Security · Networking · VPN Clients

Safepipe Centre > Self-test courses > The Domain Name System (DNS)

Documentation
- Printed guides
- HowTo
- Q & A
- Interoperability

Reference
- Encyclopedia
- Other resources

Training
- Self-test courses

Download
- Software
- Service Tool

Support
- Contact

The Domain Name System (DNS)

The Domain Name System (DNS) – an overview

The following is a general, non-technical introduction to the Domain Name System and how it works on the Internet. If you are looking for specific information on how the domain name space is organised, how domains can be acquired, or how DNS servers do what they do, you might want to go directly to **'More about the Domain Name System (DNS)'.**

What is DNS?

Having a domain, e.g. 'mycompany.com', is an important step in establishing an identity for a business on the Internet. People enter the domain as part of an e-mail address or a Web address. Really, what the network uses to route traffic is not domain names as such, but the corresponding IP addresses. The translation between fully qualified domain names and IP addresses is taken care of by DNS servers. Thus, one of the advantages of DNS is that it saves us from having to memorise long IP addresses because we can use intuitive domain names instead.

The abbreviation DNS is used to describe two related things: the Domain Name System and the Domain Name Service. The Domain name system is the distributed database responsible for the domain name-to-IP address conversion, while the Domain Name Service, as the name implies, is the service offered by this system.

DNS affects almost every other Internet service, from e-mail, surfing on the Web, to audio conferencing. For instance, when someone enters a domain name (e.g. 'www.company.com') into the address field of a browser and sends off the request, they are making use of DNS. Furthermore, DNS servers (also known as name servers) hold information on what mail server that e-mail for a given domain should be delivered to, enabling us to use e-mail addresses in the format 'username@domain', which doesn't specify a particular mail server.

to the top

How does DNS work?

DNS is a distributed database. DNS service is offered by thousands of DNS servers on the Internet, each responsible for a portion of the name space called a *zone.* The servers that have access to the DNS information (zone file) for a zone is said to have authority for that zone. When queried by for instance a Web server, the DNS server translates the domain name into the corresponding IP address. For example, the domain name 'www.example.com' might translate to '195.24.22.209'.

When TCP/IP software is installed on a Windows workstation, the IP address of one or more name server(s) is one of the configured parameters. This is the name server that the host (or really, the browser application on the host) should direct its query to when looking for the IP address of for instance a Web server on the Internet (given that this server has a fully qualified domain name). It is also the server responsible for telling other servers on the Internet how to get in touch with the workstation, if this should be desired (again given that the workstation has a fully qualified domain name). A fully qualified domain name, like 'www.example.com' consists of a hostname ('www') as well as a domain ('example.com').

No single one of the thousands of name servers on the Internet knows all the keys for translating domain names into IP addresses and vice versa, but each server knows the names and IP address of every user's computer on its branch of the Internet (zone). The server then exchanges this information with other domain name servers from other corners of the net, thus enabling domain name addressed communication between hosts on different networks.

The Internet would work without DNS, of course, but it would mean that all traffic would have to be addressed using IP addresses.

to the top

More about the Domain Name System (DNS)

The following provides a closer look at the specifics of DNS, including a description of how the domain name space is organised, how domains can be acquired, and how DNS servers do what they do.

Pros and cons of maintaining your own DNS server

There are two basic ways to configure DNS. One option is to use the DNS server of an Internet service provider. The other is to set up a DNS server on your local network.

Having and maintaining your own primary DNS server leaves the job of configuring and updating the server on your shoulders. On the other hand, it also gives you a number of benefits:

- Firstly, a local DNS server can give your company a ***measure of security***. If you are running IP network-based applications inside your network that require users to connect to internal machines by name, it is not a great idea to advertise the names and addresses of these machines. DNS can give hackers a map of your network, so setting up an internal DNS server that does not publish information to the Internet can be a good idea.

- Secondly, a local DNS server lets you be the ***master of your own domain***. If your Web site often moves around or changes, managing your own primary DNS server allows you to make changes, additions, and delitions at your own pace without involving your Internet service provider. For instance you can add hosts and create subdomains within your domain, e.g. 'subdomain.company.com' – maybe to advertise your company's new exciting product by giving the product its own Web site on the Internet.

- Another important issue is ***reverse name mapping***; some Internet service providers keep reverse name information only for servers and not for the individual host systems. In this case, users may not be able to connect to FTP or other information servers that attempt to reconcile the user's hostname with the IP address before granting access.

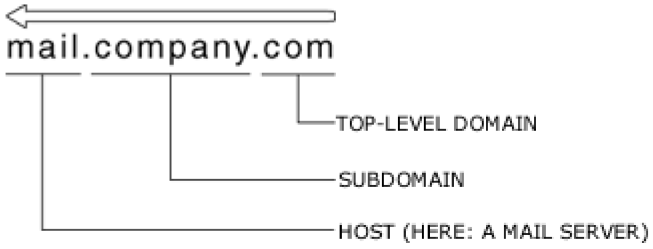
Having a DNS server on your local network, however, will mean increased traffic to and from your network, which you may be billed for. Another issue to consider is whether your DNS server is to have authority for hosts on other networks than your local area network, e.g. for the Web server of a remote branch of your company. If this is the case, access from the Internet to the Web server on the remote network may be cut off if for some reason your DNS server could not be reached. (As for Web servers etc. on your own network, they would be unreachable regardless of where your DNS server was situated if connectivity to your network was lost.)

to the top

The domain hierarchy

The domain name space is organised in a hierarchy comprising a ***top-level domain***, and/or a ***subdomain***, and/or a ***hostname***. The Internet naming hierarchy is best understood if a fully qualified domain name, like 'mail.company.com' is read from right to left. In 'mail.company.com', 'com' is the top-level domain,'company' is a subdomain under this top-level domain, while 'mail' indicates the particular host – here a mail server. Generic top-level domains include 'com', 'org' and 'net'. Other top-level domains use international two-letter country codes, such as 'ca' (Canada), 'de' (Germany), 'es' (Spain), and 'fr' (France).

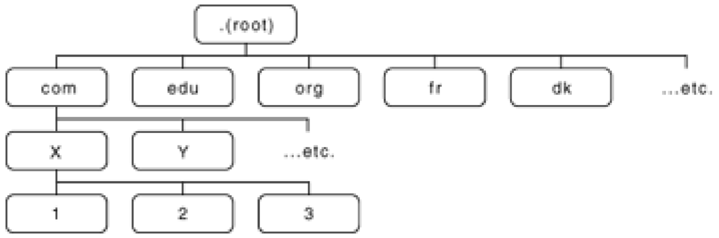
Strictly speaking, fully qualified domain names should be written with a dot at the end (the root zone), e.g. 'mail.company.com.", however, in this course we imply the last dot.



A fully qualified domain name consists of different parts

Top-level domains and IP network addresses are assigned and maintained by The Internet Corporation for Assigned Names and Numbers (ICANN), which is responsible for the overall co-ordination and management of the DNS. A second and third level of domain administrators are responsible for the hostname and address assignment within the subdomains.

The domain hierarchy can be represented by an inverted tree (also called the domain name space). At the very top of the hierarchy one finds a small number of root name servers, which contain pointers to master name servers for each of the top-level domains. There are currently thirteen such root servers on the Internet. These servers all contain identical information - there are ten servers purely for backup reasons.



The DNS tree: At the very top of the hierarchy one finds a small number of root name servers. The next level, ‘com’, ‘edu’, etc., consists of top-level domains, while x and y indicate subdomains (for instance ‘ CompanyX’ and ‘CompanyY’). The hosts,’Computer 1’,’ Computer 2’, etc., reside at the lowest level in the hierarchy

In the DNS tree, everything under a particular point in the tree (e.g. ‘com´) falls into that particular ***domain.*** In the illustration above, both ‘computer 1´, ‘company X´ as well as ‘company Y´ thus fall within the ‘com’ domain. The fully qualified domain name of any host in the tree is the path from that host to the root (‘up’ the tree) with dots separating the names in the path. Thus the fully qualified domain name of the host ‘computer 1’ is ‘computer1.companyX.com’. All devices in the same same domain share a part of their IP address.

A DNS server’s ***zone*** is all the domain names that the server has been given authority over. This means that the server has a list of all the domain names, plus information about how to get to each of them. A zone is a pruned domain and thus might not include all the subdomains and hosts within a given domain. The pruning occurs when zones are delegated to individual servers. The zones thus relate to the way the DNS database is partitioned and distributed.

If a company uses the mail or Web server of an Internet service provider, it can use the Internet service providers’ domain (e.g. ‘isp.net’) as part of the company’s Web address or e-mail address. However, some providers might allow companies to use their own domain (e.g. ‘mycompany.com’), if the company has one registered. If a company has its own mail or Web server and wish to use it for their own domain (e.g. 'mycompany.com'), the domain must be registered with an official domain name service registry in order to be reachable for others on the Internet.

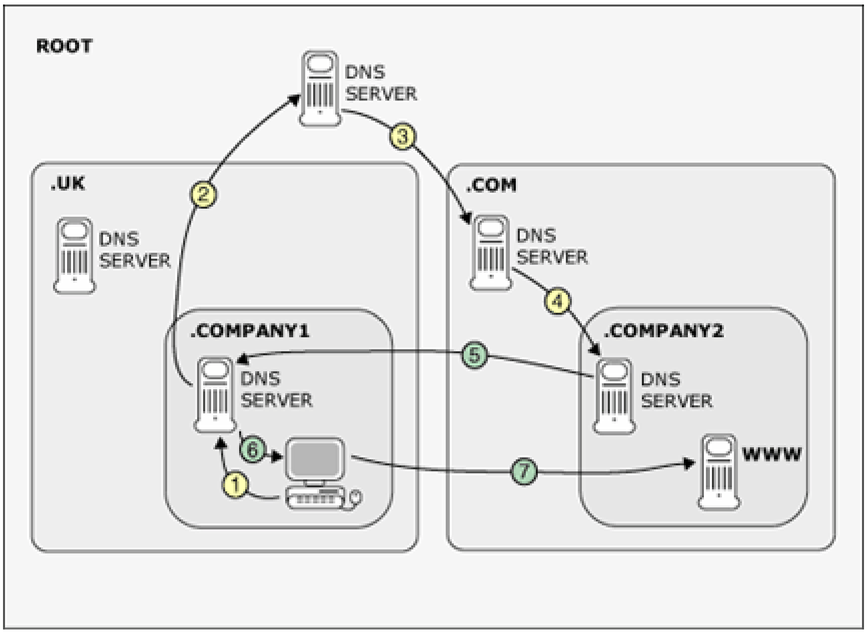
to the top

Querying a DNS server

When you type in a URL – for instance `http://www.company2.com´– in the address field of a browser program, a query is sent to the DNS server (indicated in the hosts TCP/IP configuration) asking for the corresponding IP address. The DNS server may not contain the information about the particular destination host. In that case, it will attempt to solve the problem by forwarding the query to another name server that is higher up in the domain name hierarchy. If that query is not successful, the second server will ask yet another – until it finds one that knows.

The path of enquiry follows the domain name hierarchy.

Imagine for instance, that you are working in ‘company1’ and have decided to take a look at the Web site of your closest competitor ‘company2’. The domain name for this Web site is ‘www.company2.com’. The following illustrates the steps involved in converting this domain name into the corresponding IP address:



Example of a DNS enquiry path

1. The query is initiated when you type the domain name of the computer that you wish to contact, ‘www.company2.com’, into the address/location field of your browser and hit enter. Upon this command, your workstation contacts its primary DNS server to see if it knows the IP address of `www.company2.com´.

2. The requested domain name does not fall within the zone of company1’s DNS server, and consequently, the server does not hold the required information. Therefore, the server turns to a root server for help. (Root servers are servers, which maintain information about all the top-level domains). Every DNS server on the Internet must know the IP addresses of the about 10 root domain name servers.

3. The root server holds information on the top-level domains, but usually not on subdomains. The root server therefore passes the query along to the server which is responsible for the’.com’ domain. The authoritative server for a top-level domain like ‘.com’ contains information on which name-server is responsible for each subdomain in its zone, thus also for ‘company2.com’.

4. The ‘.com’ server, in turn, refers to the authoritative server for the ‘company2.com’ domain. The DNS server for a subdomain, such as ‘company2.com’, contains detailed addressing information about the hosts in its zone, including the Web server (www).

5. Upon receiving the query, the DNS server responsible for the ‘company2.com’ domain looks at its table of hostnames and corresponding IP addresses and, via the previously queried servers, supplies your primary DNS server with the IP address of the server called 'www.company.com’.

6. Once the information is located and returned to your DNS server, your server passes the information back to your workstation.

7. You are on your way! Your workstation is now able to contact ‘www.company2.com’ using the corresponding IP address supplied by the DNS server.

Usually this entire process occurs quickly (within seconds).

However, if the server has recently answered a query the same hostname (within a time period set by the administrator of the authoritative DNS server's zone to prevent passing old information), it will not have to go through this whole process again, as it can quickly locate the information in its cache and reply directly.

In the example above, the DNS servers behave as 'recursive'. A ***recursive*** DNS server takes on the burden of querying other name servers to come up with a more fulfilling answers to queries than its own data provides. A ***non-recursive*** DNS server, on the other hand, simply looks in its local data and returns the best answer it has to a given query. In other words, while the recursive server asks the next server for more information, the non-recursive server only goes as far as to suggest to the server who queried it, that it go query someone else!

The DNS is also used the other way around – for instance someone running a Web site might like to log the names of the computers which have visited the site. The server programs do this by doing a reverse query ***(reverse name mapping)***, that is, asking the DNS system for the corresponding domain name for a given IP address.

The DNS records can be examined with a number of common TCP/IP tools. The most common DNS lookup utilities are ‘NSLOOKUP’ and ‘Host’.

DNS, which was designed in 1984, is one of the key elements that has allowed the Internet to grow as it has. The old system consisted of a single file, known as the host table, maintained by the Stanford Research Institute's Network Information Center (SRI-NIC). If this system had continued, the static file would not only be absolutely gigantic, but would also be constantly out of date. With DNS, when any site needs to add or remove computers, they simply update their portion of the database and, after a short period, everyone on the Internet can see the change!

to the top

Acquiring a domain name

Domain names have to be registered with an ***official domain name service registry***. There are two ways of doing this. You either pay an Internet service provider (ISP) to do it for you, or you contact a domain name registry and complete the application process yourself.

Top-level domains (like ‘.com’) and IP network addresses are assigned and maintained by The Internet Corporation for Assigned Names and Numbers (ICANN), which is responsible for the overall co-ordination and management of the DNS. The ICANN in turn allocates blocks of IP address space to Regional Internet Registries (RIRs): InterNIC in North America, RIPE in Europe, and APNIC in Asia. These regional domain registries again allocate blocks of IP address space to Local Internet Registries (LIRs), which in turn assign the addresses to end users (with or without the aid of an Internet service provider).

When a domain name is registered, it is added to the DNS of the top-level domain. If the domain ‘example.com’ was registered, it would be entered in the DNS of the ‘.com’ domain.

When registering a domain, the registrant has to supply a (number of) contact person(s) which is responsible for the administration, technical maintenance of the DNS server and financing of the domain. One person or company can be responsible for more than one area.

The three contacts which need to be supplied are:

1. **The administrative contact.** This is the owner of the domain, that is, the person to contact is someone wants to reach whoever is in authority of the domain. The technical term for this contact is ADMIN-C.

2. **The technical contact.** This is the person or organisation responsible for the host mastering of the domain. This is whom should be contacted if the domain has been set-up wrong, or if changes or additions have been made to the domain. The technical term for this contact is TECH-C.

3. **The billing contact.** The person who pays the bills.

In addition to the contact information, you also need to supply information about on which name servers the domain is going to reside. These name servers are called the authoritative name servers, because they have authority over the domain, which means that these are the servers that all other name servers and hosts will ask for information about the domain – such as the IP addresses of hosts and where to deliver mail.

The authoritative name servers must be updated when a change to the domain is made. The technical term for this is ‘reloading a zone’.

If you want ***reverse name mapping*** (finding the domain name for a given IP address) to be possible for your domain, you must also register a so-called 'in-addr.arpa' domain. This is a special domain which has been added at the top-level of the domain tree to make reverse DNS mapping possible. It is special in the sense that there can be exactly 256 subdomains at each level, and that the label names can only be the numbers 0-255. An example is 1.12.123.195.in-addr.arpa. The registered domains under in-addr.arpa corresponds to reversed IP addresses (the IP address in the label name above would normally be written 195.123.12.1). The IP addresses are reversed because the label to the left is the most specific in a domain name, while the most specific label in an IP address is the octet to the right. It is of course only possible to register an 'IP address domain' if the IP address in question is assigned to you.

Primary and secondary DNS servers

There should always be one and only one DNS server which has direct access to the DNS information in the zone file for a particular domain. This DNS server is called the ***primary domain name server*** for the domain. The ***secondary domain name server*** is a DNS server which downloads a copy of the primary domain name server's zone file periodically. Secondary domain name servers do this by querying the primary domain name server (usually every 6 hours or so, but the domain administrator can set this check as often as he or she likes) to see if the primary's information has changed. If it has, the secondary simply downloads the entire table again from the primary.

A zone can have as many secondary domain name servers as the DNS administrator likes. To make them useful, the administrator has to make sure that the parent level domain DNS administrator knows about them, or else the secondary servers will never get queried even if the primary server cannot be contacted.

Some administrators choose to use the primary server as secondary server as well by entering the primary server in both fields in the domain registration form. It saves the administrator from having to provide two servers, but it has the disadvantage that the whole network is unreachable from the Internet if the server cannot be contacted. However, in scenarios where the primary name server and for instance a Web server is hosted on the same computer, this is less of a problem. In this case, the physical set up means that if the computer cannot be contacted and connectivity between the DNS server and the Internet is thus lost, having a secondary name server on a different network will still not enable anyone to access the information on the Web server, as the Web server cannot be reached if the computer cannot be reached!

to the top

Resource records

The information that a DNS server needs to answer queries from hosts on the Internet is kept in a number of *resource records* (RRs). The information in the resource records is entered and updated by the server administrator.

The two most common resource records are:

- **A: The Address Record.** This record supplies the IP address for a given hostname. The hostnames are assigned by the DNS server administrator. You will need A records for any public servers you maintain (servers which should be accessible from the Internet). The most common hostnames are ‘www’ and ‘mail’ that are used to identify Web servers and mail servers. You may also want to set up A records for each of your workstations if your users use FTP (File Transfer Protocol) to download software from the Internet. This is because some FTP sites perform a lookup to get the domain name of the machine from which they receive download requests. If the machine has no name, the site rejects the request. Since hosts can have multiple IP addresses, corresponding to multiple physical network interfaces, it is possible for multiple A records to match a given domain name. Similarly, one IP address may also have several corresponding hostnames. This may be configured using CNAME records (se below).

- **MX: The Mail Exchange Record.** This record indicates which host (s) handles electronic mail for the domain, and offers a method of prioritising the order of mail servers that e-mails to the domain should be attempted delivered to. An MX record has two parts: the name of the machine that will accept mail for the domain, and a preference value. A domain can have multiple MX records such as the following: mail1.company.com 0; mail2.company.com 10, and mail3.company.isp.net 100. In this case, mail delivery will be attempted to mail1.company.com first because it has the lowest preference value (0). If delivery fails (for instance because this server is down or deems the e-mail address unknown), mail2.company.com will be tried next as specified in the MX record, and so on. Each of the hosts mentioned in your MX records needs an A record to associate them with an IP address.

The Mail exchange record is what makes it possible to have e-mail addresses in the format ‘user@domain.com’ that use the domain without specifying the specific host (the mail server). If no MX record was created for a domain, the specific domain of every mail server within the domain would have to be specified though an entry in the address record (A), and the e-mail address for the user would look something like user@mail1.domain.com.

Other DNS resource records include:

- **SOA: The Start of Authority Record**, which indicates the primary name server (origin), the responsible DNS administrator, the rules that govern the secondary name servers' queries to the primary name server for zone file updates, as well as a default TTL. TTL (Time To Live) is the length of time that non-authoritative name servers are allowed to keep the resource records in their short term memory (cache), before they have to be discarded.
- **CNAME: The Canonical Name Record**, which supplies host alias names. It is possible to define multiple A records for a given address, thus providing alias, or alternate, hostnames. It is usually easier to supply one A record for a given address and use CNAME records to define alias hostnames for that address.
- **PTR: The Pointer Record**, which associates a hostname with a given IP address. These records are used for reverse name lookups. Reverse lookups can be used to limit access to for instance a server on the net to those from a specific domain or with a specific domain name. An example: you have a Web server on the Internet that you only want certain users to be allowed to connect to. Through a reverse look up when someone intends to connect to the server, it can be established if the visitor's domain is on the list of allowed visitors and, thus, if access should be granted. Reverse look up is also useful for companies wishing to monitor what kind of Internet users visit their Web site.
- **NS: The Name Server Record**, which defines the name server(s) for a given domain.

to the top

Test your knowledge

1.

What does DNS stand for?

● Domain Name System and Domain Name Service

● Data Name System and Data Name Server

● Domain Number Server and Domain Number Service

2.

What does a DNS server do?

● It translates data between incompatible systems

● It translates IP addresses to fully qualified domain names and vice versa

● It translates IP addresses into MAC addresses

3.

What is a zone?

● All the devices on a local area network

● A portion of the name space that a DNS server has authority over

● A portion of the name space that no DNS server has authority over

4.

What does it mean that a

That the server has access to the information

DNS server has authority over a zone?

○ in the zone file (including IP addresses and fully qualified domain names for the devices in the zone)

○ That the DNS server administrator must give his or her permission when a website in the zone is visited

○ That the zone has not yet been set up

5.

Which of the following is a fully qualified domain name?

○ 192.168.0.1

○ lasat.dk

○ www.lasat.dk.

6.

Indicate the domain in www.lasat.dk

○ www.lasat

○ www

○ lasat.dk

7.

Indicate the top-level domain in www.lasat.dk

○ www.lasat.dk

○ .lasat

○ .dk

8.

What is a root server?

○ A back-up server you must always have on your local area network

○ A server that contains pointers to the authoritative name servers for all top-level domains

○ A server that contains direct pointers to all DNS servers on the Internet

9.

What is a primary DNS server?

○ The DNS server which has direct access to the zone file with DNS information for a domain

○ The DNS server which downloads a copy of the zone file with DNS information periodically

○ Another name for a root server

10.

What is an MX record?

○ The record that specifies where undeliverable mail should be sent to

○ The record that specifies the modem records for a domain

○ The record that specifies which mail servers handle mail for a domain

Evaluate Reset