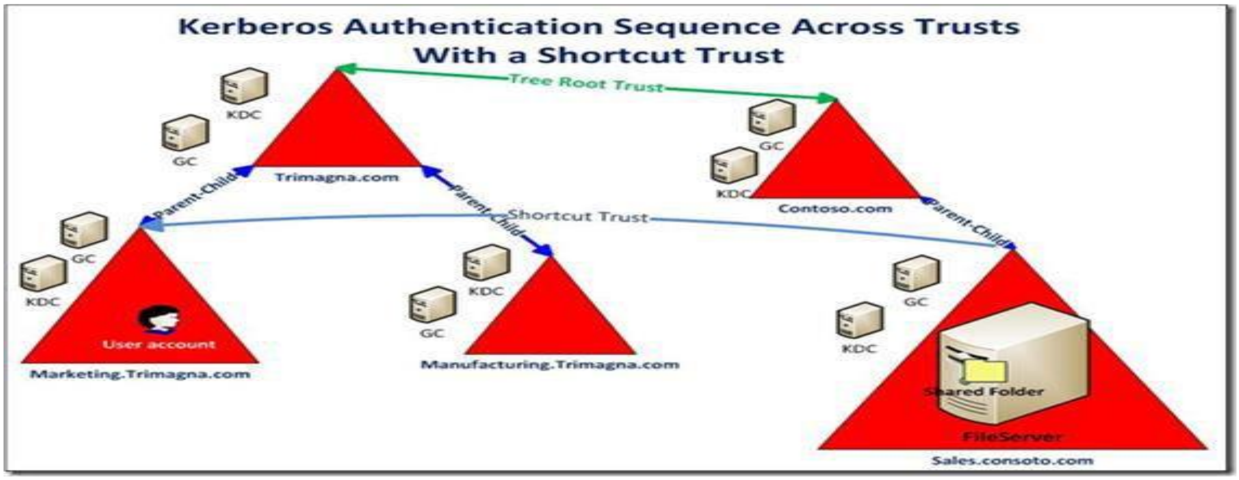
a. The ST is populated with the domain local group memberships from the sales.contoso.com domain.

11. The user presents FileServer.sales.contoso.com the ST to the server to gain access to resources on the server in sales.contoso.com.

12. The server, FileServer.sales.contoso.com compares the SIDs include in the session ticket to the ACEs on the requested resource to determine if the user is authorized to access the resource. If there is, the user is permitted to access the resource based on the ACL permissions.

Kerberos Authentication with a Shortcut Trust

If a shortcut trust exists from the sales.contoso.com domain to the marketing.trimagna.com domain, then the trust path will shortened, therefore the user authentication path will be direct between the two domains.



Group policy

Group policy is a feature of Microsoft Windows Active Directory that adds additional controls to user and computer accounts. Group policies provide centralized management and operating systems configurations of user's computing environments. Group policies are another method of securing user's computers from infiltration and data breaches.

If you care about data security, you need to understand group policies. We will discuss what group policies and GPOs are and how system administrators use them to protect, secure, and lock down computers and user accounts. We will also discuss how attackers can disable group policies as part of their infiltration.

What is Group Policy Object (GPO)?

A Group Policy Object is a collection of settings systems administrators create with the Microsoft Management Console (MMC) Group Policy Editor. The GPO can be associated with one or more of the Active Directory containers, such as sites, domains, or organizational units (OUs).

How Group Policy Objects Are Processed

Active Directory applies GPOs in the following predictable and logical order.

1. Local policies
2. Site policies
3. Domain policies
4. OU policies
   GPOs in nested OUs apply from the OU closest to the root first, and then continue from there

Do I Need a Group Policy?

Assuming the goal of your organization is to become more secure, then yes, you need to understand and implement group policies.

There are ways to rectify those deficiencies through GPOs. Microsoft didn't assume how you wanted to secure your systems, but GPOs can move you closer.

For example, with GPOs you can completely disable Local Administrator rights globally in your network and instead, grant administrative permissions to a single individual or group based on their job. Ideally, you are implementing a least-privileged model where even the system administrators are limited to administering only the servers they are assigned.

Group policies can disable outdated protocols like SSLv2, prevent users from making changes to local group policies, and much more.

Benefits of Group Policy

There are several advantages to implementing GPOs outside of security.

- **Ease of management:** Setting up new users on the network used to be a long and tedious process. Pre-existing GPOs apply a standardized environment to each new user and computer that joins your domain which saves many hours of configurations.
- **One-stop administration:** Sysadmins can deploy patches, software, and other updates via GPO.
- **Password policy enforcement:** Passwords can be easily brute-forced if they aren't changed regularly, contain simple words, or are short. GPOs establish length, reuse rules, and other requirements for passwords to keep your network safe.
- **Folder redirections:** Do you want users to keep important company files on a centralized and monitored storage system? Use a folder redirection GPO to redirect their user folder to your NAS.

Limitations of Group Policy

GPOs update randomly every 90 to 120 minutes or so, or when the computer gets rebooted. You can specify an update rate from 0 to 64,800 minutes (or 45 days), but if you select 0 minutes, the computer tries to update GPOs every 7 seconds. That's going to murder a network with traffic. If you must implement an emergency GPO update, you have to keep this in mind and use another method to get users to reboot.

Also, the GPO editor isn't the best and most intuitive thing in the world. You can learn to use PowerShell instead to make all the updates, which could be easier for a command line person.

If you do implement GPOs, consider the possibility that an attacker tries to circumvent security by changing local GPOs on a computer they have infiltrated. For example, if you locked down the Local Administrator account with a GPO, an attacker can try to reverse that GPO and take over Local Admin. Or they might re-enable a less secure network protocol.

Network Policy Server (NPS)

Network Policy Server (NPS) is the Microsoft implementation of a Remote Authentication Dial-in User Service (RADIUS) server and proxy. It is the successor of Internet Authentication Service (IAS).

As a RADIUS server, NPS performs authentication, authorization, and accounting for wireless, authenticating switch, and remote access dial-up and virtual private network (VPN) connections.

NPS is also a health evaluator server for Network Access Protection (NAP). NPS performs authentication and authorization of network connection attempts and, based on configured system health policies, evaluates computer health compliance and determines how to limit a noncompliant computer's network access or communication. This is a new feature specific to NPS only; IAS does not support it. See Internet Authentication Service and Network Policy Server for a complete list of features new to NPS.

NPS includes two API sets: NPS Extensions API and Server Data Objects (SDO) API. Both NPS Extensions API and SDO API are also supported by the precursor of NPS, the Internet Authentication Service.

NPS Extensions API can be used to extend the authentication, authorization, and accounting methods offered by NPS and previously by IAS.

Server Data Objects API can be used to manipulate the network policy configuration on a computer that runs NPS or IAS.

Active Directory Certificate Services

What is a Certificate

Before we delve into the Active Directory Certificate Services (AD CS), let us understand certificates. A digital certificate and traditional certificate have quite a number of similarities.

- The certificates contain the issuing authority's name. While a traditional certificate contains particulars of an university, organization or government agency, the digital certificate has details of the issuing authority. However, the authority who has issued the certificate must be a trusted source.
- The name of the person to whom it is granted. While the traditional certificate contains the name of the person/organization to whom it is issued, the digital certificate contains the name of the users or computer or device to whom the certificate is issued to.
- Additionally, a digital signature is present, not unlike the seal in the traditional certificate which proves that the certificate is legit.

Another key field is the validity of the certificate, beyond which the certificate cannot be used. The difference between a traditional certificate and an digital certificate is the addition of another field called public key. The latter can be used as a public key for encrypting the data which can be decrypted only be the end user who has the key.

AD CS is one of the server roles introduced in Windows Server 2000 facilitating certificate infrastructure which issues and manages public key certificates. The applications supported by AD CS are secure wireless networks, virtual private networks (VPN), Internet Protocol Security (IPSec), Network Access Protection (NAP), Encrypting File System (EFS), smart card logon, etc.

In the earlier versions of Windows Server 2008 R2, AD CS is a forest level resource. Enterprises with multiple Active Directory Domain Services (AD DS) forests had to install certificate authority in each forest where users or computers required automatic enrollment of certificates.

AD CS has the following components for configuration

- Certification Authority
- Certification Authority Web Enrollment
- Online Responder
- Network Device Enrollment Service
- Certificate Enrollment Web Service
- Certificate Enrollment Policy Web Service

Certification Authority

A certification authority (CA) is used to issue and manage public key certificates. Multiple CAs can be linked to form a public key infrastructure (PKI). A typical PKI consists of software, hardware, standards, and policies to manage the digital certificates. CA can be of two types: enterprise CA and stand-alone CA. The enterprise CA must be a domain member and can issue certificates for digital signatures, authentication to access protected web browsers, and secure e-mail transactions. A stand-alone CA does not require Active Directory Domain Services and can function offline.

Certification Authority Web Enrollment

The CA Web Enrollment allows external clients who are not part of the domain network to connect to the CA via web browser. CA Web Enrollment only supports interactive requests that the requester creates and uploads manually through the web site. The certificate can be downloaded from the browser after the CA issues the certificate. In case of users who are a part of the domain, the trust relationship allows the CA to issue certificate securely.

Web enrollment allows the external clients to request certificates and revoke certificates list from the CA. The enrollment could also be done across forests. This means the clients in one forest can obtain certificates from a CA in another forest. In order to use enrollment across forests, you must establish trust between all the involved forests, and the forest trust and forest level must be set to Windows Server 2008 R2.

Online Responder

Online Responder receives and processes requests on the status of the certificates. The validity of the certificate and digital signature is verified to identify if the certificate is genuine. In addition to that, the certificate is checked to identify if it is included in the Certificate Revocation List (CRL). Due to various reasons, the certificates can be revoked temporarily or can be stripped off its rights permanently before its validity period by the CA. These certificates are listed in the CRL. Apart from CRL, the revocation checking can also be by Online Certificate Status Protocol (OCSP) response. The OCSP checks the status of website in question by sending the URL to the Certificate Authority . The Certificate Authority gives a signed response containing the requested certificate’s status.

Network Device Enrollment Service

The Network Device Enrollment Service (NDES) is a function of AD CS which can issue certificates to network devices managing traffic such as routers, firewalls, and switches. These devices are not Active Directory domain members and thus do not possess exclusive Active Directory credentials. NDES enables one-time enrollment passwords for the network devices. These password requests are sent to the CA for processing and the certificates obtained from the CA are forwarded to the device. Thus NDES service is used by the administrators for authentication of such devices.

Certificate Enrollment Web Service

The Certificate Enrollment Web Service allows users and computers to enroll and renew certificates using HTTPS protocol. A non-enterprise user or a member who is outside the security boundary of the domain can avail this service. The Certificate Enrollment Web Service focusses on automated client requests and processes certificate requests with the help of a native client.

Certificate Enrollment Policy Web Service

The Certificate Enrollment Policy Web Service allows computers and users to retrieve information about their certificate enrollment policy. The certificate enrollment policy gives the location of the CAs and the types of certificates requested from them. Along with the Certificate Enrollment Web Service, this service will allow policy-based web enrollment to a non-domain client or a member outside a domain. The enrollment policy can be enabled using group policy settings or can be applied individually to client computers.

Thus AD CS service provides an efficient way for managing certificate infrastructure for any entity in Windows domain network.

Active Directory Federation Service (ADFS)

What is ADFS?

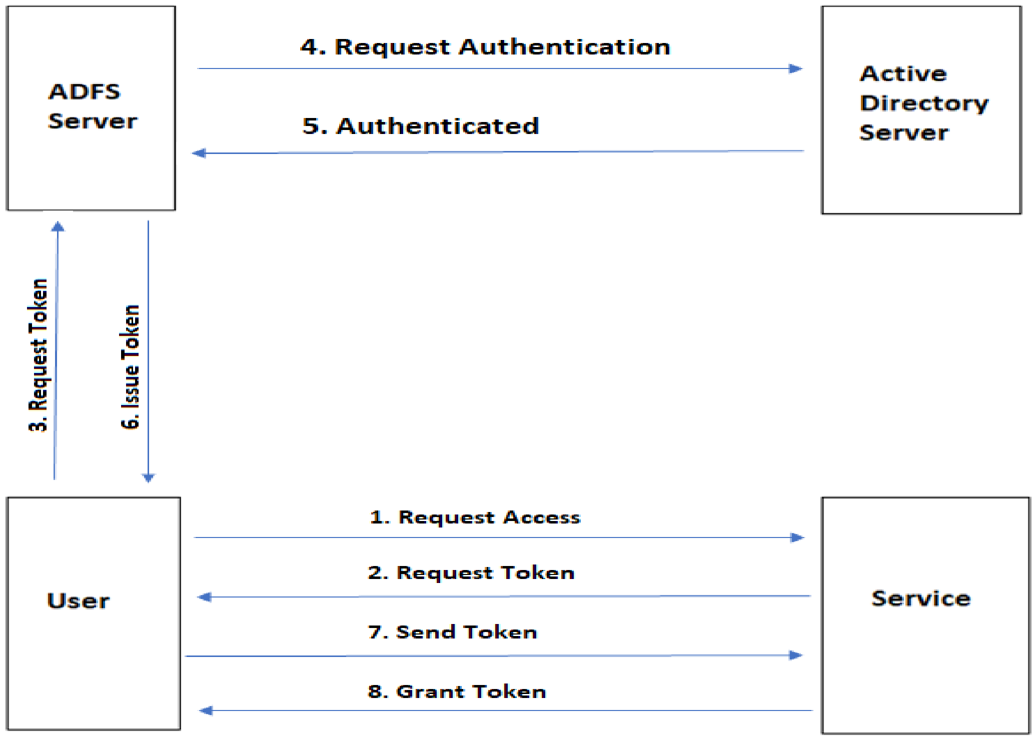
Active Directory Federation Service (ADFS) is a software component developed by Microsoft to provide Single Sign-On (SSO) Authorization service to users on Windows Server Operating Systems. ADFS allows users across organizational boundaries to access applications on Windows Server Operating Systems using a single set of login credentials.

ADFS makes use of claims-based Access Control Authorization model to ensure security across applications using federated identity. Claims-based authentication is a process in which a user is identified by a set of claims related to their identity. The claims are packaged into a secure token by the identity provider.

How does ADFS work?

The authentication process using the Active Directory Federation Service (ADFS), takes place in the following steps:

1. The user navigates to a service, for example, a partner-company website (http://example.com) to obtain pricing or product details.
2. The website requests an authentication token.
3. User requests token from the ADFS server.
4. ADFS server issues token containing users set of claims.
5. User forwards token to the partner-company website.
6. The website grants authorization access to the user.



ADFS Components

- Active Directory: The Identity Information which is to be used by ADFS is stored on the Active Directory.
- Federation Server: It contains the tools needed to manage federated trusts between business partners. It processes authentication requests coming in from external users and hosts a security token service that issues tokens for claims based on verification of credentials from AD.
- Federation Server Proxy: The Proxy is deployed on the extranet of the organization, to which external clients connect when requesting a security token. It forwards these requests to the Federation Server. The Federation server is not exposed directly to the internet to prevent security risks.
- ADFS Web Server: It hosts the ADFS Web Agent which manages the security tokens and authentication cookies sent to it for authentication purposes.

Why ADFS is used by Organizations?

Using Active Directory (AD) in the connected online world creates authentication challenges. AD cannot authenticate users who try to access integrated applications externally. In the modern workplace, users often need to access applications that are not owned or managed by their organization's AD. ADFS is able to resolve and simplify these third-party authentication challenges.

ADFS allows users from one organization to access applications of partner organizations using the standard credentials of their organization's Active Directory (AD). ADFS also lets users access AD-integrated applications while working remotely using their standard organizational AD credentials via a web interface. When establishing a partnership to use another organization's web applications, ADFS provides a central place to manage and audit the employee identity information that is shared with their organization's partners.

Over 90% of organizations use Active Directory, which means many use ADFS as well.

ADFS can be used in the below scenarios:

1. **Single Sign-On (SSO):** ADFS can be used to provide Single Sign-On (SSO) authorization to users who want to access applications located in different networks or organizations. It provides seamless Single Sign-On (SSO) access to Internet-facing applications or services.
2. **Identity Federation (Identity Management):** Federated Identity is a concept where a user's identity is centralized. This makes Identity Management easier. Identity Management is done to maintain security while keeping the costs associated with managing user identities, low.

ADFS Limitations:

- **Maintenance Costs:** ADFS generates a high cost of maintenance which consists of infrastructure maintenance, management of multiple federations, SSL certificate costs.
- **ADFS Complexity:** Adding an application or system to an ADFS service is complex and time-consuming. It doesn’t have a user-friendly management dashboard for managing users, groups and authentication policies.
- **ADFS Security issue:** ADFS runs on Windows Server, that have more security issues like vulnerability to malware and other security related errors.
- ADFS does not allow file sharing or printing using print servers.
- ADFS cannot access Active Directory resources.
- ADFS does not support Remote Desktop connections.

Microsoft Web Application Proxy - ADFS WAP 2016 Server

Microsoft Web Application Proxy [WAP] is a service in Windows Server 2016 that allows you to access web applications from outside your network. WAP functions as a reverse proxy and an Active Directory Federation Services [AD FS] proxy to pre-authenticate user access.

□For the IT organization, it enables you to provide sign on and access control to both modern and legacy applications, on premises and in the cloud, based on the same set of credentials and policies..

□For the user, it provides seamless sign on using the same, familiar account credentials.

□For the developer, it provides an easy way to authenticate users whose identities live in the organizational directory so that you can focus your efforts on your application, not authentication or identity.

Features of Web Application Proxy (WAP) in Server 2016:

1. Pre-authentication for HTTP Basic application

Rich Clients and Smartphones uses ActiveSync Protocol to connect to the Exchange Mailboxes. ActiveSync Protocol uses HTTP Basic Authentication.. Many other Protocols also uses HTTP Basic as the Authorization Protocol. But Web Application Proxy traditionally interacts with AD FS using redirections which is not supported on ActiveSync clients. This new version of Web Application Proxy provides support to publish an app using HTTP basic by enabling the HTTP app to receive a non-claims relying party trust for the application to the Federation Service.

2. Wildcard domain publishing of applications

The external URL for the application can now include a wildcard to enable you to publish multiple applications from within a specific domain, for example, https://*.apps.windowstechpro.com. This will simply works perfect for SharePoint Application publishing.

3. HTTP to HTTPS redirection

In order to make sure your users can access your app, even if they neglect to type HTTPS in the URL, Web Application Proxy now supports HTTP to HTTPS redirection.

4. HTTP Publishing

It is now possible to publish HTTP applications using pass-through pre-authentication

5. Publishing of Remote Desktop Gateway apps

And also it has below **few improvements**

1. New debug log for better troubleshooting and improved service log for complete audit trail and improved error handling,
2. Administrator Console UI improvements
3. Propagation of client IP address to backend applications

Microsoft System Center Configuration Manager (SCCM)

What is SCCM and How it Works?

Microsoft System Center Configuration Manager (SCCM) is a Windows product which enables administrators to manage security and deployment of applications, devices that are part of an Enterprise. System Center is the family or suite of management tools from Microsoft. Organizations would rather purchase System Center Configuration Manager than purchasing a component in the System Center for updating or patching their systems.

How SCCM Works:

Now we will know the step by step procedure on how System Center Configuration Manager (SCCM) works:

Step1: To install the application, create packages in the SCCM console which consists of the command line and executed files.

Step2: Configuration manager admin creates virtual application packages and replicates to selected Distribution Points.

(Distribution points are nothing but file servers, they store the packages for a particular region)

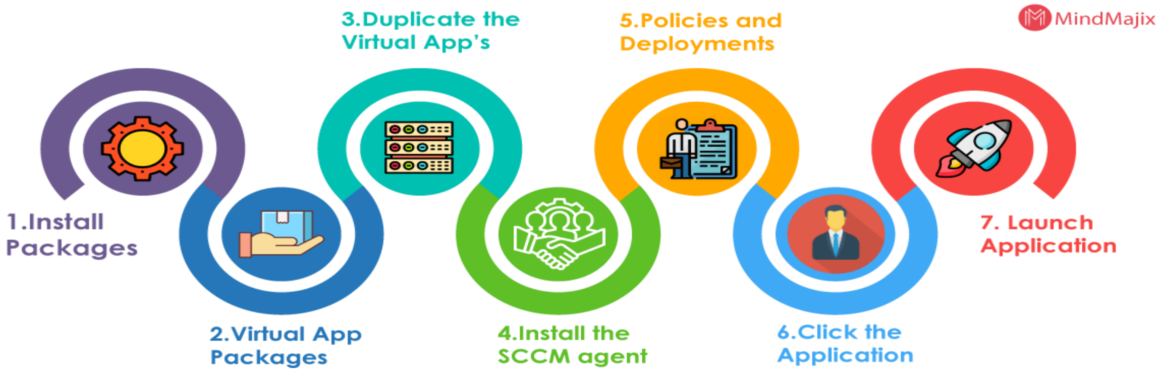
Step3: If the user wants to download any application, then the user can directly download the application from the distribution points rather than connecting to the SCCM primary server.

Step4: Now, install the SCCM agent which helps a machine to communicate with the SCCM servers.

Step5: In this step, the SCCM agent keeps on checking for the new policies and deployments. Using the updates SCCM admin creates deployment where an application is targeted on a bunch of machines.

Step6: Once the policy reached the end machine, the SCCM agent evaluates the policy and reach out to its particular regional distribution points for downloading the packages.

Step7: Once the executed files are downloaded in a temp folder, users can install those packages in the local system. Now the file status sent back to the SCCM server to update in the database.



These are the basic steps to explain how SCCM works, and a lot more additional steps need to be considered in the background. But the core components used in the software distribution (Application packages, Distribution points, SCCM agents, servers) are the same for any infrastructure.

Let us dive into the SCCM concepts one by one.

Systems Management in Enterprise

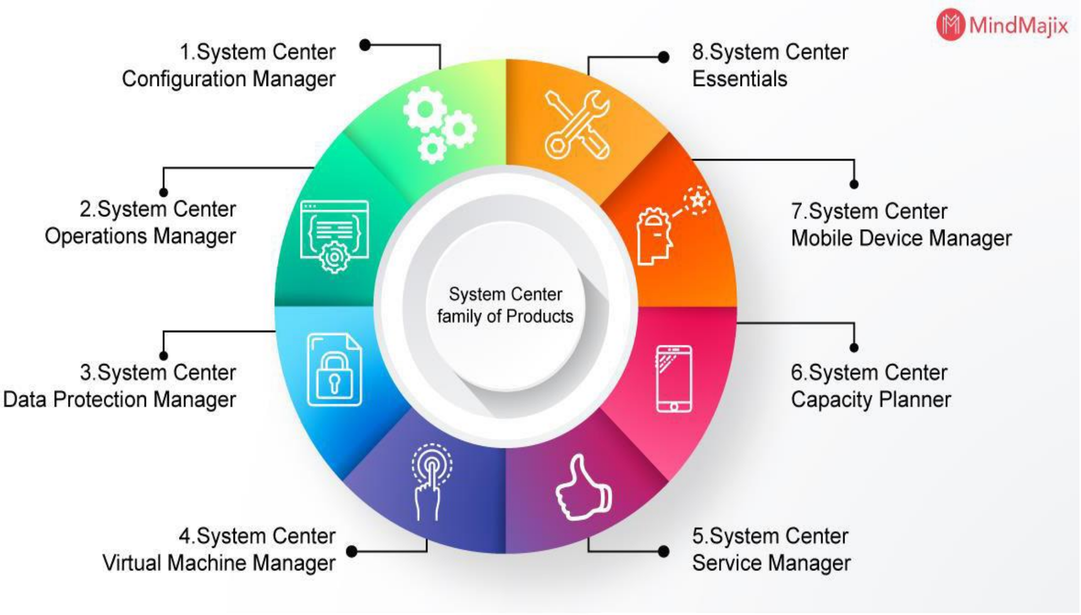
Earlier to the advent of any Systems Management tools, IT departments struggled a lot with the server and client system management. With the tools like Microsoft System Center, patching a computer, imaging workstations, rolling out software, monitoring servers, network devices and backups were all done in a tedious manner. As tools evolved around the systems management, there used to be dedicated servers for these requirements and this had to repeat for another set of requirements. This was all a clumsy process as there was no communication between these separate servers.

To understand this, consider an example where an organization keeps track of assets through one product and have a separate one to put images onto these systems. It has a product to update or patch the systems when required and another one to monitor the system and alert the administrators in any unforeseen situations. Finally, a different product to backup data and a different product to provide security management of the system also exist. Having said this, Microsoft was in a situation like this for about 5 to 8 years when all of these were handled via different products.

After many years, Microsoft had put all of these products into a single suite of products called the System Center and spent enough time to get all of these products to work together. Now, an organization which wants to buy a new license can actually buy a suite license to work with all these products under a single umbrella and leverage benefits out of these products for their own enterprises. The section focuses on bringing in a product as like System Center which can handle all the activities of a system from imaging, deployment, patching, updating, maintenance, support, and retire under a single life-cycle management tool.

System Center family of Products

There are many products that constitute System Center, and the whole suite complements each other with their functionalities. Based on the licenses that are purchased, organizations can work along with more than one of these products or tools within their Enterprise. With each successful release, more and more functionalities and capabilities are added which help each other. Let us now take a look at each of these products individually to see their functionality set:


MindMajix
1.System Center Configuration Manager
2.System Center Operations Manager
3.System Center Data Protection Manager
4.System Center Virtual Machine Manager
5.System Center Service Manager
6.System Center Capacity Planner
7.System Center Mobile Device Manager
8.System Center Essentials
System Center family of Products

1. System Center Configuration Manager

System Center Configuration Manager (SCCM) comes with the ability of imaging and installing the base operating system on a system based on the configuration provided. Once an operating system in installed, SCCM kicks in to update or patch the system. It keeps track of the system inventory and remote control capabilities. It enables IT, administrators, to keep up with the system configuration of all the machines based on a single and common organizational configuration.

2. System Center Operations Manager

SCCM is the product that lays down the base configuration of a system and keeps it updated and patched. System Center Operations Manager then takes over the responsibility of monitoring the health of the system along with all other applications installed on that specific system. There are specific set of rules that track down the normal functioning of the system, and if there are any deviations, the necessary personnel is notified of the changes.

3. System Center Data Protection Manager

Data Protection Manager (DPM) comes in handy when SCOM reports any faults on a physical machine. DPM helps in recovery from the backups that it holds. DPM takes backups of the server file system, SharePoint data, exchange databases, SQL databases on a standard schedule. This helps in recovering a system by full data recovery which is either corrupted or damaged.

4. System Center Virtual Machine Manager

There is a shift of organization’s physical systems to virtual systems for a development, maintenance, and production, and hence comes a tool that handles all the life cycle-related activities for the virtual machines - System Center Virtual Machine Manager (VMM). If there is an instance where a physical or a virtual system is about to fail, SCOM can trigger the automatic creation of a new session using SCCM and Hyper-V to build a new virtual system. VMM also helps in transferring the operating system, application, and data to a virtual machine in an automated Physical To Virtual (P2V) process.

5. System Center Service Manager

Most of the tools from the System Center suite of products revolve around the IT related tasks such as patching, imaging, monitoring, backups - there are other organizational needs such as managing processes and change control. System Center Service Manager (SCSM) is an incident management and change control system which integrates with SCCM and the like seamlessly. It helps in logging all the issues identified with these tools and gathers all the details around the issue for a one-point reference to the Desk personnel or the Support personnel.

6. System Center Capacity Planner

With the growing needs of an organization, there is always a need to upgrade the infrastructure for an organization. System Center Capacity Planner helps in identifying and testing performance demands from the current setup and plan for the future requirements aptly. Based on the current requirement, it helps in identifying the relative requirements on the hardware to meet the performance demands for your organization.

7. System Center Mobile Device Manager

Organizations run on Servers and Clients for their related operations, but with the advent of smartphones with equal computing power, mobile devices also have joined the bandwagon for operations carried out in organizations. System Center Mobile Device Manager (MDM) joins hands with System Center Configuration Manager (SCCM) to handle all the life cycle stages from inception to completion for all mobile devices and in simple words, MDM is to mobile devices what SCCM is for servers. Provisioning, monitoring, updating, securing, wiping the devices are all the activities that can be done with MDM.

8. System Center Essentials

Not every organization might have a dedicated IT wing to handle all the system, server related stuff (organizations with less than 500 users or 50 servers). Microsoft provides System Center Essentials which enables management functions related to tracking inventory, patching and updating these systems, monitoring, deploying newer software.

All of these can be done from just this single tool, helping them to scale on their system administration capabilities.

Major & Basic Features of System Centre Configuration Manager (SCCM)

In this section, let us try and understand the major features that are provided by System Center Configuration Manager (SCCM).

1. Operating System deployment:

Installation of the core Operating System is the very first step that needs to be done to initiate the life-cycle for a server altogether. SCCM provides all the tools an organization require for Operating system deployment - either via the imaged installation or as a scripted method of installation.

2. Patching & Updating*:*

When the installation of Operating system is completed successfully, SCCM initiates patching and updating these systems. Most of the organizations rely on the free service (Windows Server Update Services) to patch and update the systems but SCCM leverages everything that WSUS provides and over that, provides the IT administrators an active patching and updating in addition to WSUS. The active update system enforces updates, forces systems to be patched or updated and later rebooted following the IT guidelines published by organizations.

3. Asset Tracking:

Once a system has been created with the Operating system that is required, and later updated, patched, such systems need to be kept in track of further timely updates or patches. SCCM includes the tools that are required to keep track of the hardware, software assets of the system that it is managing altogether.

4. Remote Control:

If a user or a system encounters an issue which might require further assistance of an IT administrator, there is a provision to take remote access of the system to analyze the problem. SCCM has a remote control process that allows an IT administrator or a support engineer to access the system remotely.

5. Software deployment:

Installing the core operating system on a physical/virtual machine is one part and the other part is the additional softwares that are required on a system. SCCM provides a tool that allows to install a simple plugin or a complex suite of applications with unique application configuration. This is one of a kind functionality that makes it more suitable for organizations where certain IT guidelines can be implemented without halting anything.

6. Desired Configuration Management:

This is the other feature that follows the IT guidelines outlaid by an organization where the standard configuration of a system cannot be altered. This ensures that the system has the same software setup, updates, drivers and configuration settings across all the systems. Desired Configuration Management (DCM) tool within SCCM ensures the stringent audit constraints are met and compliance is maintained.

7. Internet Client:

This is a significant component on the SCCM tool which enables devices like remote systems or mobile devices be accessed remotely without specifically bringing them into the VPN network for any maintenance requirements. This can now happen via an Internet Client and a PKI (Public Key Infrastructure) certificate installed on the system. With these prerequisites, SCCM will be able to connect to that device anywhere in the world automatically to inventory, patch, update, monitor the system.

8. Reporting:

SCCM provides an out of the box integration with a report generation tool that generates reports based on the requirements outlaid by the IT administrators. These reports may vary based on the requirement like report of systems that have missed the patches or updates, report of standard configuration, inventory reports, etc.

Business Solutions addressed by SCCM

System Center Configuration Manager (SCCM) helps an organization maintain consistency in the system configuration and management across all the systems. Rather than having to build a workstation or a server manually and individually, SCCM makes use of the templates to build these systems pretty quick. IT personnel can create these templates based on the guidelines outlaid and also to meet the requirements of the organization. In the case of template-based installation, organizations can very well depend on the consistency in the build configuration for all the hardware systems throughout the enterprise.

SCCM in conjunction with other components ensures achieving different functionalities. One of the best examples of such a component is System Center Operations Manager (SCOM). System Center Operations Manager (SCOM) along with System Center Configuration Manager (SCCM) helps an organization stay ahead and proactive to identify issues, faults on time and helps take necessary actions to minimize the downtime on any issues. These tools also help recover systems that have failed for various other reasons with the help of a tool called Data Protection Manager (DPM). It also enables monitoring of the normal operations of the available set of servers, workstations, and applications.

There are policies that are established to update systems of a specific functional role be updated or patched at the same time. This is a feature that is provided by one of the SCCM components called the Desired Configuration Management (DCM). It ensures specific updates are pushed to systems that meet a functional role. This further helps in ensuring all the audit requirements, and also in maintaining compliance at an organization level. This helps in answering all the questions related to audits and compliance requirements with just reports and nothing at all.

New look of SCCM

If you are well aware of the SCCM tool altogether, then you would be able to appreciate what has been developed and released in the new releases. If you are not aware of the tool anyway, then the following few points should be good enough to appreciate what is available in the latest releases. Let us take a closer look at the following points then:

1. User focus:

IT consumerization is the fact of day and resistance against this will not allow an organization to scale further. With more and more devices being available in the market, there is always an expectation to support all of these. As SCCM has always been about systems management, considering the changing landscape, user has been given all the attention that it requires. This allows them to gain more control over the software that is installed. An example of this is the definition of user's working hours and based on these timings, the upgrades and patches are applied on the system.

There can be more one device tagged to a single user, meaning that there can be more than one primary user for every device that is being worked upon. These relationships are handled using the User Device Affinity (UDA). Users can manage their own systems using a new interface called the Software Center. This is more like a shopping cart approach where users search and find what they want to request for installations. Based on the applications, few might be installed right away and few others that require administrative approvals.

2. Role-based Access Control:

Based on the recent trends amongst the products in the industry (in general), there is a growing adoption towards role-based security. This has now been introduced in SCCM 2012 and is controlled by Role-Based Access Control (RBAC) hiding the elements that the user doesn’t have access to. The tasks are grouped into security roles administratively. There are few roles provided with the tool and, in addition to that, business-specific roles and scopes will be added later.

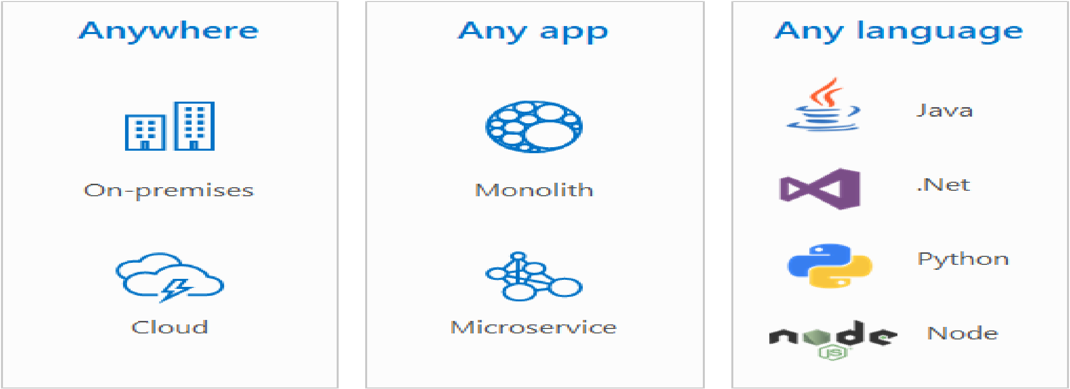
The multilayer approach helps you leverage the power of cloud, and at the same time protecting on-premise clients from any possible potential threats from the internet. SCCM 2012 comes with a new console altogether. This no longer relies on Microsoft Management Console (MMC).

3. Smartphone support:

System Center Mobile Device Manager (MDM) 2008 wasn’t exactly a success but its functionality was rebuilt into SCCM 2012. Support for iPhone, Android, and Windows phones was covered through the Exchange Active-Sync connector.

Windows and containers

Containers are a technology for packaging and running Windows and Linux applications across diverse environments on-premises and in the cloud. Containers provide a lightweight, isolated environment that makes apps easier to develop, deploy, and manage. Containers start and stop quickly, making them ideal for apps that need to rapidly adapt to changing demand. The lightweight nature of containers also make them a useful tool for increasing the density and utilization of your infrastructure.



The Microsoft container ecosystem

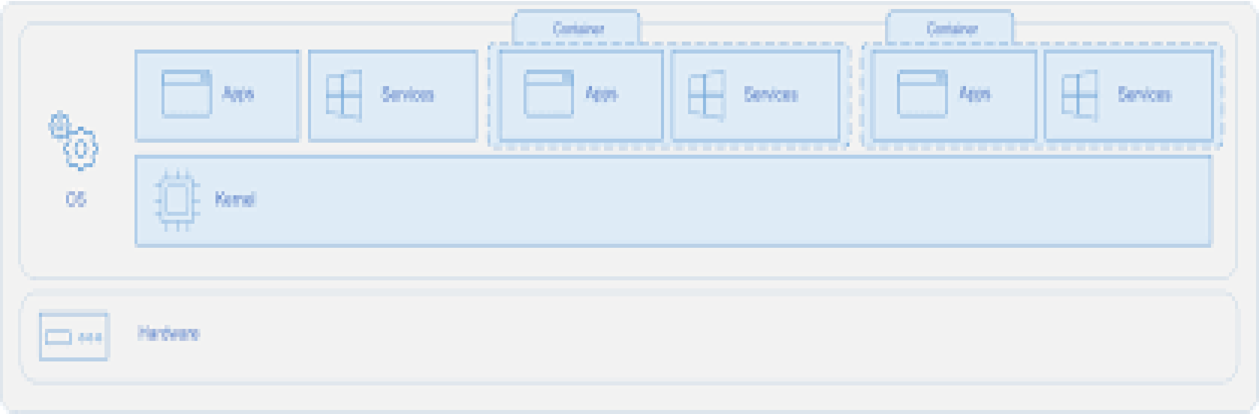
Microsoft provides a number of tools and platforms to help you develop and deploy apps in containers:

- **Run Windows-based or Linux-based containers on Windows 10** for development and testing using Docker Desktop, which makes use of containers functionality built-in to Windows. You can also run containers natively on Windows Server.
- **Develop, test, publish, and deploy Windows-based containers** using the powerful container support in Visual Studio and Visual Studio Code, which include support for Docker, Docker Compose, Kubernetes, Helm, and other useful technologies.
- **Publish your apps as container images** to the public DockerHub for others to use, or to a private Azure Container Registry for your org's own development and deployment, pushing and pulling directly from within Visual Studio and Visual Studio Code.

- **Deploy containers at scale on Azure** or other clouds:
    - Pull your app (container image) from a container registry, such as the Azure Container Registry, and then deploy and manage it at scale using an orchestrator such as Azure Kubernetes Service (AKS) (in preview for Windows-based apps) or Azure Service Fabric.
    - Azure Kubernetes Service deploys containers to Azure virtual machines and manages them at scale, whether that's dozens of containers, hundreds, or even thousands. The Azure virtual machines run either a customized Windows Server image (if you're deploying a Windows-based app), or a customized Ubuntu Linux image (if you're deploying a Linux-based app).
- **Deploy containers on-premises** by using Azure Stack with the AKS Engine (in preview with Linux containers) or Azure Stack with OpenShift. You can also set up Kubernetes yourself on Windows Server (see Kubernetes on Windows), and we're working on support for running Windows containers on RedHat OpenShift Container Platform as well.

How containers work

A container is an isolated, lightweight silo for running an application on the host operating system. Containers build on top of the host operating system's kernel (which can be thought of as the buried plumbing of the operating system), as shown in this diagram.



While a container shares the host operating system's kernel, the container doesn't get unfettered access to it. Instead, the container gets an isolated–and in some cases virtualized–view of the system. For example, a container can access a virtualized version of the file system and registry, but any changes affect only the container and are discarded when it stops. To save data, the container can mount persistent storage such as an Azure Disk or a file share (including Azure Files).

A container builds on top of the kernel, but the kernel doesn't provide all of the APIs and services an app needs to run–most of these are provided by system files (libraries) that run above the kernel in user mode. Because a container is isolated from the host's user mode environment, the container needs its own copy of these user mode system files, which are packaged into something known as a base image. The base image serves as the foundational layer upon which your container is built, providing it with operating system services not provided by the kernel. But we'll talk more about container images later.

Containers vs. virtual machines

In contrast to a container, a virtual machine (VMs) runs a complete operating system–including its own kernel.

Containers and virtual machines each have their uses–in fact, many deployments of containers use virtual machines as the host operating system rather than running directly on the hardware, especially when running containers in the cloud.

Container images

All containers are created from container images. Container images are a bundle of files organized into a stack of layers that reside on your local machine or in a remote container registry. The container image consists of the user mode operating system files needed to support your app, your app, any runtimes or dependencies of your app, and any other miscellaneous configuration file your app needs to run properly.

Microsoft offers several images (called base images) that you can use as a starting point to build your own container image:

- **Windows** - contains the full set of Windows APIs and system services (minus server roles).
- **Windows Server Core** - a smaller image that contains a subset of the Windows Server APIs–namely the full .NET framework. It also includes most server roles, though sadly to few, not Fax Server.
- **Nano Server** - the smallest Windows Server image, with support for the .NET Core APIs and some server roles.
- **Windows 10 IoT Core** - a version of Windows used by hardware manufacturers for small Internet of Things devices that run ARM or x86/x64 processors.

As mentioned earlier, container images are composed of a series of layers. Each layer contains a set of files that, when overlaid together, represent your container image. Because of the layered nature of containers, you don't have to always target a base image to build a Windows container. Instead, you could target another image that already carries the framework you want. For example, the .NET team publishes a .NET core image that carries the .NET core runtime. It saves users from needing to duplicate the process of installing .NET core–instead they can reuse the layers of this container image. The .NET core image itself is built based upon Nano Server.

Container users

Containers for developers

Containers help developers build and ship higher-quality apps, faster. With containers, developers can create a container image that deploys in seconds, identically across environments. Containers act as an easy mechanism to share code across teams and to bootstrap a development environment without impacting your host file system.

Containers are portable and versatile, can run apps written in any language, and they're compatible with any machine running Windows 10, version 1607 or later, or Windows Server 2016 or later. Developers can create and test a container locally on their laptop or desktop, and then deploy that same container image to their company's private cloud, public cloud, or service provider. The natural agility of containers supports modern app development patterns in large-scale, virtualized cloud environments.

Containers for IT professionals

Containers help admins create infrastructure that's easier to update and maintain, and that more fully utilizes hardware resources. IT professionals can use containers to provide standardized environments for their development, QA, and production teams. By using containers, systems administrators abstract away differences in operating system installations and the underlying infrastructure.

Container Orchestration

Orchestrators are a critical piece of infrastructure when setting up a container-based environment. While you can manage a few containers manually using Docker and Windows, apps often make use of five, ten, or even hundreds of containers, which is where orchestrators come in.

Container orchestrators were built to help manage containers at scale and in production. Orchestrators provide functionality for:

- Deploying at scale
- Workload scheduling
- Health monitoring
- Failing over when a node fails
- Scaling up or down
- Networking
- Service discovery
- Coordinating app upgrades
- Cluster node affinity

There are many different orchestrators that you can use with Windows containers; here are the options Microsoft provides:

- Azure Kubernetes Service (AKS) - use a managed Azure Kubernetes service
- Azure Service Fabric - use a managed service
- Azure Stack with the AKS Engine - use Azure Kubernetes Service on-premises
- Kubernetes on Windows - set up Kubernetes yourself on Windows

Network Load Balancing

In this topic, we provide you with an overview of the Network Load Balancing (NLB) feature in Windows Server 2016. You can use NLB to manage two or more servers as a single virtual cluster. NLB enhances the availability and scalability of Internet server applications such as those used on web, FTP, firewall, proxy, virtual private network (VPN), and other mission-critical servers.

Note

Windows Server 2016 includes a new Azure-inspired Software Load Balancer (SLB) as a component of the Software Defined Networking (SDN) infrastructure. Use SLB instead of NLB if you are using SDN, are using non-Windows workloads, need outbound network address translation (NAT), or need Layer 3 (L3) or non-TCP based load balancing. You can continue to use NLB with Windows Server 2016 for non-SDN deployments. For more information about SLB, see **Software Load Balancing (SLB) for SDN**.

The Network Load Balancing (NLB) feature distributes traffic across several servers by using the TCP/IP networking protocol. By combining two or more computers that are running applications into a single virtual cluster, NLB provides reliability and performance for web servers and other mission-critical servers.

The servers in an NLB cluster are called *hosts*, and each host runs a separate copy of the server applications. NLB distributes incoming client requests across the hosts in the cluster. You can configure the load that is to be handled by each host. You can also add hosts dynamically to the cluster to handle increased load. NLB can also direct all traffic to a designated single host, which is called the *default host*.

NLB allows all of the computers in the cluster to be addressed by the same set of IP addresses, and it maintains a set of unique, dedicated IP addresses for each host. For load-balanced applications, when a host fails or goes offline, the load is automatically redistributed among the computers that are still operating. When it is ready, the offline computer can transparently rejoin the cluster and regain its share of the workload, which allows the other computers in the cluster to handle less traffic.

Practical applications

NLB is useful for ensuring that stateless applications, such as web servers running Internet Information Services (IIS), are available with minimal downtime, and that they are scalable (by adding additional servers as the load increases). The following sections describe how NLB supports high availability, scalability, and manageability of the clustered servers that run these applications.

High availability

A high availability system reliably provides an acceptable level of service with minimal downtime. To provide high availability, NLB includes built-in features that can automatically:

- Detect a cluster host that fails or goes offline, and then recover.
- Balance the network load when hosts are added or removed.
- Recover and redistribute the workload within ten seconds.

Scalability

Scalability is the measure of how well a computer, service, or application can grow to meet increasing performance demands. For NLB clusters, scalability is the ability to incrementally add one or more systems to an existing cluster when the overall load of the cluster exceeds its capabilities. To support scalability, you can do the following with NLB:

- Balance load requests across the NLB cluster for individual TCP/IP services.
- Support up to 32 computers in a single cluster.
- Balance multiple server load requests (from the same client or from several clients) across multiple hosts in the cluster.
- Add hosts to the NLB cluster as the load increases, without causing the cluster to fail.
- Remove hosts from the cluster when the load decreases.
- Enable high performance and low overhead through a fully pipelined implementation. Pipelining allows requests to be sent to the NLB cluster without waiting for a response to a previous request.

Manageability

To support manageability, you can do the following with NLB:

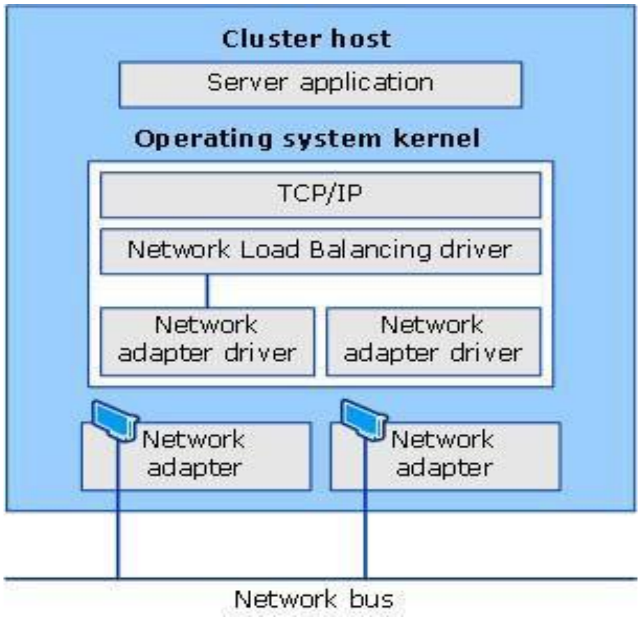
- Manage and configure multiple NLB clusters and the cluster hosts from a single computer by using NLB Manager or the Network Load Balancing (NLB) Cmdlets in Windows PowerShell.
- Specify the load balancing behavior for a single IP port or group of ports by using port management rules.
- Define different port rules for each website. If you use the same set of load-balanced servers for multiple applications or websites, port rules are based on the destination virtual IP address (using virtual clusters).
- Direct all client requests to a single host by using optional, single-host rules. NLB routes client requests to a particular host that is running specific applications.
- Block undesired network access to certain IP ports.
- Enable Internet Group Management Protocol (IGMP) support on the cluster hosts to control switch port flooding (where incoming network packets are sent to all ports on the switch) when operating in multicast mode.
- Start, stop, and control NLB actions remotely by using Windows PowerShell commands or scripts.
- View the Windows Event Log to check NLB events. NLB logs all actions and cluster changes in the event log.

Important functionality

NLB is installed as a standard Windows Server networking driver component. Its operations are transparent to the TCP/IP networking stack. The following figure shows the relationship between NLB and other software components in a typical configuration. Following are the primary features of NLB.

- Requires no hardware changes to run.
- Provides Network Load Balancing Tools to configure and manage multiple clusters and all of the hosts from a single remote or local computer.
- Enables clients to access the cluster by using a single, logical Internet name and virtual IP address, which is known as the cluster IP address (it retains individual

names for each computer). NLB allows multiple virtual IP addresses for multihomed servers.



- Enables NLB to be bound to multiple network adapters, which enables you to configure multiple independent clusters on each host. Support for multiple network adapters differs from virtual clusters in that virtual clusters allow you to configure multiple clusters on a single network adapter.
- Requires no modifications to server applications so that they can run in an NLB cluster.
- Can be configured to automatically add a host to the cluster if that cluster host fails and is subsequently brought back online. The added host can start handling new server requests from clients.
- Enables you to take computers offline for preventive maintenance without disturbing the cluster operations on the other hosts.

Hardware requirements

Following are the hardware requirements to run an NLB cluster.

- All hosts in the cluster must reside on the same subnet.
- There is no restriction on the number of network adapters on each host, and different hosts can have a different number of adapters.

- Within each cluster, all network adapters must be either multicast or unicast. NLB does not support a mixed environment of multicast and unicast within a single cluster.
- If you use the unicast mode, the network adapter that is used to handle client-to-cluster traffic must support changing its media access control (MAC) address.

Software requirements

Following are the software requirements to run an NLB cluster.

- Only TCP/IP can be used on the adapter for which NLB is enabled on each host. Do not add any other protocols (for example, IPX) to this adapter.
- The IP addresses of the servers in the cluster must be static.

Note

NLB does not support Dynamic Host Configuration Protocol (DHCP). NLB disables DHCP on each interface that it configures.

VPN

VPN stands for **"Virtual Private Network"** and describes the opportunity to establish a protected network connection when using public networks. VPNs encrypt your internet traffic and disguise your online identity. This makes it more difficult for third parties to track your activities online and steal data. The encryption takes place in **real time**.

How does a VPN work?

A VPN hides your IP address by letting the network redirect it through a specially configured remote server run by a VPN host. This means that if you surf online with a VPN, the VPN server becomes the source of your data. This means your Internet Service Provider (ISP) and other third parties cannot see which websites you visit or what data you send and receive online. A VPN works like a filter that turns all your data into "gibberish". Even if someone were to get their hands on your data, it would be useless.

What are the benefits of a VPN connection?

A VPN connection disguises your data traffic online and protects it from external access. Unencrypted data can be viewed by anyone who has network access and wants to see it. With a VPN, hackers and cyber criminals can’t decipher this data.

Secure encryption: To read the data, you need an *encryption key* . Without one, it would take millions of years for a computer to decipher the code in the event of a brute force attack . With the help of a VPN, your online activities are hidden even on public networks.

Disguising your where abouts : VPN servers essentially act as your proxies on the internet. Because the demographic location data comes from a server in another country, your actual location cannot be determined. In addition, most VPN services do not store logs of your activities. Some providers, on the other hand, record your behavior, but do not pass this information on to third parties. This means that any potential record of your user behavior remains permanently hidden.

Access to regional content: Regional web content is not always accessible from everywhere. Services and websites often contain content that can only be accessed from certain parts of the world. Standard connections use local servers in the country to determine your location. This means that you cannot access content at home while traveling, and you cannot access international content from home. With **VPN location spoofing** , you can switch to a server to another country and effectively “change” your location.

Secure data transfer: If you work remotely, you may need to access important files on your company’s network. For security reasons, this kind of information requires a secure connection. To gain access to the network, a VPN connection is often required. VPN services connect to private servers and use encryption methods to reduce the risk of data leakage.

Why should you use a VPN connection?

Your ISP usually sets up your connection when you connect to the internet. It tracks you via an IP address. Your network traffic is routed through your ISP's servers, which can log and display everything you do online.

Your ISP may seem trustworthy, but it may share your browsing history with advertisers, the police or government, and/or other third parties. ISPs can also fall victim to attacks by cyber criminals: If they are hacked, your personal and private data can be compromised.

This is especially important if you regularly connect to public Wi-Fi networks. You never know who might be monitoring your internet traffic and what they might steal from you, including passwords, personal data, payment information, or even your entire identity.

What should a good VPN do?

You should rely on your VPN to perform one or more tasks. The VPN itself should also be protected against compromise. These are the features you should expect from a comprehensive VPN solution:

- **Encryption of your IP address:** The primary job of a VPN is to hide your IP address from your ISP and other third parties. This allows you to send and receive information online without the risk of anyone but you and the VPN provider seeing it.
- **Encryption of protocols:** A VPN should also prevent you from leaving traces, for example, in the form of your internet history, search history and cookies. The encryption of cookies is especially important because it prevents third parties from gaining access to confidential information such as personal data, financial information and other content on websites.
- **Kill switch:** If your VPN connection is suddenly interrupted, your secure connection will also be interrupted. A good VPN can detect this sudden downtime and terminate preselected programs, reducing the likelihood that data is compromised.
- **Two-factor authentication:** By using a variety of authentication methods, a strong VPN checks everyone who tries to log in. For example, you might be prompted to enter a password, after which a code is sent to your mobile device. This makes it difficult for uninvited third parties to access your secure connection.

The history of VPNs

Since humans have been using the internet, there has been a movement to protect and encrypt internet browser data. The US Department of Defense already got involved in projects working on the encryption of internet communication data back in the 1960s.

The predecessors of the VPN

Their efforts led to the creation of **ARPANET** (Advanced Research Projects Agency Network), a packet switching network, which in turn led to the development of the Transfer Control Protocol/Internet Protocol (TCP/IP).

The **TCP/IP** had four levels: **Link, internet, transport and application**. At the internet level, local networks and devices could be connected to the universal network – and this is where the risk of exposure became clear. In 1993, a team from Columbia University and AT&T Bell Labs finally succeeded in creating a kind of first version of the modern VPN, known as swIPe: Software IP encryption protocol.

In the following year, Wei Xu developed the IPSec network, an internet security protocol that authenticates and encrypts information packets shared online. In 1996, a Microsoft employee named Gurdeep Singh-Pall created a Peer-to-Peer Tunneling Protocol (PPTP).

Early VPNs

Contiguous to Singh-Pall developing PPTP, the internet was growing in popularity and the need for consumer-ready, sophisticated security systems emerged. At that time, anti-virus programs were already effective in preventing malware and spyware from infecting a computer system. However, people and companies also started demanding encryption software that could hide their browsing history on the internet.

The first VPNs therefore started in the early 2000s, but were almost exclusively used by companies. However, after a flood of security breaches, especially in the early 2010s, the consumer market for VPNs started to pick up.

VPNs and their current use

According to the ***GlobalWebIndex***, the number of VPN users worldwide increased more than fourfold between 2016 and 2018. In countries such as Thailand, Indonesia and China, where internet use is restricted and censored, **one in fiveinternet users** uses a VPN. In the USA, Great Britain and Germany, the proportion of VPN users is **lowerat around 5%**, but is growing.

One of the biggest drivers for VPN adoption in recent years has been the increasing demand for content with geographical access restrictions. For example, video streaming services such as Netflix or YouTube make certain videos available only in certain countries. With contemporary VPNs, you can encrypt your IP address so that you appear to be surfing from another country, enabling you to access this content from anywhere.

Here’s how to surf securely with a VPN

A VPN encrypts your surfing behavior, which can only be decoded with the help of a key. Only your computer and the VPN know this key, so your ISP cannot recognize where you are surfing. Different VPNs use different encryption processes, but generally function in three steps:

1. Once you are online, start your VPN. The VPN acts as a secure tunnel between you and the internet. Your ISP and other third parties cannot detect this tunnel.
2. Your device is now on the local network of the VPN, and your IP address can be changed to an IP address provided by the VPN server.
3. You can now surf the internet at will, as the VPN protects all your personal data.

What kind of VPNs are there?

There are many different types of VPNs, but you should definitely be familiar with the three main types:

SSL VPN

Often not all employees of a company have access to a company laptop they can use to work from home. During the corona crisis in Spring 2020, many companies faced the problem of not having enough equipment for their employees. In such cases, use of a private device (PC, laptop, tablet, mobile phone) is often resorted to. In this case, companies fall back on an **SSL-VPN** solution, which is usually implemented via a corresponding hardware box.

The prerequisite is usually an HTML-5-capable browser, which is used to call up the company's login page. HTML-5 capable browsers are available for virtually any operating system. Access is guarded with a username and password.

Site-to-site VPN

A **site-to-site VPN** is essentially a private network designed to hide private intranets and allow users of these secure networks to access each other's resources.

A site-to-site VPN is useful if you have multiple locations in your company, each with its own local area network (LAN) connected to the WAN (Wide Area Network). Site-to-site VPNs are also useful if you have two separate intranets between which you want to send files without users from one intranet explicitly accessing the other.

Site-to-site VPNs are mainly used in large companies. They are complex to implement and do not offer the same flexibility as SSL VPNs. However, they are the most effective way to ensure communication within and between large departments.

Client-to-Server VPN

Connecting via a **VPN client** can be imagined as if you were connecting your home PC to the company with an extension cable. Employees can dial into the company network from their home office via the secure connection and act as if they were sitting in the office. However, a VPN client must first be installed and configured on the computer.

This involves the user not being connected to the internet via his own ISP, but establishing a direct connection through his/her VPN provider. This essentially shortens the tunnel phase of the VPN journey. Instead of using the VPN to create an encryption tunnel to disguise the existing internet connection, the VPN can automatically encrypt the data before it is made available to the user.

This is an increasingly common form of VPN, which is particularly useful for providers of insecure public WLAN. It prevents third parties from accessing and compromising the network connection and encrypts data all the way to the provider. It also prevents ISPs from accessing data that, for whatever reason, remains unencrypted and bypasses any restrictions on the user's internet access (for instance, if the government of that country restricts internet access).

The advantage of this type of VPN access is greater efficiency and universal access to company resources. Provided an appropriate telephone system is available, the employee can, for example, connect to the system with a headset and act as if he/she were at their company workplace. For example, customers of the company cannot even tell whether the employee is at work in the company or in their home office.

How do I install a VPN on my computer?

Before installing a VPN, it is important to be familiar with the different implementation methods:

VPN client

Software must be installed for standalone VPN clients. This software is configured to meet the requirements of the endpoint. When setting up the VPN, the endpoint executes the VPN link and connects to the other endpoint, creating the encryption tunnel. In companies, this step usually requires the entry of a password issued by the company or the installation of an appropriate certificate. By using a password or certificate, the firewall can recognize that this is an authorized connection. The employee then identifies him/herself by means of credentials known to him/her.

Browser extensions

VPN extensions can be added to most web browsers such as Google Chrome and Firefox. Some browsers, including Opera, even have their own integrated VPN extensions. Extensions make it easier for users to quickly switch and configure their VPN while surfing the internet. However, the VPN connection is only valid for information that is shared in this browser. Using other browsers and other internet uses outside the browser (e.g. online games) cannot be encrypted by the VPN.

While browser extensions are not quite as comprehensive as VPN clients, they may be an appropriate option for occasional internet users who want an extra layer of internet security. However, they have proven to be more susceptible to breaches. Users are also advised to choose a reputable extension, as ***data harvesters*** may attempt to use fake VPN extensions. Data harvesting is the collection of personal data, such as what marketing strategists do to create a personal profile of you. Advertising content is then personally tailored to you.

Router VPN

If multiple devices are connected to the same internet connection, it may be easier to implement the VPN directly on the router than to install a separate VPN on each device. A router VPN is especially useful if you want to protect devices with an internet connection that are not easy to configure, such as smart TVs. They can even help you access geographically restricted content through your home entertainment systems.

A router VPN is easy to install, always provides security and privacy, and prevents your network from being compromised when insecure devices log on. However, it may be more difficult to manage if your router does not have its own user interface. This can lead to incoming connections being blocked.

Company VPN

A company VPN is a custom solution that requires personalized setup and technical support. The VPN is usually created for you by the company's IT team. As a user, you have no administrative influence from the VPN itself and your activities and data transfers are logged by your company. This allows the company to minimize the potential risk of data leakage. The main advantage of a corporate VPN is a fully secure connection to the company's intranet and server, even for employees who work outside the company using their own internet connection.

Can I also use a VPN on my smartphone or other devices?

Yes, there are a number of VPN options for smartphones and other internet-connected devices. A VPN can be essential for your mobile device if you use it to store payment information or other personal data or even just to surf the internet. Many VPN providers also offer mobile solutions - many of which can be downloaded directly from Google Play or the Apple App Store, such as Kaspersky VPN Secure Connection.

Is a VPN really so secure?

It is important to note that VPNs do not function like comprehensive anti-virus software. While they protect your IP and encrypt your internet history, a VPN connection does not protect your computer from outside intrusion. To do this, you should definitely use anti-virus software such as Kaspersky Internet Security . Because using a VPN on its own does not protect you from Trojans, viruses, bots or other malware.

Once the malware has found its way onto your device, it can steal or damage your data, whether you are running a VPN or not. It is therefore important that you use a VPN together with a comprehensive anti-virus program to ensure maximum security.

Selecting a secure VPN provider

It is also important that you choose a VPN provider that you can trust. While your ISP cannot see your internet traffic, your VPN provider can. If your VPN provider is compromised, so are you. For this reason, it is crucial that you choose a trusted VPN provider to ensure both the concealment of your internet activities and ensure the highest level of security.

How to install a VPN connection on your smartphone

As already mentioned, there are also VPN connections for Android smartphones and iPhones. Fortunately, smartphone VPN services are easy to use and generally include the following:

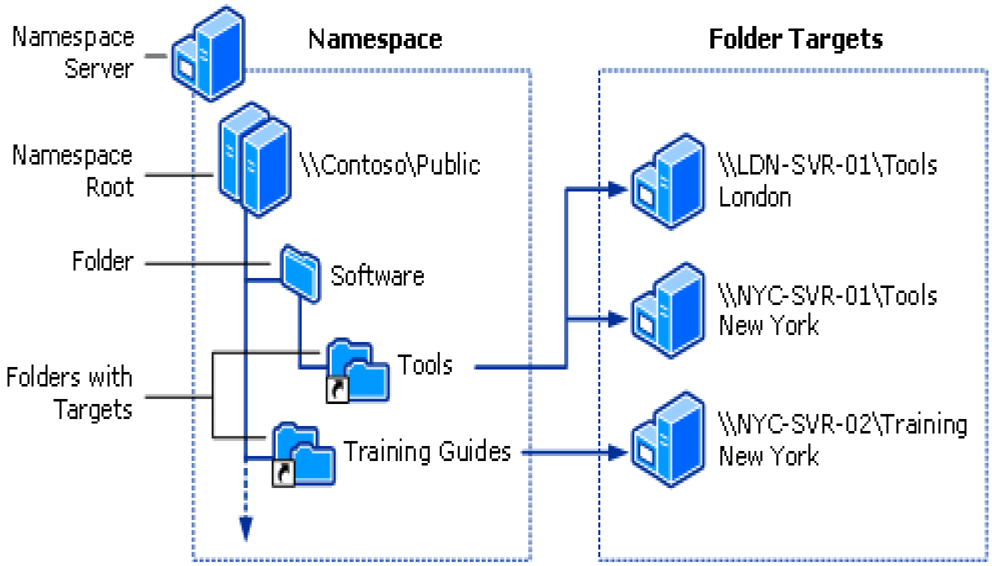
- The installation process usually only downloads one app from the iOS App Store or Google Play Store. Although free VPN providers exist, it’s wise to choose a professional provider when it comes to security.
- The setup is extremely user-friendly, as the default settings are already mostly designed for the average smartphone user. Simply log in with your account. Most apps will then guide you through the key functions of the VPN services.
- Switching on the VPN literally works like a light switch for many VPN apps. You will probably find the option directly on the home screen.
- Server switching is usually done manually if you want to fake your location. Simply select the desired country from the offer.
- Advanced setup is available for users requiring a higher degree of data protection. Depending on your VPN, you can also select other protocols for your encryption method. Diagnostics and other
- Functions may also be available in your app. Before you subscribe, learn about these features to find the right VPN for your needs.
- In order to surf the internet safely from now on, all you have to do is first activate the VPN connection through the app.

But keep the following in mind: A VPN is only as secure as the data usage and storage policies of its provider. Remember that the VPN service transfers your data to their servers and these servers connect over the internet on your behalf. If they store data logs, make sure that it is clear for what purpose these logs are stored. Serious VPN providers usually put your privacy first and foremost. You should therefore choose a trusted provider such as Kaspersky Secure Connection .

Remember that only internet data is encrypted. Anything that does not use a cellular or Wi-Fi connection will not be transmitted over the internet. As a result, your VPN will not encrypt your standard voice calls or texts.

DFS Namespaces overview

DFS Namespaces is a role service in Windows Server that enables you to group shared folders located on different servers into one or more logically structured namespaces. This makes it possible to give users a virtual view of shared folders, where a single path leads to files located on multiple servers, as shown in the following figure:



Here's a description of the elements that make up a DFS namespace:

- **Namespace server** - A namespace server hosts a namespace. The namespace server can be a member server or a domain controller.
- **Namespace root** - The namespace root is the starting point of the namespace. In the previous figure, the name of the root is Public, and the namespace path is \\Contoso\Public. This type of namespace is a domain-based namespace because it begins with a domain name (for example, Contoso) and its metadata is stored in Active Directory Domain Services (AD DS). Although a single namespace server is shown in the previous figure, a domain-based namespace can be hosted on multiple namespace servers to increase the availability of the namespace.
- **Folder** - Folders without folder targets add structure and hierarchy to the namespace, and folders with folder targets provide users with actual content. When users browse a folder
-

- that has folder targets in the namespace, the client computer receives a referral that transparently redirects the client computer to one of the folder targets.
- **Folder targets** - A folder target is the UNC path of a shared folder or another namespace that is associated with a folder in a namespace. The folder target is where data and content is stored. In the previous figure, the folder named Tools has two folder targets, one in London and one in New York, and the folder named Training Guides has a single folder target in New York. A user who browses to \\Contoso\Public\Software\Tools is transparently redirected to the shared folder \\LDN-SVR-01\Tools or \\NYC-SVR-01\Tools, depending on which site the user is currently located in.

This topic discusses how to install DFS, what's new, and where to find evaluation and deployment information.

You can administer namespaces by using DFS Management, the DFS Namespace (DFSN) Cmdlets in Windows PowerShell, the **DfsUtil** command, or scripts that call WMI.

Server requirements and limits

There are no additional hardware or software requirements for running DFS Management or using DFS Namespaces.

A namespace server is a domain controller or member server that hosts a namespace. The number of namespaces you can host on a server is determined by the operating system running on the namespace server.

Servers that are running the following operating systems can host multiple domain-based namespaces in addition to a single stand-alone namespace.

- Windows Server 2019
- Windows Server 2016
- Windows Server 2012 R2
- Windows Server 2012
- Windows Server 2008 R2 Datacenter and Enterprise Editions
- Windows Server (Semi-Annual Channel)

Servers that are running the following operating systems can host a single stand-alone namespace:

- Windows Server 2008 R2 Standard

The following table describes additional factors to consider when choosing servers to host a namespace.

Server Hosting Stand-Alone Namespaces	**Server Hosting Domain-Based Namespaces**
Must contain an NTFS volume to host the namespace.	Must contain an NTFS volume to host the namespace.
Can be a member server or domain controller.	Must be a member server or domain controller in the domain in which the namespace is configured. (This requirement applies to every namespace server that hosts a given domain-based namespace.)
Can be hosted by a failover cluster to increase the availability of the namespace.	The namespace cannot be a clustered resource in a failover cluster. However, you can locate the namespace on a server that also functions as a node in a failover cluster if you configure the namespace to use only local resources on that server.

Network Policy Server (NPS)

Network Policy Server (NPS) allows you to create and enforce organization-wide network access policies for connection request authentication and authorization.

You can also configure NPS as a Remote Authentication Dial-In User Service (RADIUS) proxy to forward connection requests to a remote NPS or other RADIUS server so that you can load balance connection requests and forward them to the correct domain for authentication and authorization.

NPS allows you to centrally configure and manage network access authentication, authorization, and accounting with the following features:

- **RADIUS server**. NPS performs centralized authentication, authorization, and accounting for wireless, authenticating switch, remote access dial-up and virtual private network (VPN) connections. When you use NPS as a RADIUS server, you configure network access servers, such as wireless access points and VPN servers, as RADIUS clients in NPS. You also configure network policies that NPS uses to authorize connection requests, and you can configure RADIUS accounting so that NPS logs accounting information to log files on the local hard disk or in a Microsoft SQL Server database.

- **RADIUS proxy**. When you use NPS as a RADIUS proxy, you configure connection request policies that tell the NPS which connection requests to forward to other RADIUS servers and to which RADIUS servers you want to forward connection requests. You can also configure NPS to forward accounting data to be logged by one or more computers in a remote RADIUS server group

- **RADIUS accounting**. You can configure NPS to log events to a local log file or to a local or remote instance of Microsoft SQL Server.

Windows Server Editions and NPS

NPS provides different functionality depending on the edition of Windows Server that you install.

Windows Server 2016 or Windows Server 2019 Standard/Datacenter Edition

With NPS in Windows Server 2016 Standard or Datacenter, you can configure an unlimited number of RADIUS clients and remote RADIUS server groups. In addition, you can configure RADIUS clients by specifying an IP address range.

Note

The WIndows Network Policy and Access Services feature is not available on systems installed with a Server Core installation option.

The following sections provide more detailed information about NPS as a RADIUS server and proxy.

RADIUS server and proxy

You can use NPS as a RADIUS server, a RADIUS proxy, or both.

RADIUS server

NPS is the Microsoft implementation of the RADIUS standard specified by the Internet Engineering Task Force (IETF) in RFCs 2865 and 2866. As a RADIUS server, NPS performs centralized connection authentication, authorization, and accounting for many types of network access, including wireless, authenticating switch, dial-up and virtual private network (VPN) remote access, and router-to-router connections.

Using NPS as a RADIUS server

You can use NPS as a RADIUS server when:

- You are using an AD DS domain or the local SAM user accounts database as your user account database for access clients.
- You are using Remote Access on multiple dial-up servers, VPN servers, or demand-dial routers and you want to centralize both the configuration of network policies and connection logging and accounting.
- You are outsourcing your dial-up, VPN, or wireless access to a service provider. The access servers use RADIUS to authenticate and authorize connections that are made by members of your organization.
- You want to centralize authentication, authorization, and accounting for a heterogeneous set of access servers.

The following illustration shows NPS as a RADIUS server for a variety of access clients.

RADIUS proxy

As a RADIUS proxy, NPS forwards authentication and accounting messages to NPS and other RADIUS servers. You can use NPS as a RADIUS proxy to provide the routing of RADIUS messages between RADIUS clients (also called network access servers) and RADIUS servers that perform user authentication, authorization, and accounting for the connection attempt.

When used as a RADIUS proxy, NPS is a central switching or routing point through which RADIUS access and accounting messages flow. NPS records information in an accounting log about the messages that are forwarded.

The following illustration shows NPS as a RADIUS proxy between RADIUS clients and RADIUS servers.

With NPS, organizations can also outsource remote access infrastructure to a service provider while retaining control over user authentication, authorization, and accounting.

NPS configurations can be created for the following scenarios:

- Wireless access
- Organization dial-up or virtual private network (VPN) remote access
- Outsourced dial-up or wireless access
- Internet access
- Authenticated access to extranet resources for business partners

PowerShell

Windows PowerShell is an object-oriented automation engine and scripting language. It is designed mainly for IT professionals and system administrators to control & automate the administration of Windows OS and other applications. It provides compelling new concepts to extend the knowledge you have gained and scripts you have created within the Windows Command Prompt and Windows Script Host environments.

Why Use Powershell?

Here, are some important reason for using PowerShell:

- PowerShell offers a well-integrated command-line experience for the operation system
- PowerShell allows complete access to all of the types in the .NET framework
- Trusted by system administrators.
- PowerShell is a simple way to manipulate server and workstation components
- It's geared toward system administrators by creating a more easy syntax
- PowerShell is more secure than running VBScript or other scripting languages

Features of Powershell

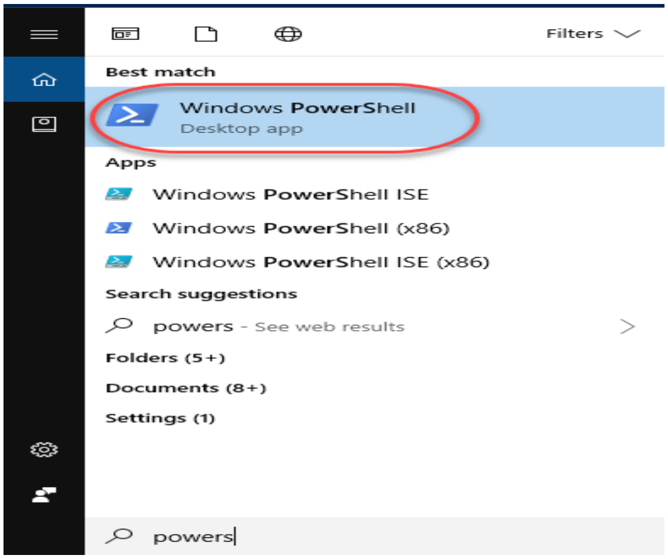
- **PowerShell Remoting**: PowerShell allows scripts and cmdlets to be invoked on a remote machine.
- **Background Jobs**: It helps you to invoked script or pipeline asynchronously. You can run your jobs either on the local machine or multiple remotely operated machines.
- **Transactions**: Enable cmdlet and allows developers to perform
- **Evening:** This command helps you to listen, forwarding, and acting on management and system events.
- **Network File Transfer:** Powershell offers native support for prioritized, asynchronous, throttled, transfer of files between machines using the Background Intelligent Transfer Service (BITS) technology.

How to launch PowerShell

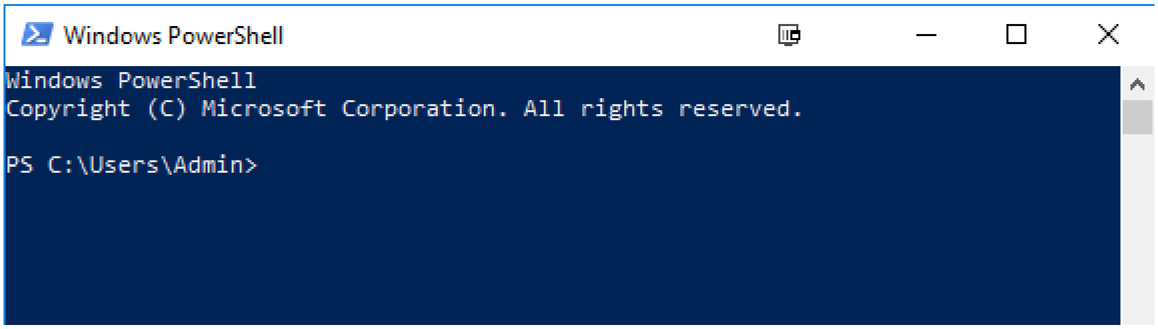
Now in this PowerShell script tutorial, we will learn how to launch PowerShell on Windows OS.

PowerShell is pre-installed in all latest versions of Windows. We need to launch PowerShell for that we need to follow the given steps:

Step 1) Search for PowerShell in Windows. Select and Click



Step 2) Power Shell Window Opens



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\Admin>
```

PowerShell Cmdlet

A cmdlet which is also called Command let is a is a lightweight command used in the Window base PowerShell environment. PowerShell invokes these cmdlets in the command prompt. You can create and invoke cmdlets command using PowerShell APIS.

Cmdlet vs. Command:

Cmdlets are different from commands in other command-shell environments in the following manners ?

- Cmdlets are .NET Framework class objects It can't be executed separately
- Cmdlets can construct from as few as a dozen lines of code
- Parsing, output formatting, and error presentation are not handled by cmdlets
- Cmdlets process works on objects. So text stream and objects can't be passed as output for pipelining
- Cmdlets are record-based as so it processes a single object at a time

Most of the PowerShell functionality comes from Cmdlet's which is always in verb-noun format and not plural. Moreover, Cmdlet's return objects not text. A cmdlet is a series of commands, which is more than one line, stored in a text file with a .ps1 extension.

A cmdlet always consists of a verb and a noun, separated with a hyphen. Some of the verbs use for you to learn PowerShell is:

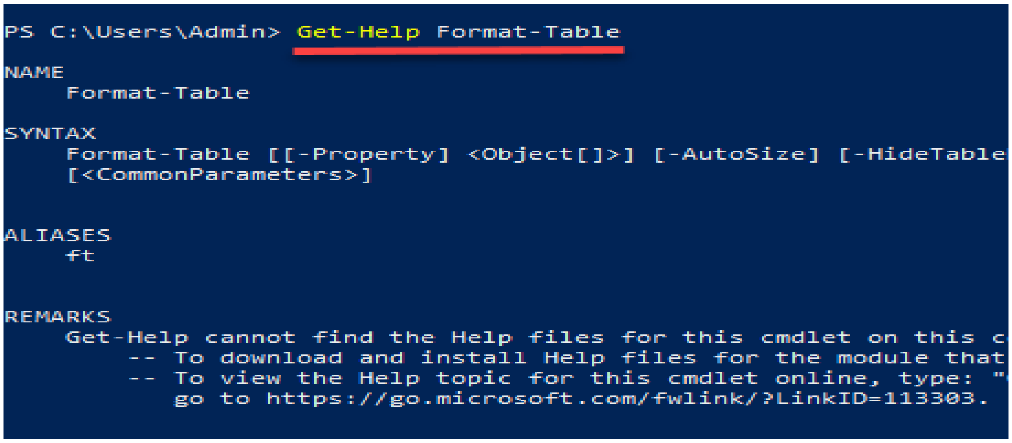
- **Get** — To get something
- **Start** — To run something
- **Out** — To output something
- **Stop** — To stop something that is running
- **Set** — To define something
- **New** — To create something

PowerShell commands

Following is a list of important PowerShell Commands:

Get-Help: Help about PowerShell commands and topics

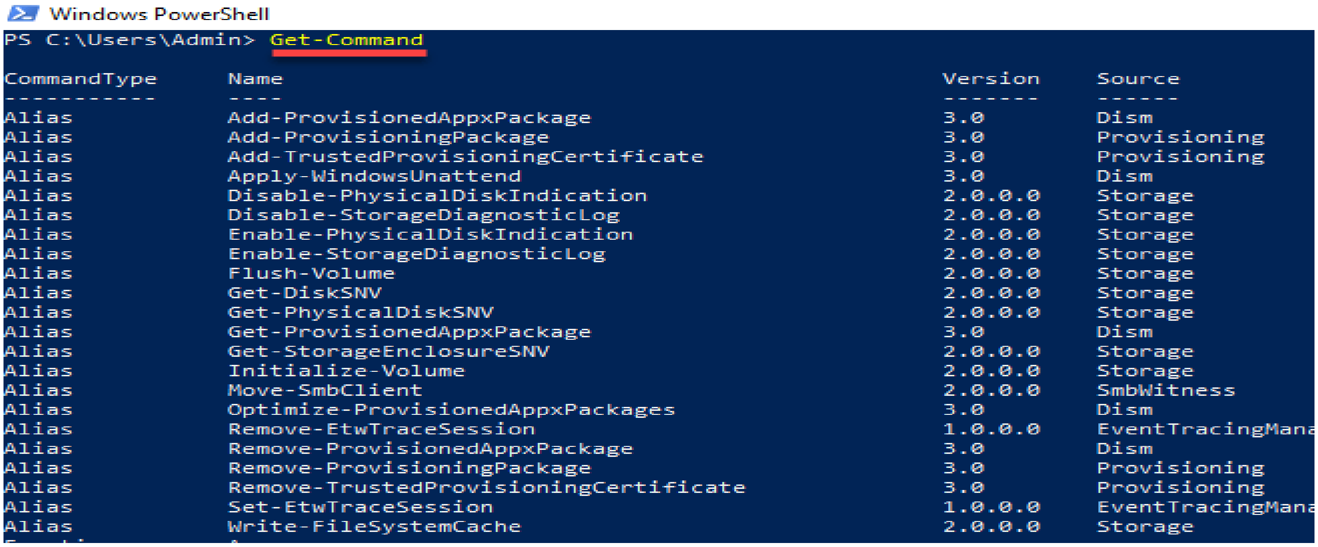
Example: Display help information about the command Format-Table



```
PS C:\Users\Admin> Get-Help Format-Table

NAME
    Format-Table

SYNTAX
    Format-Table [[-Property] <Object[]>] [-AutoSize] [-HideTable
    [<CommonParameters>]


ALIASES
    ft


REMARKS
    Get-Help cannot find the Help files for this cmdlet on this c
        -- To download and install Help files for the module that
        -- To view the Help topic for this cmdlet online, type: "
           go to https://go.microsoft.com/fwlink/?LinkID=113303.
```

Get-Command: Get information about anything that can be invoked

Powershell Script Example: To generate a list of cmdlets, functions installed in your machine

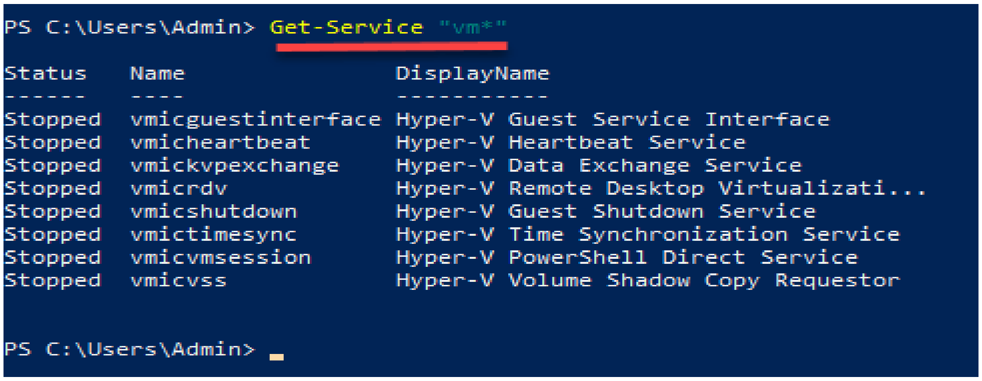


```
PS C:\Users\Admin> Get-Command

CommandType     Name                                               Version    Source
-----------     ----                                               -------    ------
Alias           Add-ProvisionedAppxPackage                         3.0        Dism
Alias           Add-ProvisioningPackage                            3.0        Provisioning
Alias           Add-TrustedProvisioningCertificate                 3.0        Provisioning
Alias           Apply-WindowsUnattend                              3.0        Dism
Alias           Disable-PhysicalDiskIndication                     2.0.0.0    Storage
Alias           Disable-StorageDiagnosticLog                       2.0.0.0    Storage
Alias           Enable-PhysicalDiskIndication                      2.0.0.0    Storage
Alias           Enable-StorageDiagnosticLog                        2.0.0.0    Storage
Alias           Flush-Volume                                       2.0.0.0    Storage
Alias           Get-DiskSNV                                        2.0.0.0    Storage
Alias           Get-PhysicalDiskSNV                                2.0.0.0    Storage
Alias           Get-ProvisionedAppxPackage                         3.0        Dism
Alias           Get-StorageEnclosureSNV                            2.0.0.0    Storage
Alias           Initialize-Volume                                  2.0.0.0    Storage
Alias           Move-SmbClient                                     2.0.0.0    SmbWitness
Alias           Optimize-ProvisionedAppxPackages                   3.0        Dism
Alias           Remove-EtwTraceSession                             1.0.0.0    EventTracingMana
Alias           Remove-ProvisionedAppxPackage                      3.0        Dism
Alias           Remove-ProvisioningPackage                         3.0        Provisioning
Alias           Remove-TrustedProvisioningCertificate              3.0        Provisioning
Alias           Set-EtwTraceSession                                1.0.0.0    EventTracingMana
Alias           Write-FileSystemCache                              2.0.0.0    Storage
```

Get-Service: Finds all cmdlets with the word ‘service’ in it.

Example: Get all services that begin with “vm”

Get-Service "vm*"

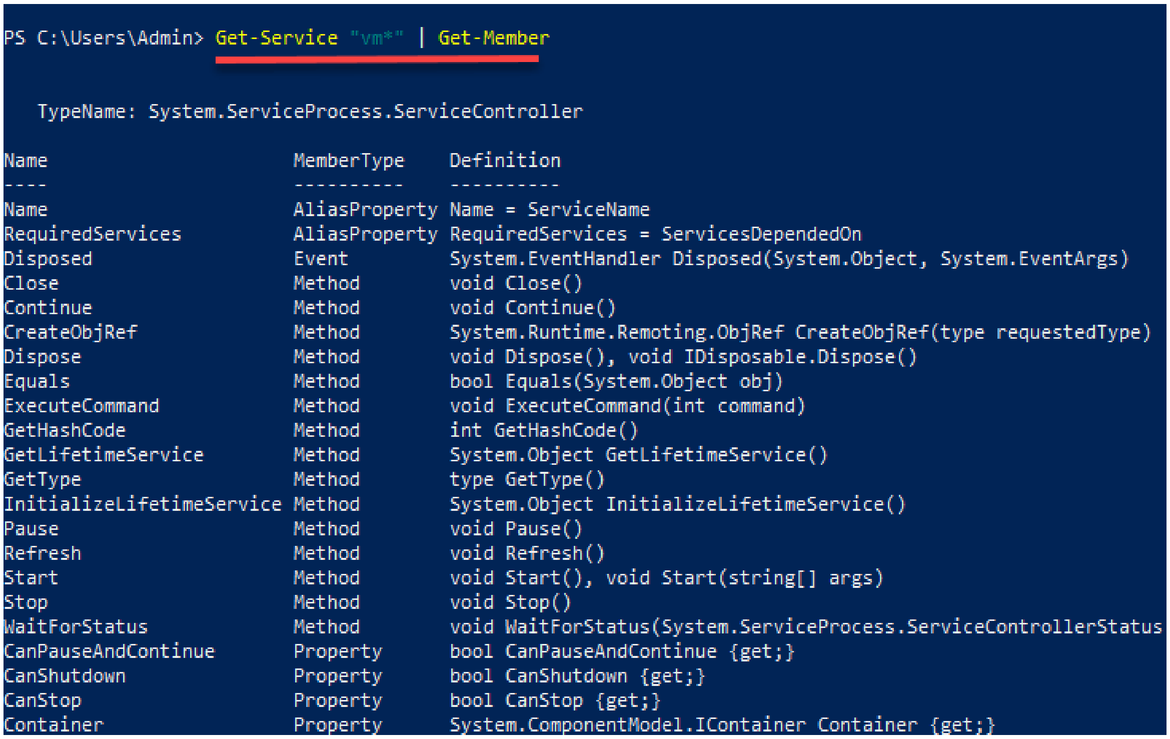


```
PS C:\Users\Admin> Get-Service "vm*"

Status   Name               DisplayName
------   ----               -----------
Stopped  vmicguestinterface Hyper-V Guest Service Interface
Stopped  vmicheartbeat      Hyper-V Heartbeat Service
Stopped  vmickvpexchange    Hyper-V Data Exchange Service
Stopped  vmicrdv            Hyper-V Remote Desktop Virtualizati...
Stopped  vmicshutdown       Hyper-V Guest Shutdown Service
Stopped  vmictimesync       Hyper-V Time Synchronization Service
Stopped  vmicvmsession      Hyper-V PowerShell Direct Service
Stopped  vmicvss            Hyper-V Volume Shadow Copy Requestor


PS C:\Users\Admin> _
```

Get- Member: Show what can be done with an object

Example: Get members of the vm processes.

Get-Service "vm*" | Get-Member



```
PS C:\Users\Admin> Get-Service "vm*" | Get-Member


   TypeName: System.ServiceProcess.ServiceController

Name                      MemberType    Definition
----                      ----------    ----------
Name                      AliasProperty Name = ServiceName
RequiredServices          AliasProperty RequiredServices = ServicesDependedOn
Disposed                  Event         System.EventHandler Disposed(System.Object, System.EventArgs)
Close                     Method        void Close()
Continue                  Method        void Continue()
CreateObjRef              Method        System.Runtime.Remoting.ObjRef CreateObjRef(type requestedType)
Dispose                   Method        void Dispose(), void IDisposable.Dispose()
Equals                    Method        bool Equals(System.Object obj)
ExecuteCommand            Method        void ExecuteCommand(int command)
GetHashCode               Method        int GetHashCode()
GetLifetimeService        Method        System.Object GetLifetimeService()
GetType                   Method        type GetType()
InitializeLifetimeService Method        System.Object InitializeLifetimeService()
Pause                     Method        void Pause()
Refresh                   Method        void Refresh()
Start                     Method        void Start(), void Start(string[] args)
Stop                      Method        void Stop()
WaitForStatus             Method        void WaitForStatus(System.ServiceProcess.ServiceControllerStatus
CanPauseAndContinue       Property      bool CanPauseAndContinue {get;}
CanShutdown               Property      bool CanShutdown {get;}
CanStop                   Property      bool CanStop {get;}
Container                 Property      System.ComponentModel.IContainer Container {get;}
```

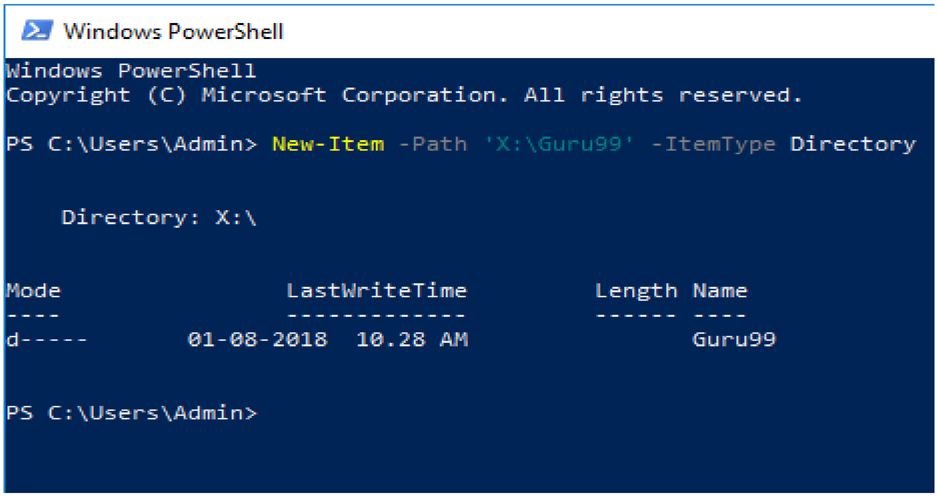
Other Commands:

- Get Module Shows packages of commands
- Get Content This cmdlet can take a file and process its contents and do something with it
- Get- get Finds all cmdlets starting with the word ‘get-

Example: Create a Folder

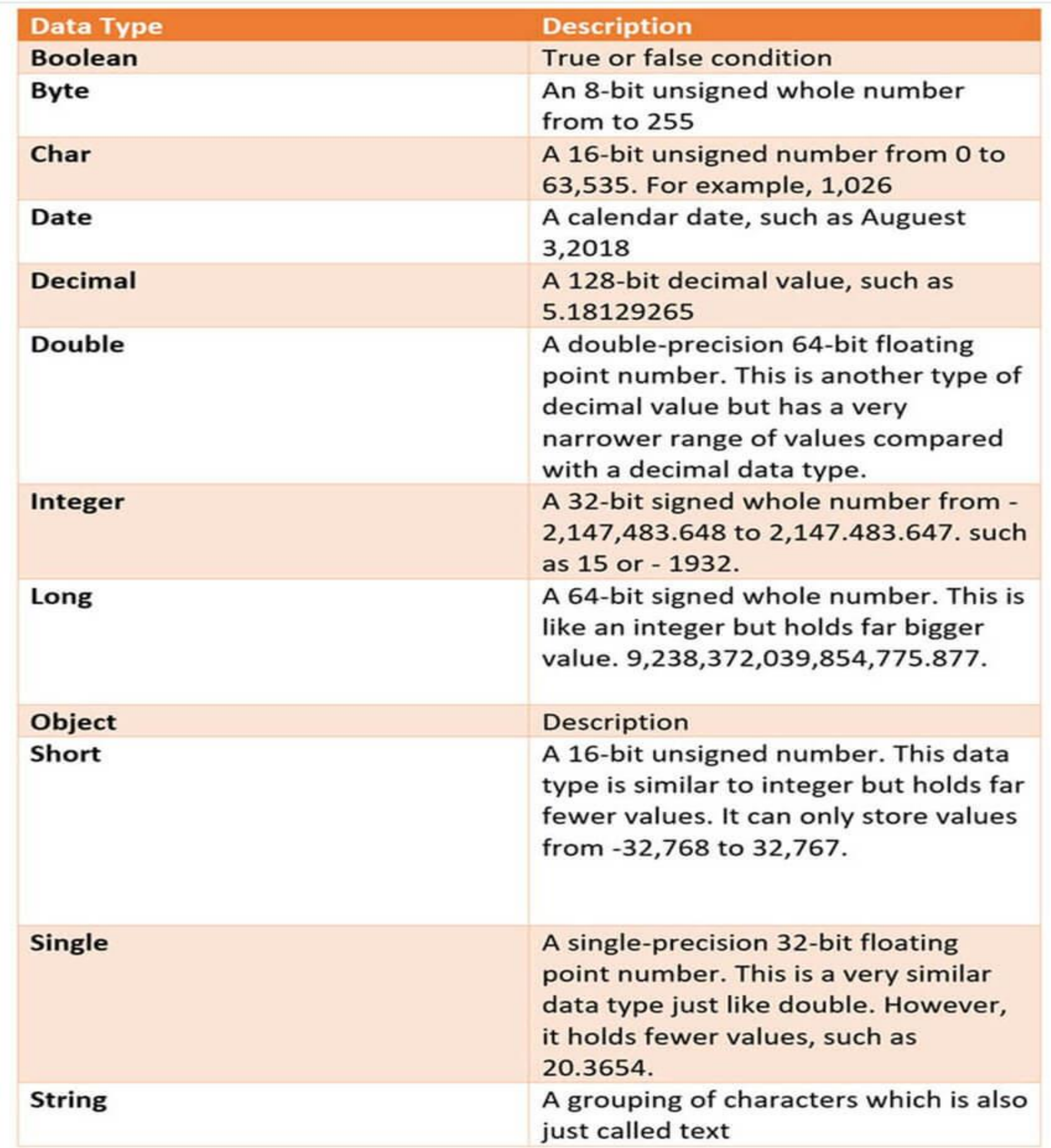
New-Item -Path 'X:\Guru99' -ItemType Directory

Output



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\Admin> New-Item -Path 'X:\Guru99' -ItemType Directory


    Directory: X:\


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----       01-08-2018   10.28 AM                Guru99


PS C:\Users\Admin>
```

Powershell Data types:



Data Type	Description
Boolean	True or false condition
Byte	An 8-bit unsigned whole number from to 255
Char	A 16-bit unsigned number from 0 to 63,535. For example, 1,026
Date	A calendar date, such as Auguest 3,2018
Decimal	A 128-bit decimal value, such as 5.18129265
Double	A double-precision 64-bit floating point number. This is another type of decimal value but has a very narrower range of values compared with a decimal data type.
Integer	A 32-bit signed whole number from - 2,147,483.648 to 2,147.483.647. such as 15 or - 1932.
Long	A 64-bit signed whole number. This is like an integer but holds far bigger value. 9,238,372,039,854,775.877.
Object	Description
Short	A 16-bit unsigned number. This data type is similar to integer but holds far fewer values. It can only store values from -32,768 to 32,767.
Single	A single-precision 32-bit floating point number. This is a very similar data type just like double. However, it holds fewer values, such as 20.3654.
String	A grouping of characters which is also just called text

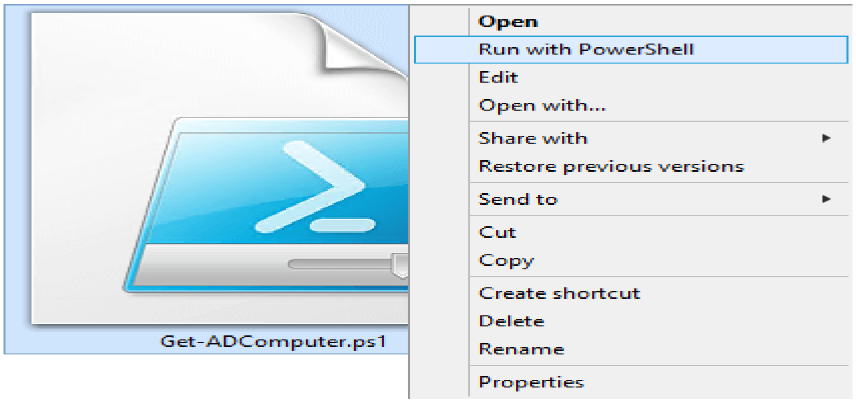
Special Variables

Special Variable	Description
$Error	An array of error objects which display the most recent errors
$Host	Display the name of the current hosting application
$Profile	Stores entire path of a user profile for the default shell
$PID	Stores the process identifier
$PSUICulture	It holds the name of the current UI culture.
$NULL	Contains empty or NULL value.
$False	Contains FALSE value
$True	Contains TRUE value

PowerShell Scripts

Powershell scripts are store in .ps1 file. By default, you can't run a script by just double-clicking a file. This protects your system from accidental harm. To execute a script:

Step 1: right-click it and click "Run with PowerShell."



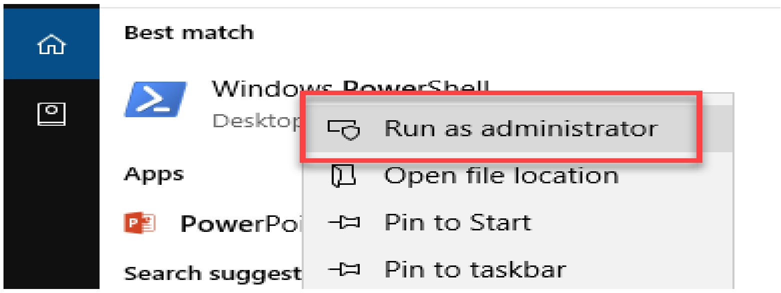
Moreover, there is a policy which restricts script execution. You can see this policy by running the Get-ExecutionPolicy command.

You will get one of the following output:

- **Restricted**— No scripts are allowed. This is the default setting, so it will display first time when you run the command.
- **AllSigned**— You can run scripts signed by a trusted developer. With the help of this setting, a script will ask for confirmation that you want to run it before executing.
- **RemoteSigned**— You can run your or scripts signed by a trusted developer.
- **Unrestricted**— You can run any script which you wants to run

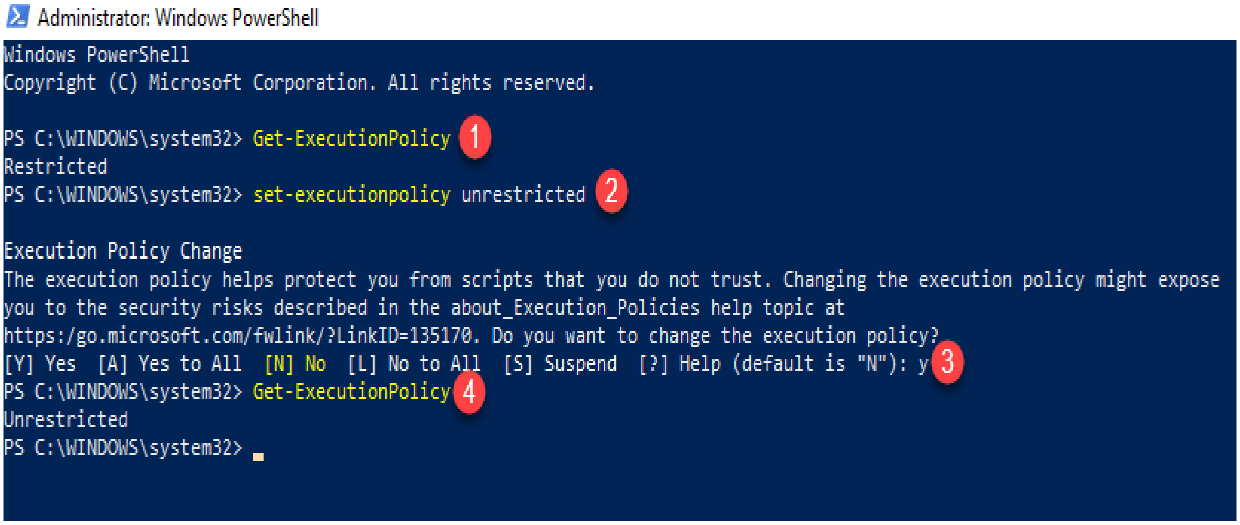
Steps to Change Execution Policy

Step 1) Open an elevated PowerShell prompt. Right Click on PowerShell and “Run as Administrator”



Step 2) Enter the Following commands

1. Get-ExecutionPolicy
2. Set-executionpolicy unrestricted
3. Enter Y in the prompt
4. Get-ExecutionPolicy



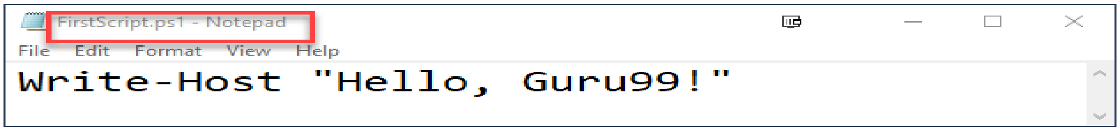
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\WINDOWS\system32> Get-ExecutionPolicy
Restricted
PS C:\WINDOWS\system32> set-executionpolicy unrestricted

Execution Policy Change
The execution policy helps protect you from scripts that you do not trust. Changing the execution policy might expose
you to the security risks described in the about_Execution_Policies help topic at
https:/go.microsoft.com/fwlink/?LinkID=135170. Do you want to change the execution policy?
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "N"): y
PS C:\WINDOWS\system32> Get-ExecutionPolicy
Unrestricted
PS C:\WINDOWS\system32> _
```

First PowerShell Script

In a notepad write the following command

Write-Host "Hello, Guru99!"

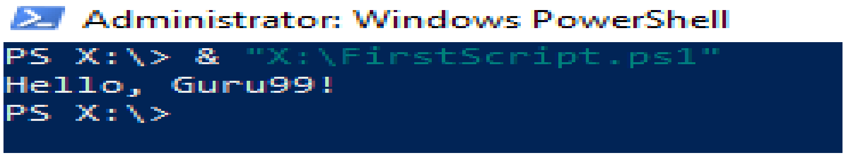
PowerShell Scripts have an extension ps1. Save the file as FirstScript.ps1



```
Write-Host "Hello, Guru99!"
```

In Powershell, call the script using the command

& "X:\FirstScript.ps1"



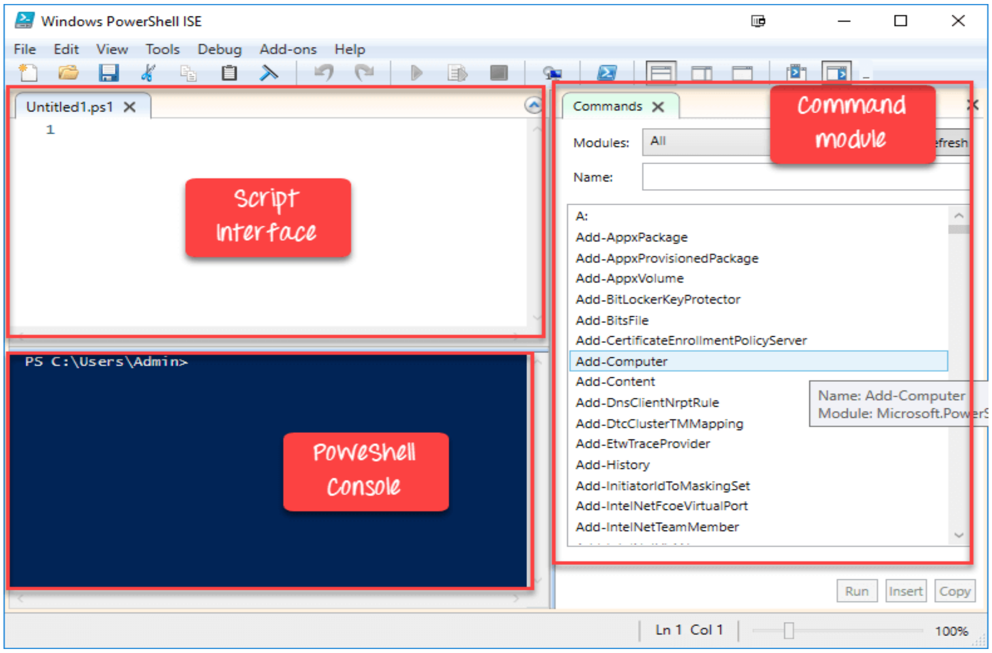
```
PS X:\> & "X:\FirstScript.ps1"
Hello, Guru99!
PS X:\>
```

What is PowerShell ISE?

The Windows PowerShell Integrated Scripting Environment(ISE) is the default editor for Windows PowerShell. In this ISE, you can run commands, writer test, and debug scripts in an in a window base GUI environment. You can do multiline editing, syntax coloring, tab completion, selective execution and lots of other things.

Windows PowerShell ISE also allows you to run commands in a console pane. However, it also supports panes that you can use to simultaneously view the source code of your script and other tools which you can plug into the ISE.

You can even open up multiple script windows at the same time. This is specifically useful when you are debugging a script which uses functions defined in other scripts or modules.



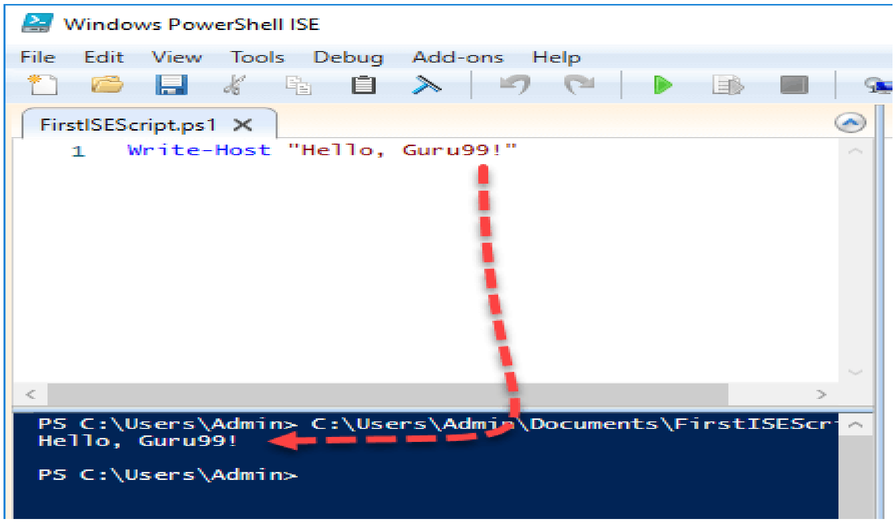
```
PS C:\Users\Admin>
```

A:
Add-AppxPackage
Add-AppxProvisionedPackage
Add-AppxVolume
Add-BitLockerKeyProtector
Add-BitsFile
Add-CertificateEnrollmentPolicyServer
Add-Computer
Add-Content
Add-DnsClientNrptRule
Add-DtcClusterTMMapping
Add-EtwTraceProvider
Add-History
Add-InitiatorIdToMaskingSet
Add-IntelNetFcoeVirtualPort
Add-IntelNetTeamMember

PowerShell ISE

The same script we created in notepad, can be created in ISE

1. Paste code into the editor
2. Save Script
3. Use F5 to run the script
4. Observe output in the console



```
Write-Host "Hello, Guru99!"
```

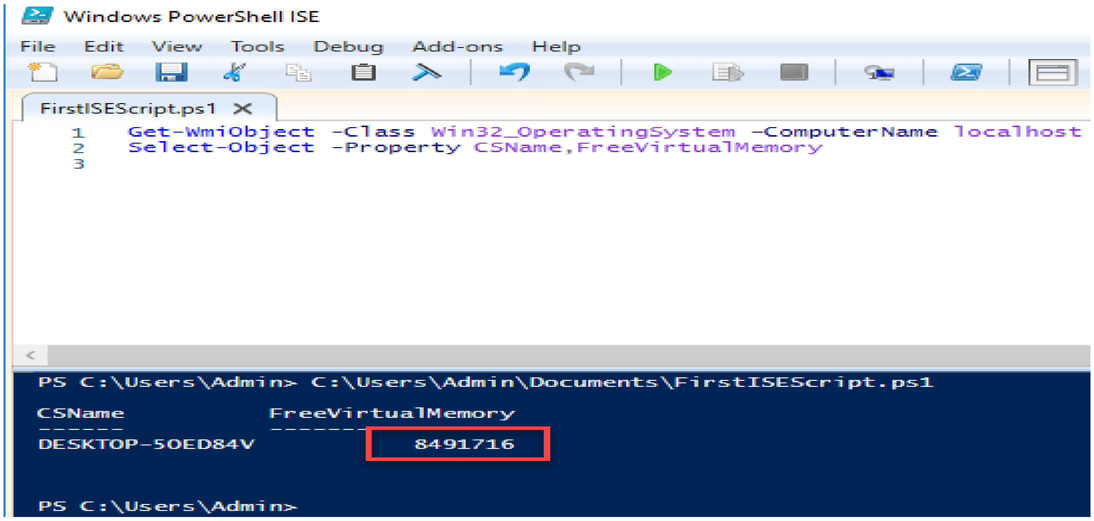
```
PS C:\Users\Admin> C:\Users\Admin\Documents\FirstISEScr
Hello, Guru99!

PS C:\Users\Admin>
```

Sample 2:

The following code will give the Free Virtual Memory in your machine

Get-WmiObject -Class Win32_OperatingSystem –ComputerName localhost |

Select-Object -Property CSName,FreeVirtualMemory



```
Get-WmiObject -Class Win32_OperatingSystem -ComputerName localhost
Select-Object -Property CSName,FreeVirtualMemory
```

```
PS C:\Users\Admin> C:\Users\Admin\Documents\FirstISEScript.ps1

CSName              FreeVirtualMemory
------              -----------------
DESKTOP-50ED84V           8491716


PS C:\Users\Admin>
```

PowerShell Concepts

Now in this PowerShell for beginners tutorial, we will learn about important PowerShell concepts:

Cmdlets	Cmdlet are build-command written in .net languages like VB or C#. It allows developers to extend the set of cmdlets by loading and write PowerShell snap-ins.
Functions	Functions are commands which is written in the PowerShell language. It can be developed without using other IDE like Visual Studio and devs.
Scripts	Scripts are text files on disk with a .ps1 extension
Applications	Applications are existing windows programs.
What if	Tells the cmdlet not to execute, but to tell you what would happen if the cmdlet were to run.
Confirm	Instruct the cmdlet to prompt before executing the command.
Verbose	Gives a higher level of detail.
Debug	Instructs the cmdlet to provide debugging information.
ErrorAction	Instructs the cmdlet to perform a specific action when an error occurs. Allowed actions continue, stop, silently-continue and inquire.
ErrorVariable	It specifies the variable which holds error information.
OutVariable	Tells the cmdlet to use a specific variable to hold the output information
OutBuffer	Instructs the cmdlet to hold the specific number of objects before calling the next cmdlet in the pipeline.

Advantages of using PowerShell script

- PowerShell scripts are really powerful and could do much stuff in fewer lines.
- Variables are declared in the form $<variable>
- Variables could be used to hold the output of command, objects, and values.
- “Type” of a variable need not be specified.

PowerShell Vs. Command Prompt

PowerShell	Command Prompt
PowerShell deeply integrates with the Windows OS. It offers an interactive command line interface and scripting language.	Command Prompt is a default command line interface which provided by Microsoft. It is a simple win32 application that can interact and talk with any win32 objects in the Windows operating system.
PowerShell uses what are known as cmdlets. It can be invoked either in the runtime environment or the automation scripts.	No such features offer by command prompt.
PowerShell considers them as objects. So the output can be passed as an input to other cmdlets through the pipeline.	Command Prompt or even the *nix shell, the output generated from a cmdlet is not just a stream of text but a collection of objects.
The PowerShell is very advanced regarding features, capabilities and inner functioning.	Command prompt is very basic.

Applications of Powershell

Today, PowerShell has become an ideal choice for IT administrators as it eases management operation and effort in large corporate networks. For example, let’s assume that you are managing a large network which contains more than four hundred servers. Now you want to implement a new security solution. This security solution depends on a certain service which needs to run on those servers.

You can surely log in to each server and see if they have that service install and running or not. However, it certainly takes a lot of human errors as your staff needs to spend lots of time on this non-productive process.

However, if you use PowerShell, then you could complete this task in just a few minutes. That’s because the entire operation is done with a single script which gathers information about the services running on the servers