Network Protocols – DNS & NTP

INFO-6078 – Managing Enterprise Networks

FANSHAWE

Domain Name System (DNS)

- When interacting with services, humans generally prefer to give names to identify resources
- Computers generally identify these same services numerically using IP addresses
- The Domain Name System (DNS) is used to translate memorable host names into IP addresses
- Before the creation of DNS, these host to IP mappings were stored in the host table, which was manually created/distributed on each device
- DNS is a complex and distributed system to ensure that there is no single point of failure in internet name resolution

Domain Name System (DNS)

DNS Name Space

- The structure of the DNS name space resembles a tree, with each branch representing a domain
- Each leaf represents one or more resource records, which contain information about host names, IP addresses and other domain related settings
- A domain may contain only one zone, but may also be divided into sub-zones called sub-domains

Domain Name System (DNS)

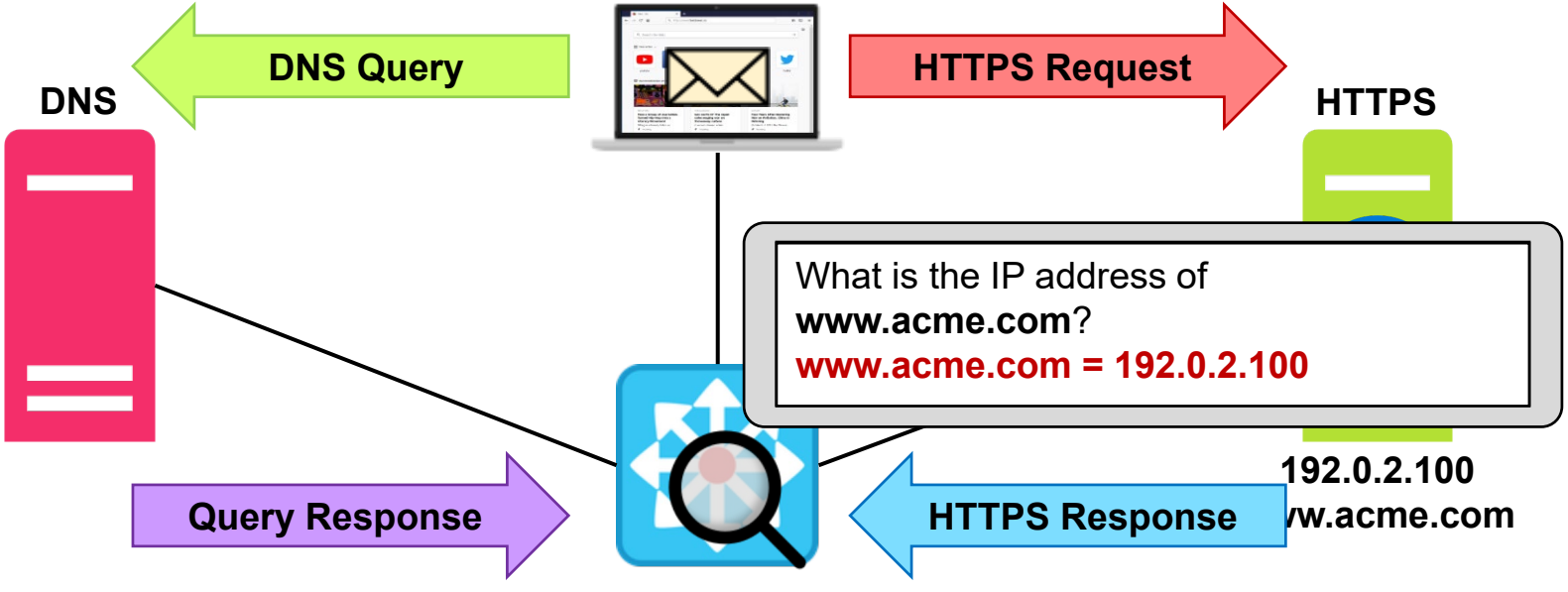
DNS Name Servers

- DNS is a distributed database system that shares administrative responsibility by dividing zones into sub-zones
- These zones are delegated to a DNS name server that is considered to be authoritative over the records in the zone
- An authoritative server is supplies definitive answers to resolution requests and indicates this to resolvers by setting the Authoritative Answer (AA) flag in responses

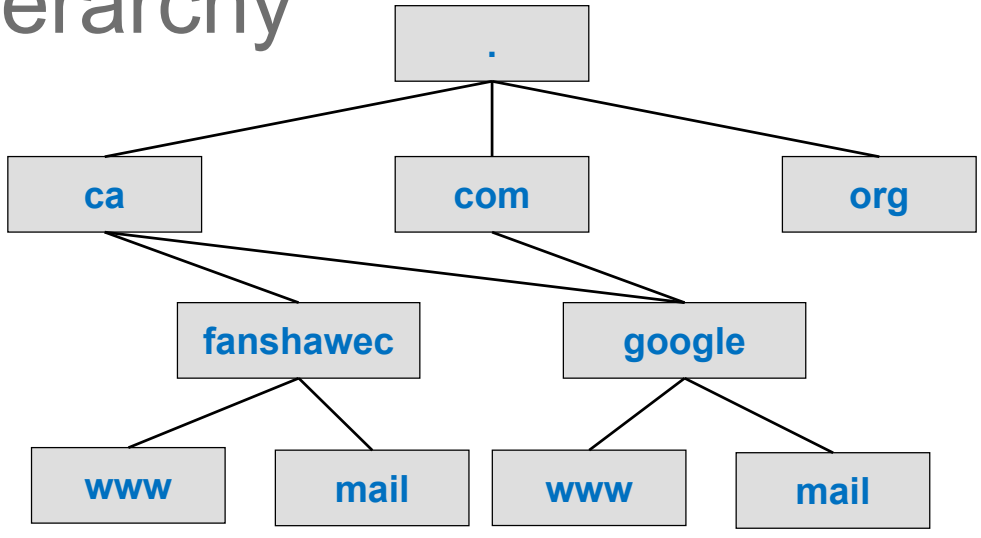
DNS Operation

Resolver

- A resolver is a client that requires name resolution and sends DNS queries to a DNS server



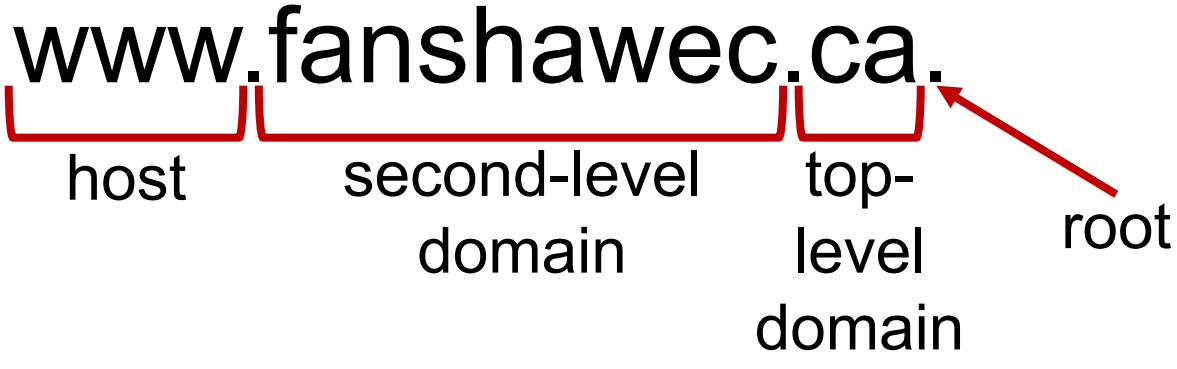
Domain Hierarchy



- The DNS namespace, is a hierarchal collection of domains, similar to an inverse tree
- The tree consists of a root, followed by top-level domains, then second-level domains, normally followed by a host
- This address format is usually called a fully qualified domain name (FQDN)

Domain Hierarchy

- The domain hierarchy makes it possible for a DNS server to locate the authoritative server for any active domain in the system
- Each server in the system contains records that identifies hosts in the next lower level
- To understand the process, we must look at the structure of a FQDN



Domain Hierarchy

Root Name Servers (Hints)

- The root name servers exist at the top of the DNS hierarchy
- Root name servers contain only information about top-level domains
- DNS servers are pre-configured with the IP addresses of the root name servers
- Administrators can change or update the root servers by configuring an alternative address

Domain Hierarchy

Top-Level Domains

- Top-Level Domains (TLDs) are the next level in the hierarchy
- Examples of top level domains include the original TLDs .com, .net, .org, as well as country code TLDs like .ca, .ie, .fr
- Many new TLDs have been added to the system since proposal in the year 2000
- Records at this level of the hierarchy contain only information about other (second-level) domain servers; no host records exist as TLDs
- TLDs are managed by the Internet Corporation for Assigned Names and Numbers (ICANN), which operates the Internet Assigned Numbers Authority (IANA)

Domain Hierarchy

Second-Level Domains

- Second-Level domains are intended to be delegated to organizations and individuals for their use
- Examples of second level domains include google.com, microsoft.com, or fanshawec.ca
- Records in a second-level domain may include additional sub-domains, or can also include host records
- Registrants of second level domains are responsible for maintaining the resource records for the zone

Domain Hierarchy

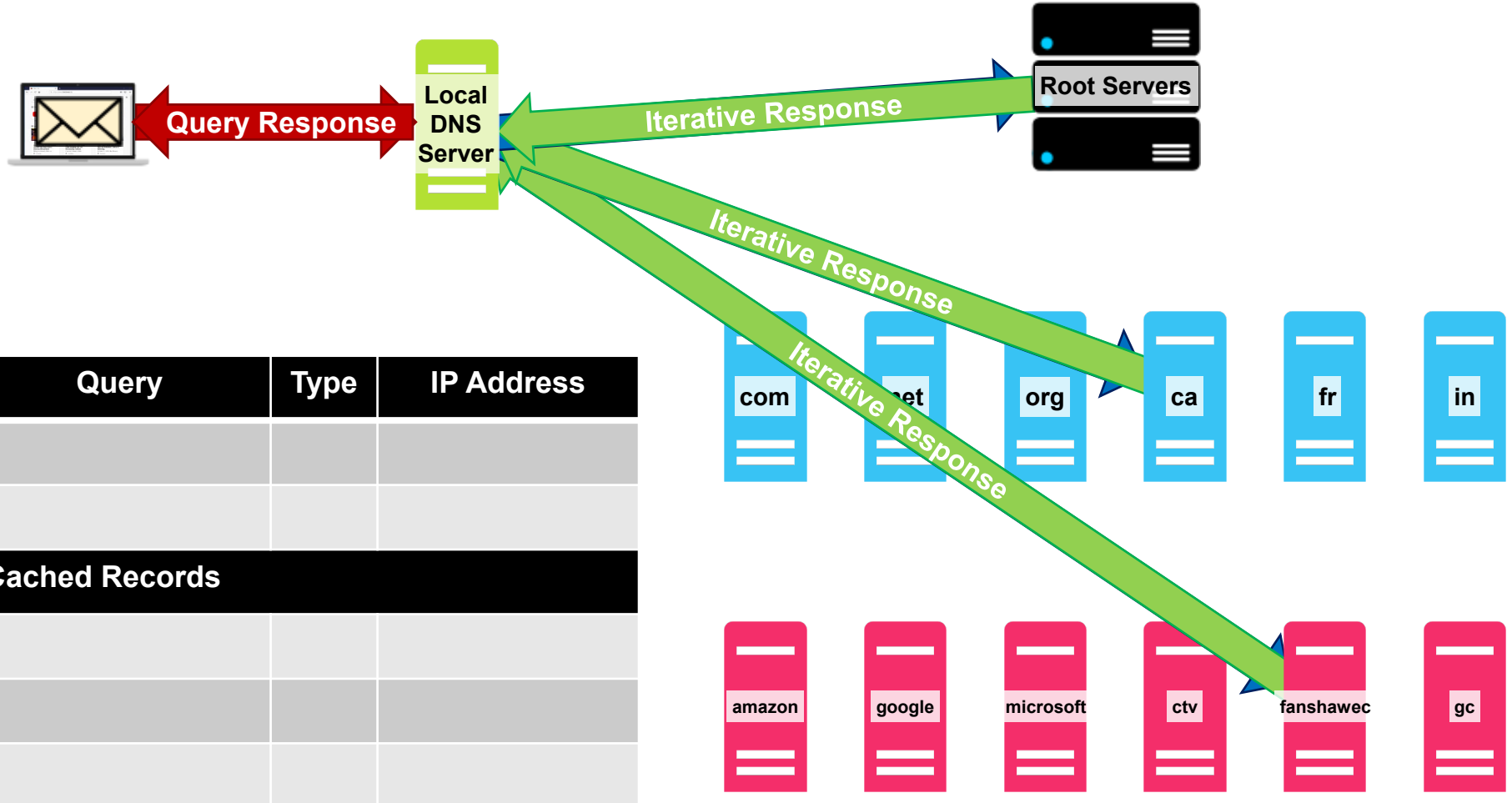
Subdomains

- Subdomains describe additional levels in the hierarchy created below the second-level domain
- In the following example, the subdomain describes sales:
  - **intranet.sales.example.com**
- Many additional subdomains can be created as long as the following statements remain true
  - The identifier at each level is 63 characters or less
  - The total FQDN is 255 characters or less

DNS Name Resolution

- DNS works from right to left on a query, first resolving the top-level domain, then the second-level domain, and so on
- DNS servers may answer two types of resolution requests:
  - **Recursive**
    - The DNS server receiving the query takes full responsibility for resolving the name in the query
    - If a server is hosting the zone (authoritative), or the record exists in cache, the answer is provided to the requestor
    - The server may send referrals to other DNS servers until it can answer the query
  - **Iterative**
    - A server that receives the query will immediately respond with the best answer available to it at the time
    - This information may be cached or authoritative, and may contain a referral

DNS Name Resolution



Query	Type	IP Address
Cached Records		

Record Caching

- A DNS server that has responded to queries maintains a cache of resolved records
- Records that are stored in cache not only include host records, but also the addresses of authoritative servers to improve the performance of future lookups in those domains
- Servers that are not authoritative for a zone is known as a caching-only server
- To manage how long a record should be retained, DNS employs a time-to-live (TTL) value for cached records

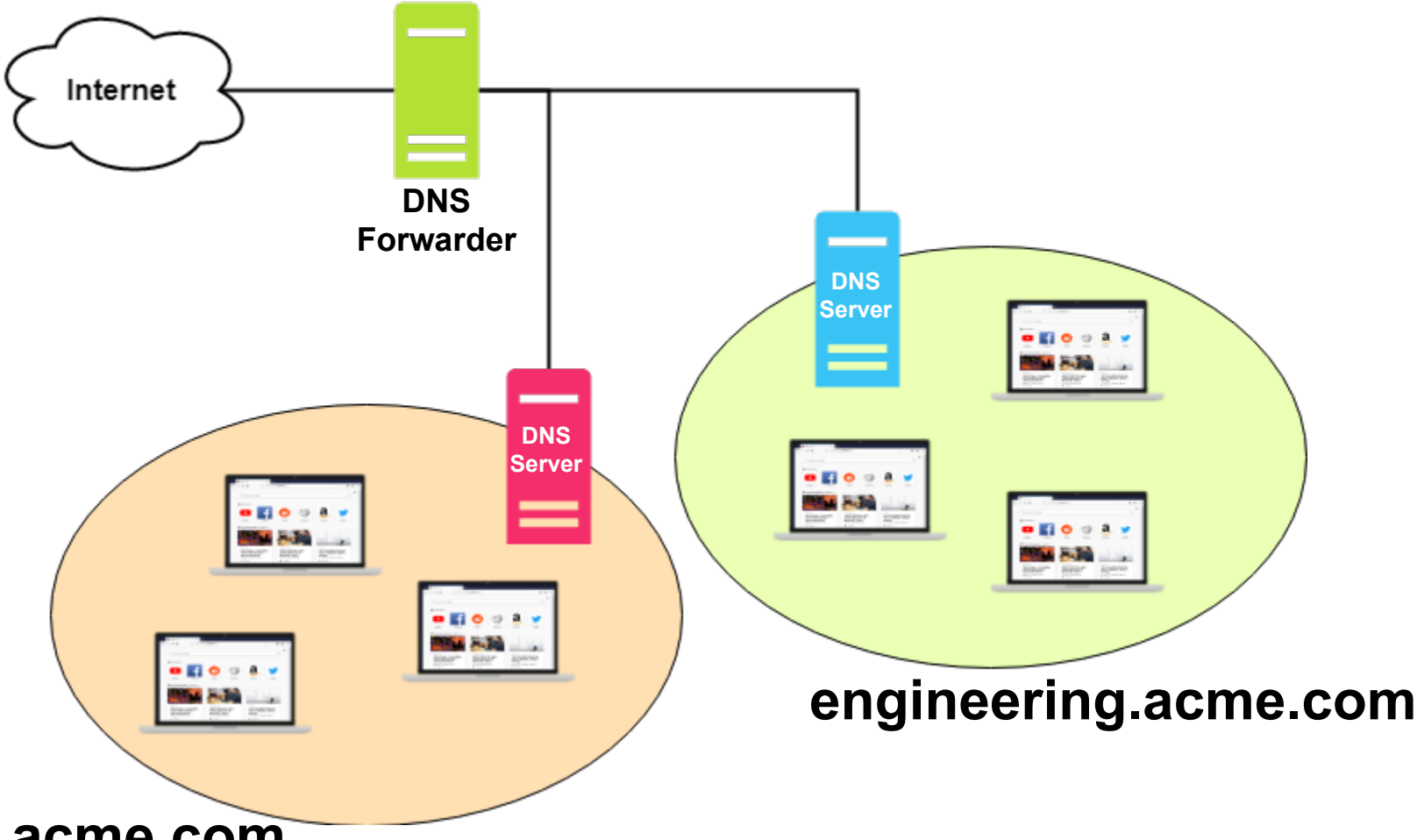
Record Caching

- The DNS TTL value is different from the IP TTL in the sense that it is measured in seconds, but also by the fact that administrator hosting the zone chooses the value of the TTL
- Some servers support a different TTL value for each record
- In addition to storing a cache of successfully resolved records, a DNS server will store a negative cache
- Negative caching occurs when a servers stores information about a record that does not exist in a domain
- Negative caching relies on the TTL of the SOA record to determine the value of the negative answer

DNS Forwarders

- In most circumstances, it is improper to send a recursive query to another DNS server
- If all DNS servers sent recursive queries to the root servers, the internet would come to a grinding halt, as the servers would be unable to process that many requests
- The only time a DNS server should send recursive queries is when it has been purposely configured as a forwarder
- Forwarders are used in networks that contain several DNS servers to limit the amount of repeated requests

DNS Forwarders



Reverse Name Lookups

- A reverse name lookup is used when we know the IP address of a system, but we need to find it's host name
- To accommodate reverse lookups, two special DNS zones exist in the top-level domain arpa
  - For IPv4 resolution the second-level domain is in-addr
  - For IPv6 resolution the second-level domain is ip6
- IP addresses are represented in reverse order when performing a reverse lookup
- For example, the reverse name query for the IPv4 address 192.168.7.65 would be 65.7.168.192.in-addr.arpa

DNS Record Types

Type	RFC	Description
A	1035	IPv4 address – Hostname to IP address mapping for IPv4
AAAA	3596	IPv6 address – Hostname to IP address mapping for IPv6
CNAME	1035	Canonical name – Alias that points to a canonical “real” name
MX	1035/7505	Mail exchange – Identifies message transfer agents for a domain
NS	1035	Name server – Identifies an authoritative nameserver for a given domain
PTR	1035	Pointer record – IP Address to hostname mapping (reverse lookup)
SOA	1035/2308	Start of authority – Indicates the server is authoritative for the zone
SRV	2782	Service locator – Specifies location of services
TXT	1035	Text record – Often used for Sender Policy Framework, DKIM & DMARC

Dynamic DNS (DDNS)

- Dynamic DNS (DDNS) provides a persistent method for updating resource records of hosts that regularly change location or IP addresses
- Generally, two functions exist to update records:
  - a standards-based extension of the DNS protocol called Dynamic DNS update, defined in RFC 2136
  - A web-based protocol where a client uses HTTP(S) accompanied with a password or API key to update the record
- Dynamic DNS updates support all record types, but is often used in coordination with DHCP to update host (A) records

Network Time Protocol (NTP)

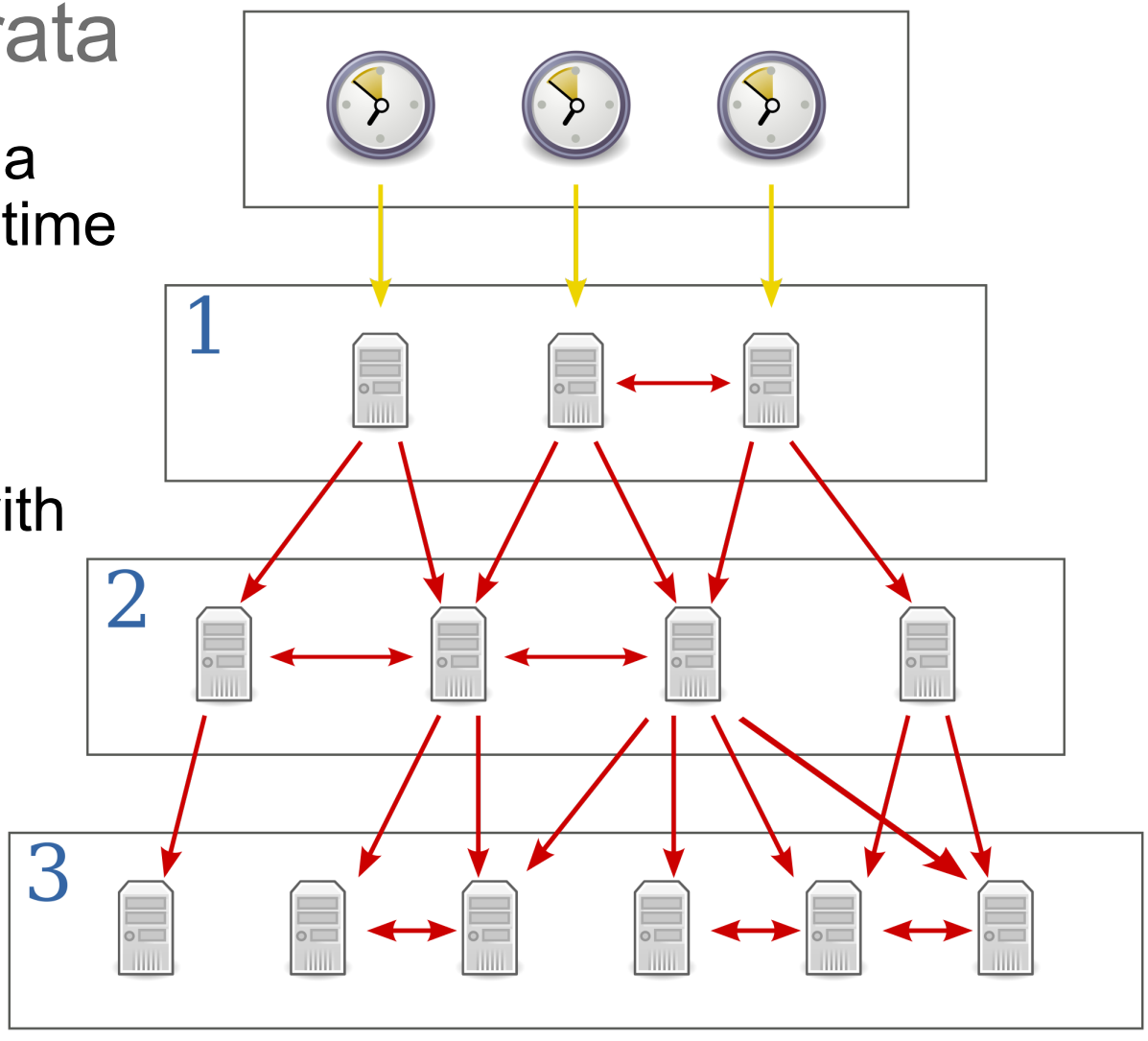
- When considering network logging and security devices, a network operating with accurate time becomes of utmost importance
- Seconds can make a huge impact when determining what step an attacker took first
- Network Time Protocol (NTP) is a client-server protocol used to synchronize clocks on network devices
- NTP is extremely efficient and can synchronize clock to within a millisecond on local networks while exchanging updates as little as once per minute

Network Time Protocol (NTP)

- Based on the intersection algorithm, NTP is intended to synchronize clocks within a few milliseconds of Coordinated Universal Time (UTC)
- NTP speakers exchange timestamps over UDP port 123
- Some configurations of NTP do not form client-server relationships and rely on broadcasting time to peer devices
- NTP has been updated as technology changes, with the current iteration being NTPv4

NTP – Clock Strata

- NTP is comprised of a hierarchal system of time sources
- Each level of the hierarchy is called a stratum, beginning with stratum 0 through 15
- As servers join the strata, they operate on a principle of n +1, where n is the stratum of the servers' time source



NTP – Clock Strata

- **Stratum 0**
  - Stratum 0 devices receive updates from a reference clock like GPS transmissions or atomic clocks
  - They cannot distribute time over the network directly
- **Stratum 1**
  - A stratum 1 server is directly connected to a stratum 0 device
  - They can peer with other stratum 1 devices for sanity checks

NTP – Clock Strata

- **Stratum 2**
  - A stratum 2 server is connected to a stratum 1 server over a network path
  - Stratum 2 servers will often link with multiple stratum 1 and stratum 2 devices

- As time is distributed through the hierarchy, the stratum number in incremented, up to stratum 15
- Stratum 16 is used to indicate that time is unsynchronized

NTP Security

- When accurate time is required, ensure the NTP time source does not fall under attack is important
- If the time source is a private clock, the administrator must ensure that the clock is secure, as if the time is successfully poisoned, an attacker can change the time in the organization
- If using public NTP servers, the administrator must trust the clock is reliable, accurate and secure
- Using IP spoofing, it may be possible to perform man-in-the-middle attacks on NTP clients and move clocks on client computers

NTP Security

- NTP has also been used in distributed denial of service (DDoS) attacks
- An NTP query with a spoofed source address is sent to an NTP server; the server will respond with a much larger message than the query, resulting in an amplification attack on the spoofed source address
- NTPv3 and later supports cryptographic authentication of messages to prevent spoofing attacks

NTP Operation

- A device that is the primary distributor of time in an environment is know as an NTP master
- NTP associations are then configured to reference the NTP master, if the device is both an NTP client and server, it will provide time to devices lower in the hierarchy
- Associations can also be configured with peer devices to enable sanity checking of movement to the NTP master's time
- NTP will compare time reported by several devices and will not synchronize to a source whose time is significantly different from the associated devices, even if it is of a lower stratum

References

- Network Time Protocol servers and clients.svg
  - Retrieved from: https://commons.wikimedia.org/wiki/File:Network_Time_Protocol_servers_and_clients.svg