System Administration (LTAT.06.003)

Lab 4
Domain Name Server
A domain name is an identification string that defines a realm of administrative autonomy, authority or control within the Internet.

In general, a domain name identifies a network domain, or it represents an Internet Protocol (IP) resource, such as a personal computer used to access the Internet, a server computer hosting a website, or the web site itself or any other service communicated via the Internet. - Wikipedia
Domain Name

 Shortly, a domain name identifies a service, host or other kinds of networked resources.

 In our case, when talking about domain names, we talk about your VM’s identity (hostname), and IP <-> Domain Name mappings.

 IP and Domain Name are mapped so, that every person accessing Google, does not need to remember many IP addresses.
DNS

The Domain Name System (DNS) is the phonebook of the Internet. Humans access information online through domain names, like nytimes.com or espn.com. Web browsers interact through Internet Protocol (IP) addresses. DNS translates domain names to IP addresses so browsers can load Internet resources.

DNS would be easy, if it was just a lookup table of IP’s and names, but when you buy a domain name, you want to control where it’s pointed.

This means this lookup table needs to be decentralized, but still trustworthy, available and reliable.
The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```

You can use “host”, “dig” and “nslookup” commands in your machines to query the DNS servers.

Each device has a list of DNS servers it trusts. Usually it’s provided by your ISP. In Linux, the file showing trusted DNS servers is “/etc/resolv.conf”.

---

**DNS structure**

The DNS servers keep and sync what we call “DNS records”. A DNS record is a mapping, for an example of a hostname and IP address. The syntax looks like this:

```plaintext
sa.cs.ut.ee. IN A 172.17.66.134
```
DNS structure

Because every little thing on the internet requires DNS, while domain owners still need to be able to edit the records, the DNS is complicated.

The Domain Name System has four server types:

- Root nameserver
- TLD nameserver
- Authoritative nameserver
- DNS Recursive resolver
A recursive resolver (also known as a DNS recursor) is the first stop in a DNS query. The recursive resolver acts as a middleman between a client and a DNS nameserver. After receiving a DNS query from a web client, a recursive resolver will either respond with cached data, or send a request to a root nameserver, followed by another request to a TLD nameserver, and then one last request to an authoritative nameserver.
Root nameservers are the first stop of every DNS request. Their job is to direct a request based on the TLD part of the request (.com, .net, .org, etc.).

There are 13 root nameservers in the world, and these are known to every recursive resolver.

The root nameservers are overseen by a nonprofit called the Internet Corporation for Assigned Names and Numbers (ICANN).
A TLD nameserver maintains information for all the domain names that share a common domain extension, such as .com, .net, or whatever comes after the last dot in a url.

If a user was searching for google.com, after receiving a response from a root nameserver, the recursive resolver would then send a query to a .com TLD nameserver, which would respond by pointing to the authoritative nameserver (see below) for that domain.
Authoritative nameserver

When a recursive resolver receives a response from a TLD nameserver, that response will direct the resolver to an authoritative nameserver.

The authoritative nameserver contains information specific to the domain name it serves (e.g. google.com) and it can provide a recursive resolver with the IP address of that server found in the DNS A record.

In case of a CNAME record, the whole process starts from the beginning again.
DNS records

DNS records (aka zone files) are instructions that live in authoritative DNS servers and provide information about a domain including what IP address is associated with that domain and how to handle requests for that domain.

These records consist of a series of text files written in what is known as DNS syntax. DNS syntax is just a string of characters used as commands that tell the DNS server what to do. All DNS records also have a ‘TTL’, which stands for time-to-live, and indicates how often a DNS server will refresh that record.
DNS record types

- A record - The record that holds the IP address of a domain.
- AAAA record - The record that contains the IPv6 address for a domain.
- CNAME record - Forwards one domain or subdomain to another domain, does NOT provide an IP address.
- MX record - Directs mail to an email server.
- TXT record - Lets an admin store text notes in the record. These records are often used for email security.
- NS record - Stores the name server for a DNS entry.
- PTR record - Provides a domain name in reverse-lookups.
- SOA record - Stores admin information about a domain.
- SRV record - Specifies a port for specific services.
DNS as a tree

Diagram showing the structure of the DNS hierarchy with examples of domain names at various levels.
DNS lab this week

- Try to wrap your head around DNS
- Configure recursive DNS locally
- Configure an authoritative DNS locally
  - Personal zone
  - Reverse mapping
- Declare the whole lab in Ansible and never worry about it again :)

DNS lab this week