École Pour l'Informatique et les Techniques Avancées – EPITA

BSc L1 – 17 May 2024

Course: Introduction to Computer Networks

Course instructor: M Salman Nadeem
mohammad-salman.nadeem@epita.fr

Introduction to Computer Networks

Course schedule (tentative)

Date & Time	No.	Topics	Duration (hours)
Fri 19/04/24 – 10:00–13:00	1	Primer, Network protocols, types, topology, architecture	3
Fri 26/04/24 – 10:00–13:00	2	Network models, TCP/IP model, Packet switching	3
Sat 27/04/24 – 10:00–13:00	3	Physical Layer (Function, Signals, Modulation, Multiplexing, Transmission media & Hardware, Optical networks)	3
Sat 27/04/24 – 14:00–17:00	4	Data Link Layer (Function, Framing, Protocols, Flow control, Access control, Error correction, Hardware)	3
Fri 03/05/24 – 14:30–17:30	5	Network Layer (Function, IP addressing and subnets)	3
Sat 04/05/24 – 10:00–13:00	6	Network Layer (Routing algorithms and protocols), Internet Control Message Protocol	3
Tue 14/05/24 – 16:30–19:30	7	Network Layer (IGP & EGP), Autonomous System, Border Gateway Protocol	3
Wed 15/05/24 – 14:30–17:30	8	Transport Layer (Function, Flow and congestion controls, Protocols)	3
Thu 16/05/24 – 11:15–13:15	9	Cross-layer process: Access Control Lists	2

Introduction to Computer Networks

Course schedule (tentative)

Date & Time	No.	Topics	Duration (hours)
Fri 17/05/24 – 14:00–17:00	**10**	**Application Layer (Function, Protocols)**	**3**
Sat 18/05/24 – 14:00–17:00	11	Cross-layer process: Network Address Translation	3
Fri 24/05/24 – 10:00–13:00	12	Review / Open-session	3
		Total	***35***
Fri 31/05/24 – 14:30–15:30		EXAM	1

GRADING criteria :

- Class participation comprising **attendance** & **reactivity**): 10%
- **Exercises** (practical work): 40%
- Final evaluation (**Quiz** & **Exercises**): 50%

Check policies in the course outline

Lecture 10 Outline

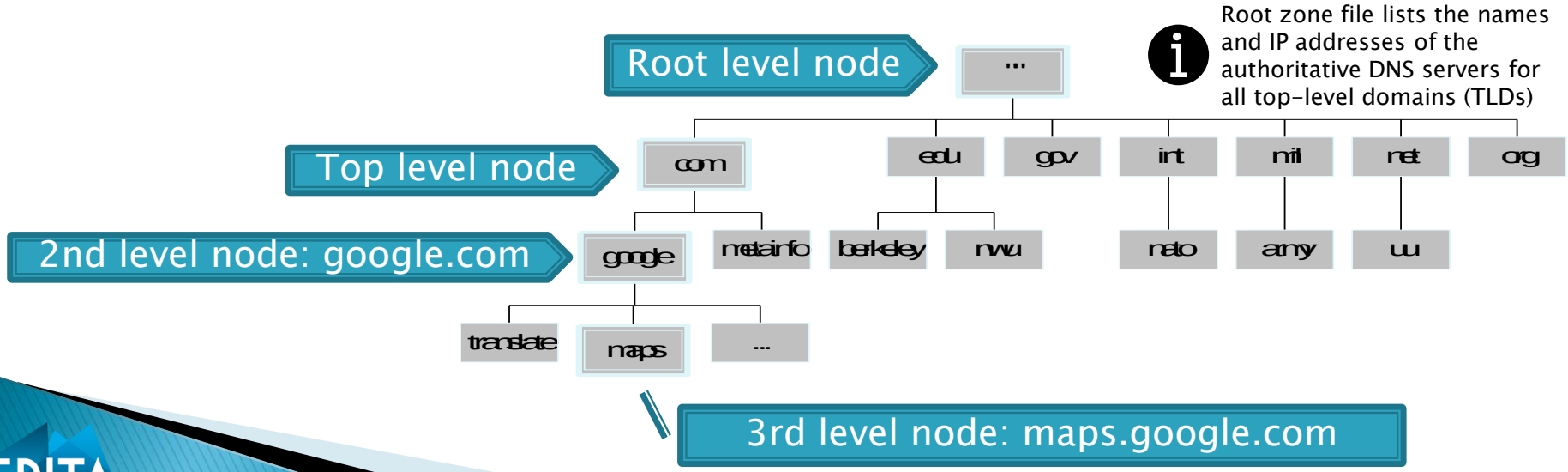
- **DNS**
  - **Lookup mechanism**
  - **Caching & updating records**
  - **Sub-domains & delegation**
  - **Domain name resolution process**
  - **Client look-up & cache**
  - **ICANN & TLDs**
  - **Registries, Registrars, and Registrants**
  - **Class exercise 15**
- Simple mail transfer protocol (SMTP)
  - SMTP & email structure
  - Working & limitations
  - PGP, S/MIME & PEM
  - Class exercise 16

Domain Name System (DNS)

- In early days of internet, all the information, like name to address mappings, was kept in a single file (e.g., hosts.txt) shared with all the computers connected to the Internet (ARPANET)
  - This became quickly unmaintainable (too much of 'bookkeeping', and was fixed by DNS
- Now, we refer to devices/hosts on the Internet by domain names 'epita.fr', or 'Fully Qualified Domain Names' (FQDN) – absolute domain name – if it also includes all labels (e.g., somehost.epita.fr.)
  - Simply because names are easy to remember (than numbers)
  - The mechanism by which they get translated to an IP address and vice versa is handled by **DNS**
  - DNS was specified in RFC 1034 & 1035, and uses the UDP on port 53 (to serve requests)
- DNS has **three major components**:
  1. Domain Name Space and Resource Records
  2. Name Servers
  3. Resolvers

Domain Name Space

- Domain Name Space is a (distributed) tree data structure
  - Each node in this tree has a label
  - The root node (written as “”) has an empty label
    - No other label can be empty
  - A sequence of **labels** from a node to the root, separated by dots (‘.’) is read from left to right, and has some limits:
    - 127 levels (root to sub-domains) at max.
    - Each level: 63 chars per label at max.
    - All labels or a domain name must can have at max. 253 chars



To re-iterate

- The parts of a domain name

  abc.epita.edu

- The particular host (or label) is called ‘abc’
- The organization that controls it is called ‘epita’
- This host is at an educational organization (TLD: .edu)

A domain is a group of labels

Resource Records (RR)

- Each node in Domain Name Space can keep information in the form of resource record sets (RRset) containing:
  - Owner: of this record (domain name of the node which keeps this information)
  - Type: a 16-bit value, specifying what this resource record is about (e.g., SOA – identifies the start of a zone, NS – DNS server, A – IPv4, AAAA – IPv6)
  - Class: a 16-bit value, identifying a protocol (e.g., IN – internet, ANY – used only in queries)
  - TTL: a 32-bit Time-To-Live value, in units of seconds, indicating how long this resource record can be cached
  - rdata: data of the resource (depends on the resource type)
- All resource records in a resource record set has the same name, type and class

Name servers (NS)

- Name servers (e.g., BIND) are server programs, which hold some information about the Domain Name Space
  - Each name server may have complete and authoritative information only about a small part of the Domain Name Space, and possibly cache some other data in the Domain Name Space
- Data is maintained locally, but retrievable globally
  - No single device has all DNS data – and DNS lookups can be performed by any device
- Act as a distributed directory service
  - Resources names <-> respective addresses

Name servers (NS): Bind

```
msn-box@msnbox-vm:~$ systemctl status bind9
● named.service - BIND Domain Name Server
     Loaded: loaded (/lib/systemd/system/named.service; enabled; vendor preset
     Active: active (running) since Sat 2020-12-12 14:27:16 CET; 21s ago
       Docs: man:named(8)
   Main PID: 2194 (named)
      Tasks: 5 (limit: 4650)
     Memory: 27.5M
     CGroup: /system.slice/named.service
             └─2194 /usr/sbin/named -f -u bind

Dez 12 14:27:17 msnbox-vm named[2194]: network unreachable resolving './DNSKEY
Dez 12 14:27:17 msnbox-vm named[2194]: network unreachable resolving './NS/IN'
Dez 12 14:27:17 msnbox-vm named[2194]: network unreachable resolving './DNSKEY
Dez 12 14:27:17 msnbox-vm named[2194]: network unreachable resolving './NS/IN'
Dez 12 14:27:17 msnbox-vm named[2194]: network unreachable resolving './DNSKEY
Dez 12 14:27:17 msnbox-vm named[2194]: network unreachable resolving './NS/IN'
Dez 12 14:27:17 msnbox-vm named[2194]: network unreachable resolving './DNSKEY
Dez 12 14:27:17 msnbox-vm named[2194]: network unreachable resolving './NS/IN'
Dez 12 14:27:17 msnbox-vm named[2194]: managed-keys-zone: Initializing automat
Dez 12 14:27:17 msnbox-vm named[2194]: resolver priming query complete
```

BIND status

```
options {
        directory "/var/cache/bind";

        // If there is a firewall between you and nameservers you want
        // to talk to, you may need to fix the firewall to allow multiple
        // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

        // If your ISP provided one or more IP addresses for stable
        // nameservers, you probably want to use them as forwarders.
        // Uncomment the following block, and insert the addresses replacing
        // the all-0's placeholder.

        // forwarders {
        //      0.0.0.0;
        // };

        //=====================================================================
===
        // If BIND logs error messages about the root key being expired,
        // you will need to update your keys.  See https://www.isc.org/bind-key
s
        //=====================================================================
===
        dnssec-validation auto;

        listen-on-v6 { any; };
};
~
<etc/bind/named.conf.options" [readonly] 24L, 846C          1,1          All
```

BIND config: /etc/bind/named.conf

Bind acts as an authoritative name server for domains, and as well as a recursive resolver in the network

Resolvers

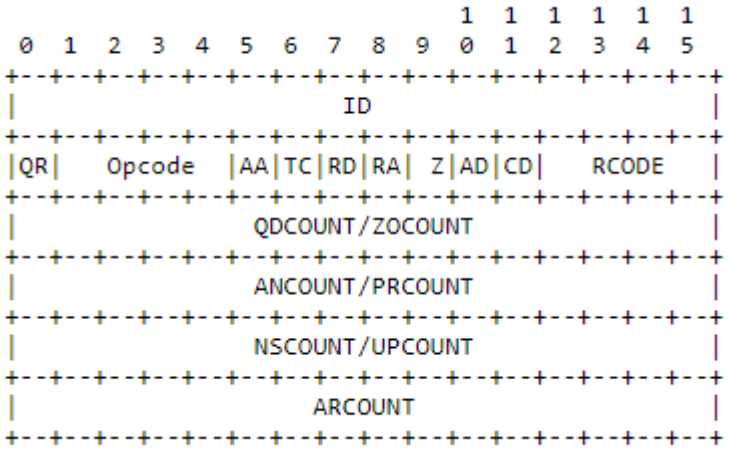
- Programs in user device
  - Send queries to name servers (NS), to extract information from Domain Name Space -> this process is called DNS lookup
    - E.g., dig: a DNS lookup utility
- Iterative vs. Recursive Queries
  - Iterative approach: where the resolver repeatedly queries different servers until it finds the answer
  - Recursive approach (common one): where the resolver sends the query only to a single server, then that server repeatedly queries different name servers until it finds and returns the answer to the resolver
- If a DNS response is large, DNS lookup can be retried over TCP

DNS Header section & format

A resolved domain name is cached by NS:

- It expires (under the defined TTL: Time To Live)
- TLD's typically get cached in local name server
- Root name servers are not often visited

Details of the Header section



DNS message always has the same format, it does not matter if it is a query or response

Header	
Question	the question for the name server
Answer	RRs answering the question
Authority	RRs pointing toward an authority
Additional	RRs holding additional information

Source: https://tools.ietf.org/html/rfc2929

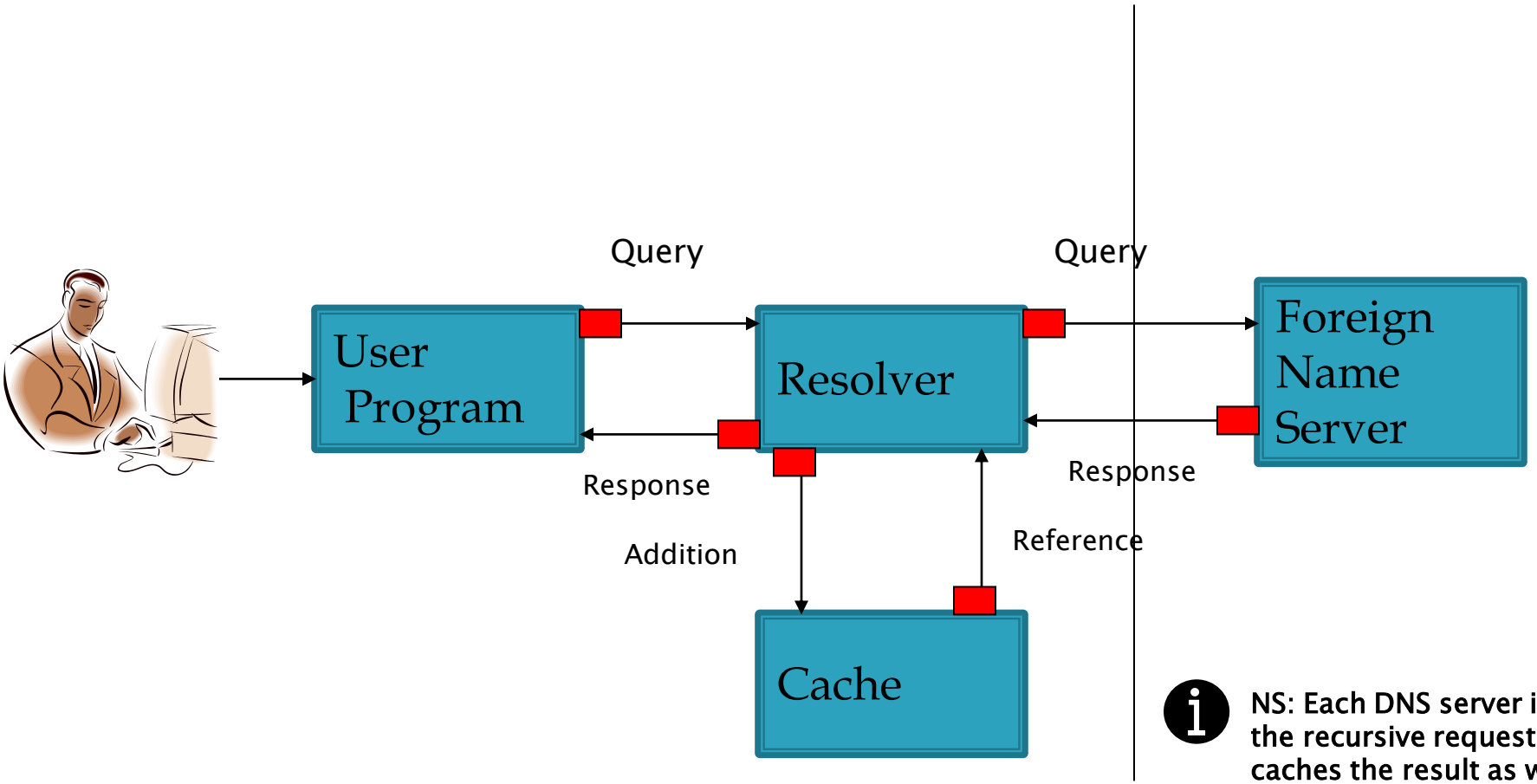
Resolution process (1/2)

- Take ‘dig’ DNS look-up utility as an example
  - The query is sent to the system default name server (e.g., on GNU/Linux -> /etc/resolv.conf)
  - Only answer is displayed, query is hidden by default
- Query any domain using private or public DNS, and enable the display of query request
  - dig @8.8.8.8 epita.fr IN A +qr
- Lets see the effect of Recursion Desired (RD) by not setting it with +nordflag:
  - dig @8.8.8.8 epita.fr IN A +qr +nordflag
  - Because Google’s Public DNS (8.8.8.8) is not authoritative for epita.fr, when no recursion is set, it simply returns nothing (or a SERVFAIL)

Resolution process (2/2)

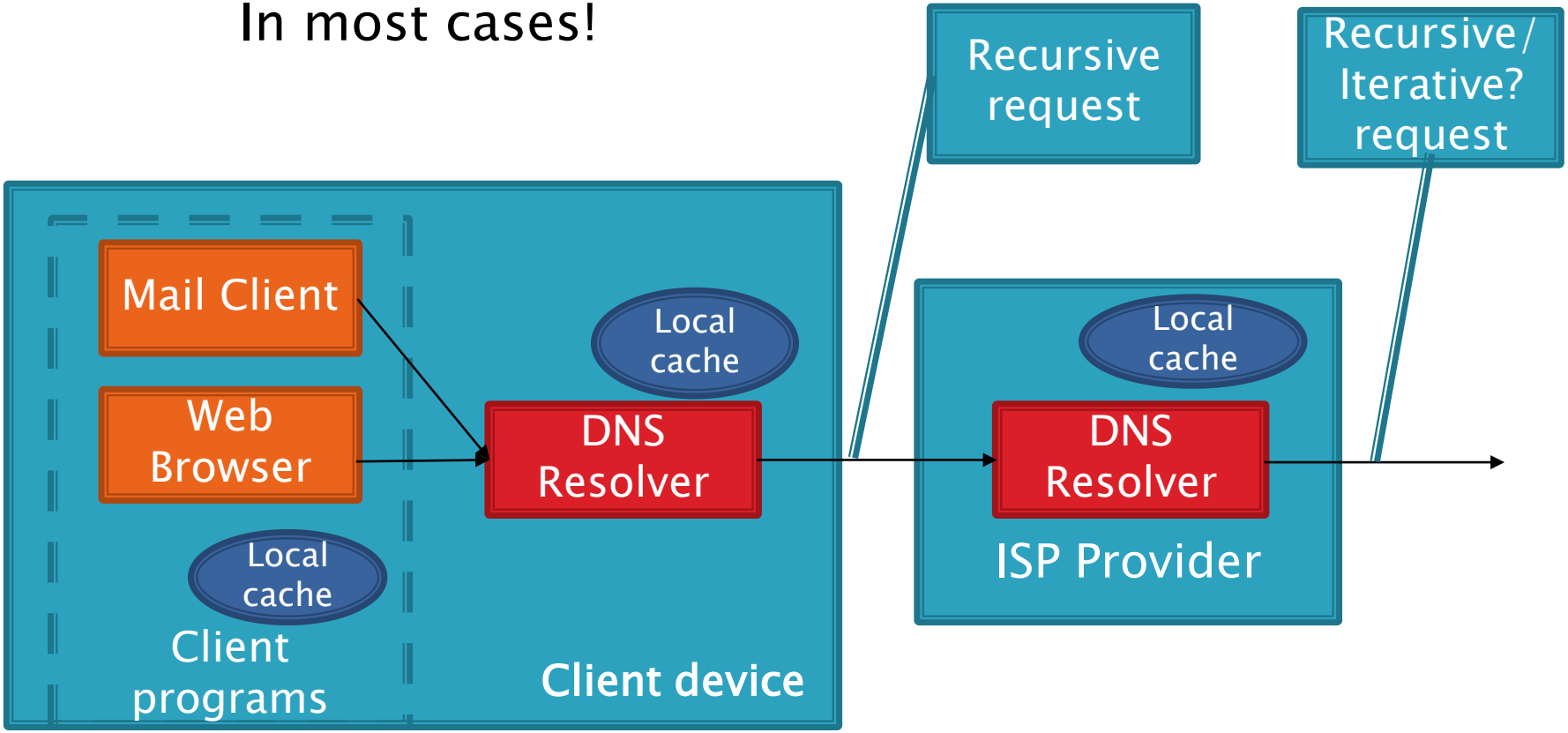
- Find the authoritative name server of a given domain name:
  - Let us start from the root domain:
    1. dig //root name servers
    2. dig @a.root-servers.net fr NS +qr +nordflag
       //name servers of 'com' -> authoritative
    3. dig @e.ext.nic.fr epita.fr NS +qr +nordflag
       //name server of 'epita.fr'
    4. Dig @banjo.ionis-it.com epita.fr A +qr +nordflag
       //Using authoritative name servers of epita.fr to ask resource records: A (IPv4)
    5. …

Client lookup & Local cache (1 /2)



Client lookup & Local cache (2/2)

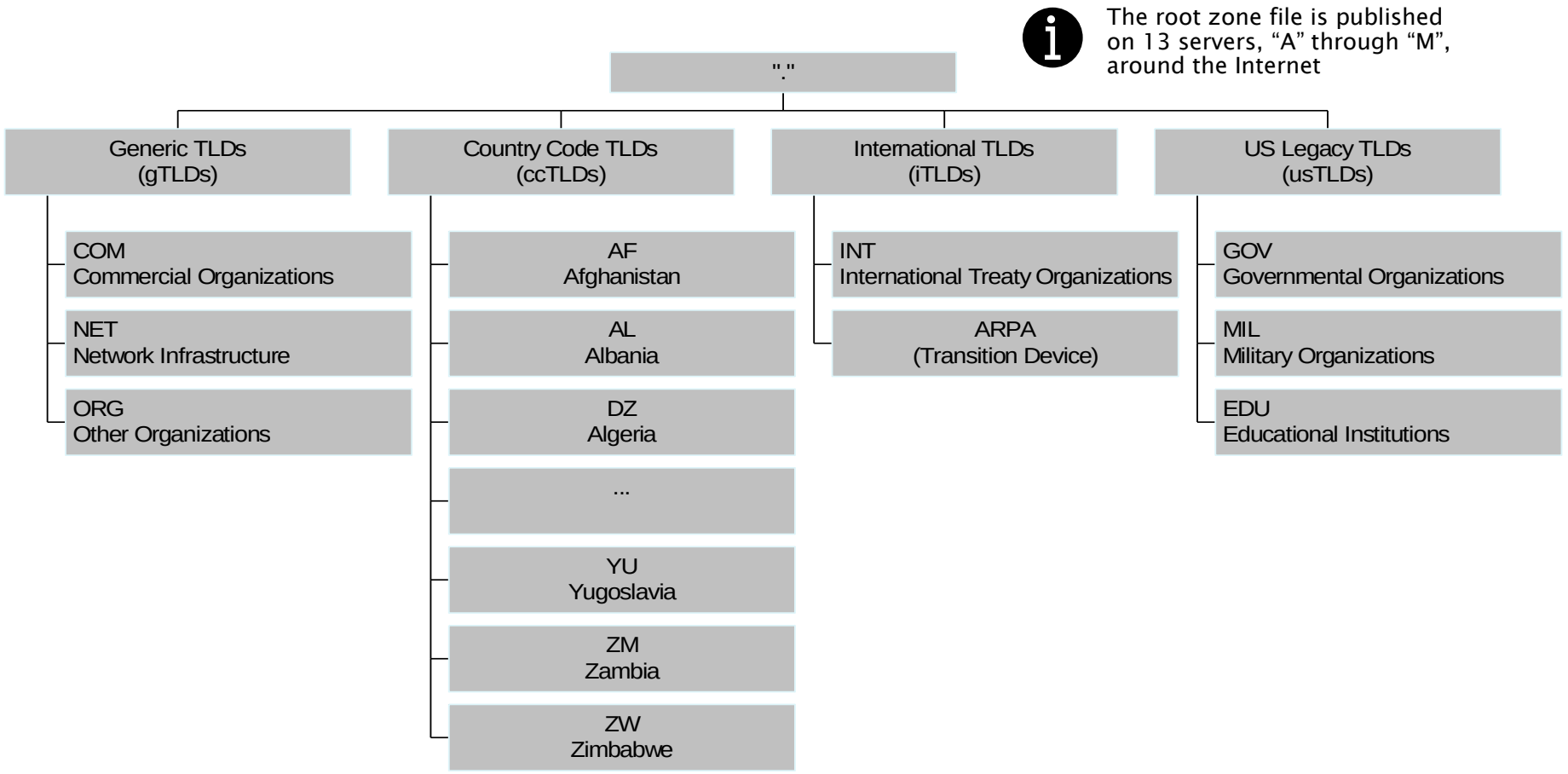
In most cases!



Internet Corporation for Assigned Names and Numbers (ICANN)

- ICANN's role in DNS hierarchy: to oversee the management of Internet resources including
  - Addresses
    - Delegating blocks of addresses to the regional registries
  - Protocol identifiers and parameters
    - Allocating port numbers, etc
  - **Names**
    - **Administration of the root zone file**
    - **Oversee the operation of the root name servers**
- Connecting to the Internet implies use of the existing DNS hierarchy – that's the rule!

The Current TLDs

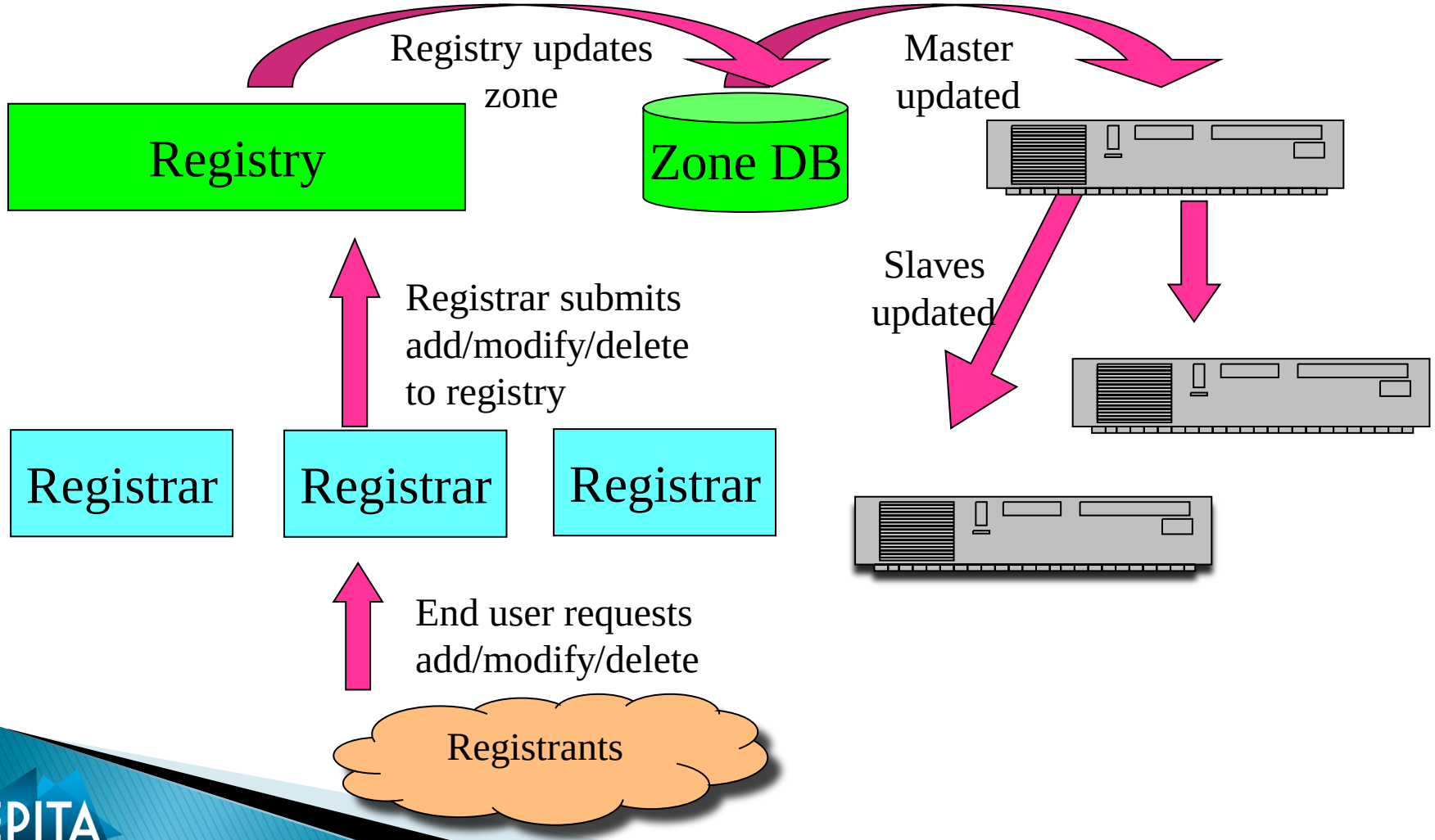


Registries, Registrars, and Registrants

A classification of roles in the operation of a domain name space

- Registry
  - The name space's database
  - The organization which has edit control of that database and runs the authoritative name servers for that name space
- Registrar
  - The agent which submits change requests to the registry on behalf of the registrant
- Registrant
  - The entity which makes use of the domain name

Registries, Registrars, and Registrants



Exercise 15a: Practical work

- Perform DNS queries and study the responses, using following tools:

-> dig
dig epita.net
dig @1.1.1.1 epita.net //1.1.1.1 is the specified name server
dig MX epita.net //Replace MX with other DNS record types e.g., A, CNAME, AAAA, NS, …

-> nslookup
nslookup epita.net
nslookup epita.net 1.1.1.1 //1.1.1.1 is the specified name server
nslookup -debug epita.net //-d2 for more verbose answer

-> host
host -a epita.fr

Direct 'traceroute' to DNS server will likely not succeed, use 'traceroute -U X.X.X.X' instead

Save your txt/doc file with your 'First_Last name'

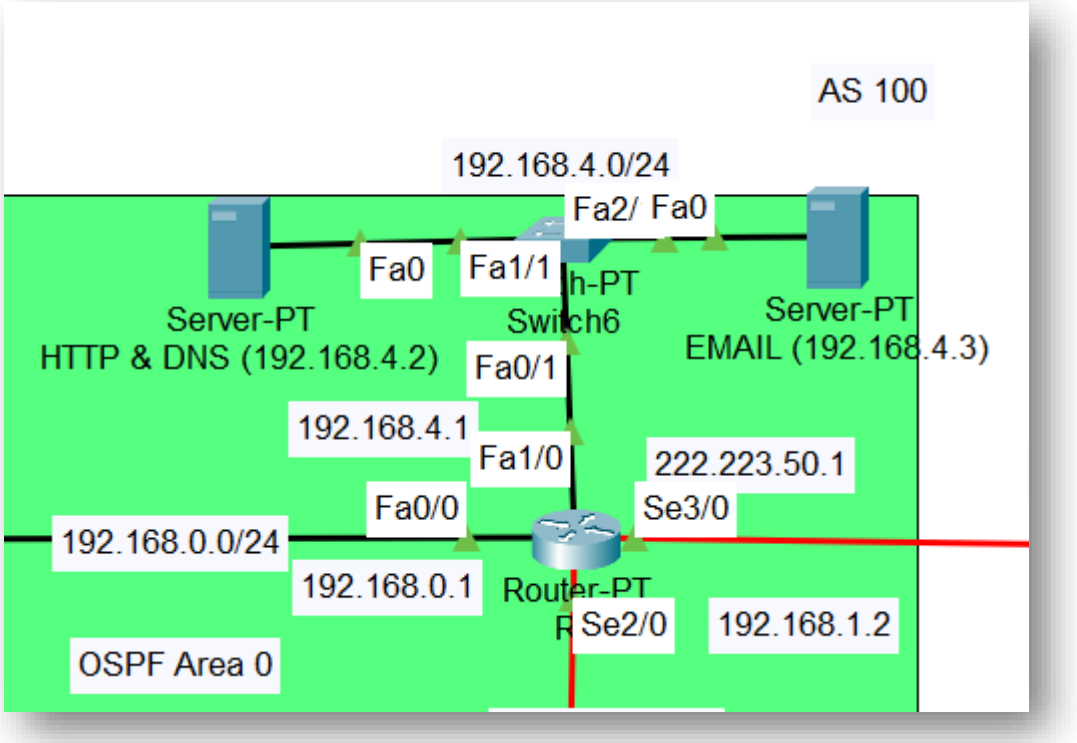
Deadline: See 'Teams' Assignment section

Exercise 15b: Practical work

- Using your last cisco packet tracer file:
  1. Configure DNS service on the HTTP server – at ASBR router (AS 100), in 192.168.4.0/25 network
  2. Configure the DNS server on all end-devices

Deadline: See 'Teams' Assignment section

Save your txt/doc file with your 'First_Last name'



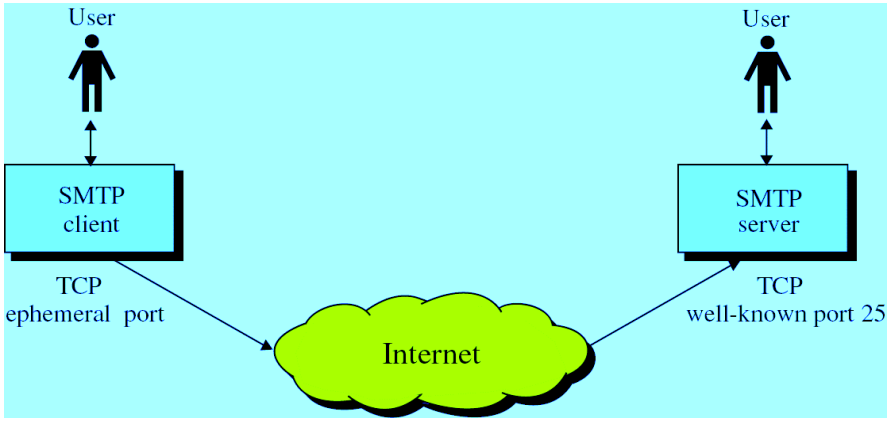
Lecture 10 Outline

- DNS
  - Lookup mechanism
  - Caching & updating records
  - Sub-domains & delegation
  - Domain name resolution process
  - Client look-up & cache
  - ICANN & TLDs
  - Registries, Registrars, and Registrants
  - Class exercise 15
- **Simple mail transfer protocol (SMTP)**
  - **SMTP & email structure**
  - **Working & limitations**
  - **PGP, S/MIME & PEM**
  - **Class exercise 16**

Simple mail transfer protocol (SMTP)

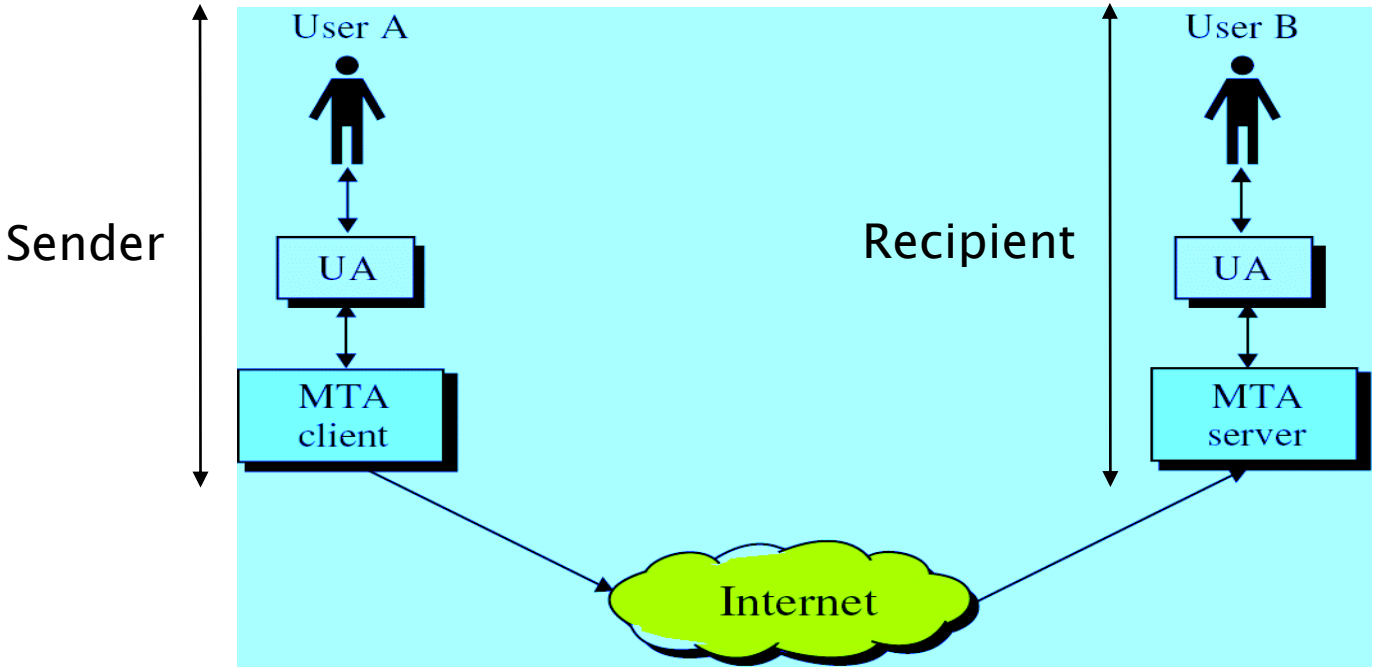
- First defined in 1982 under RFC 821
  - The goal was to make email sending mechanism simpler
- When an e-mail is sent from a sender to receiver, in most cases this involves:
  - The sender machine sends an email to local SMTP server, which then sends email to recipient's SMTP server, and finally the recipient pulls/fetches it from there

Goal: To transfer mail reliably and efficiently



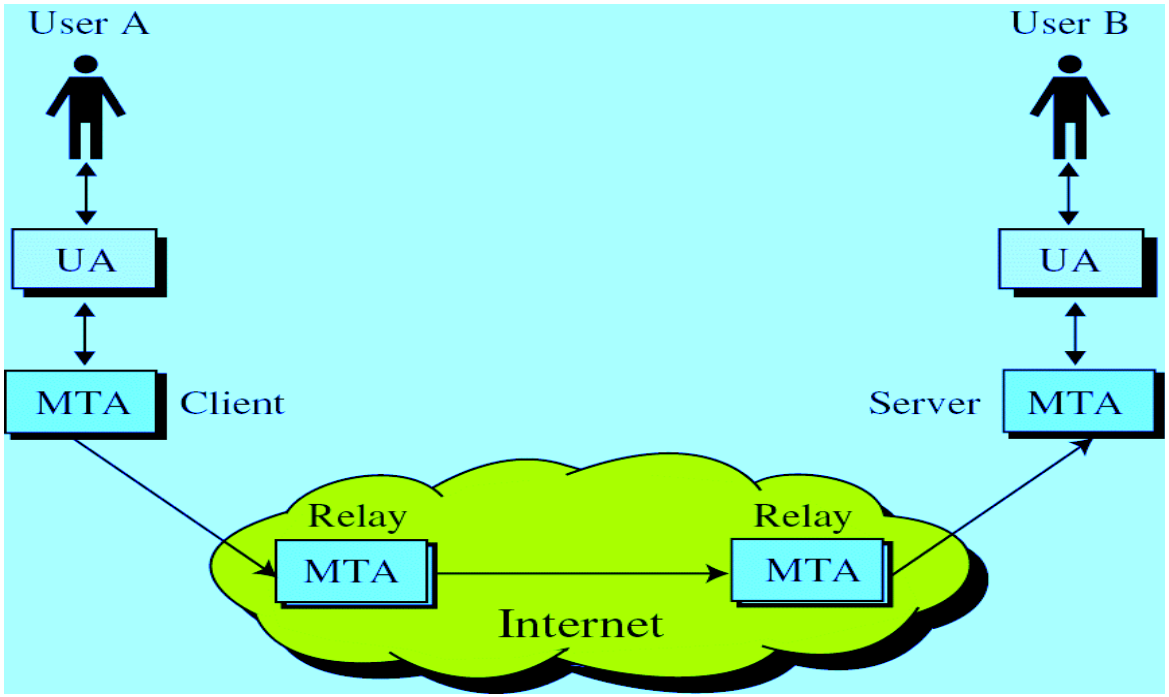
Simple mail transfer protocol (SMTP) – Cont.

- SMTP **clients** and **servers** have two main components:
  - Mail User Agent (MUA): Prepares the message and encloses it in an envelope (for e.g., Thunberbird, ...)
  - Mail Transfer Agent (MTA): Perform the actual transfer of email (e.g., Postfix, Exim, Qmail...)



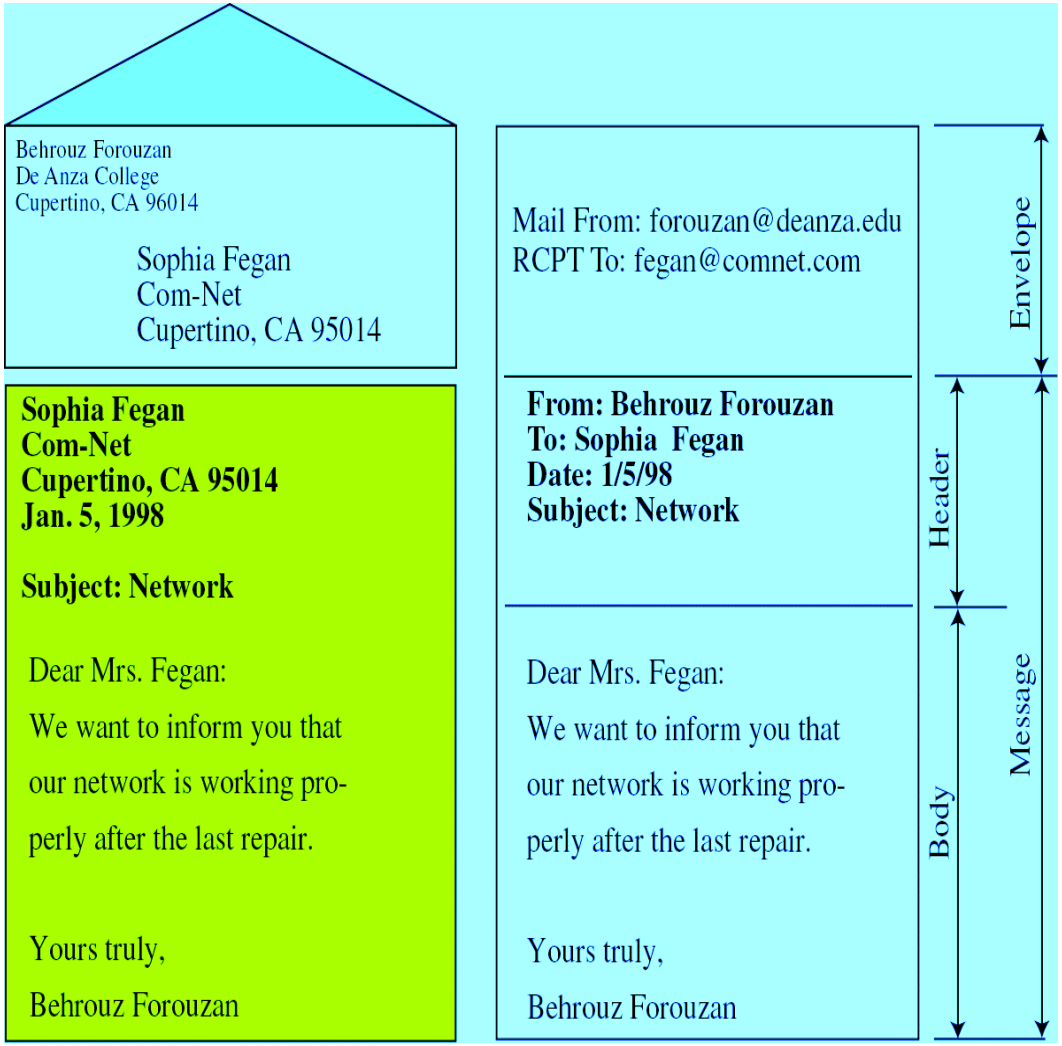
Simple mail transfer protocol (SMTP) – Cont.

- SMTP also allows the use of **Relays** (a type of MTA) allowing other MTAs to 'forward' the email to another MTA for further delivery

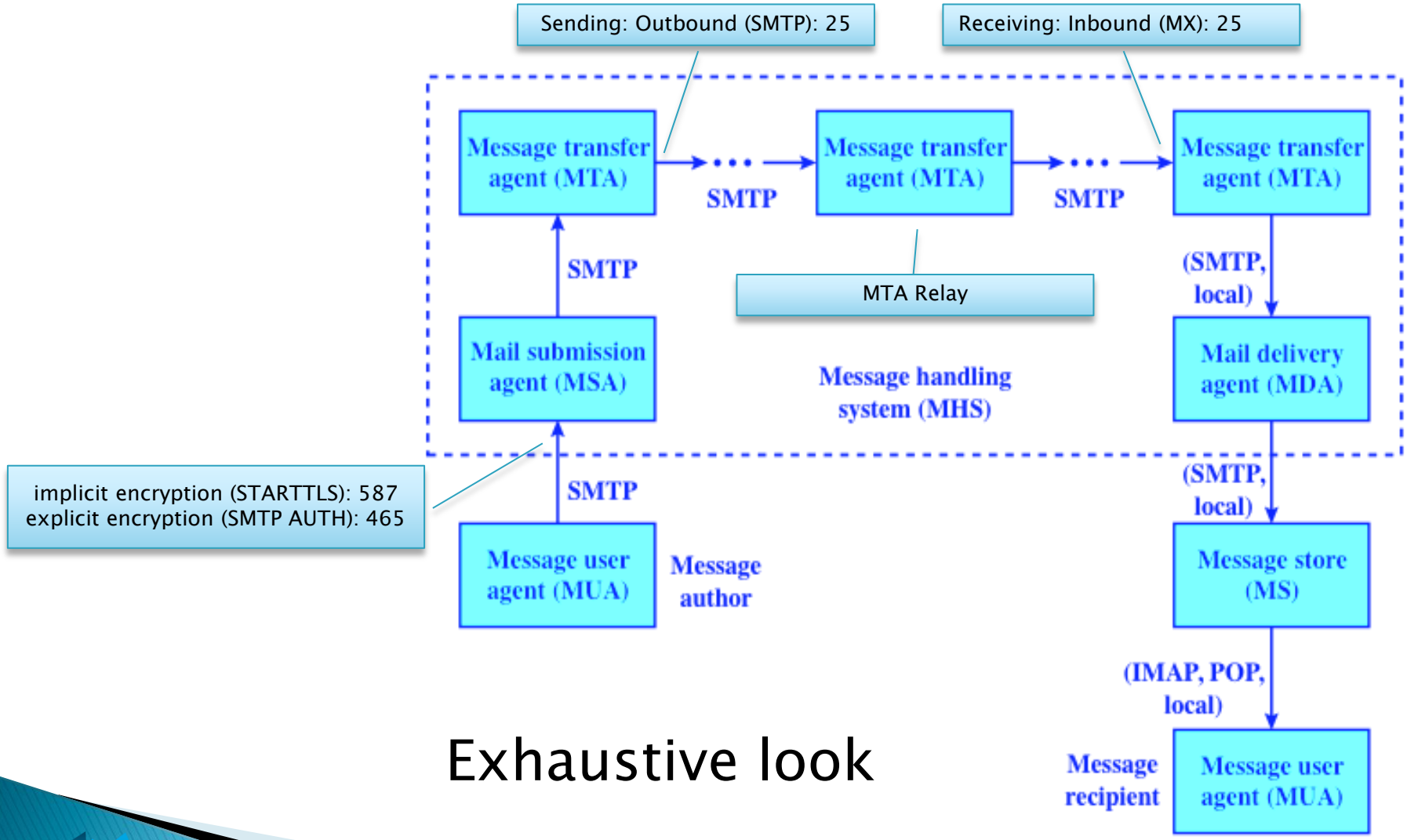


Structure of an email

- Email is a text file
- Envelope:
  - Sender address
  - Receiver address
  - Other information
- Message:
  - Mail Header: defines the sender, receiver, subject of the message, and some other information
  - Mail Body: contains the actual information in the message



Simple mail transfer protocol (SMTP) – Cont.



Exhaustive look

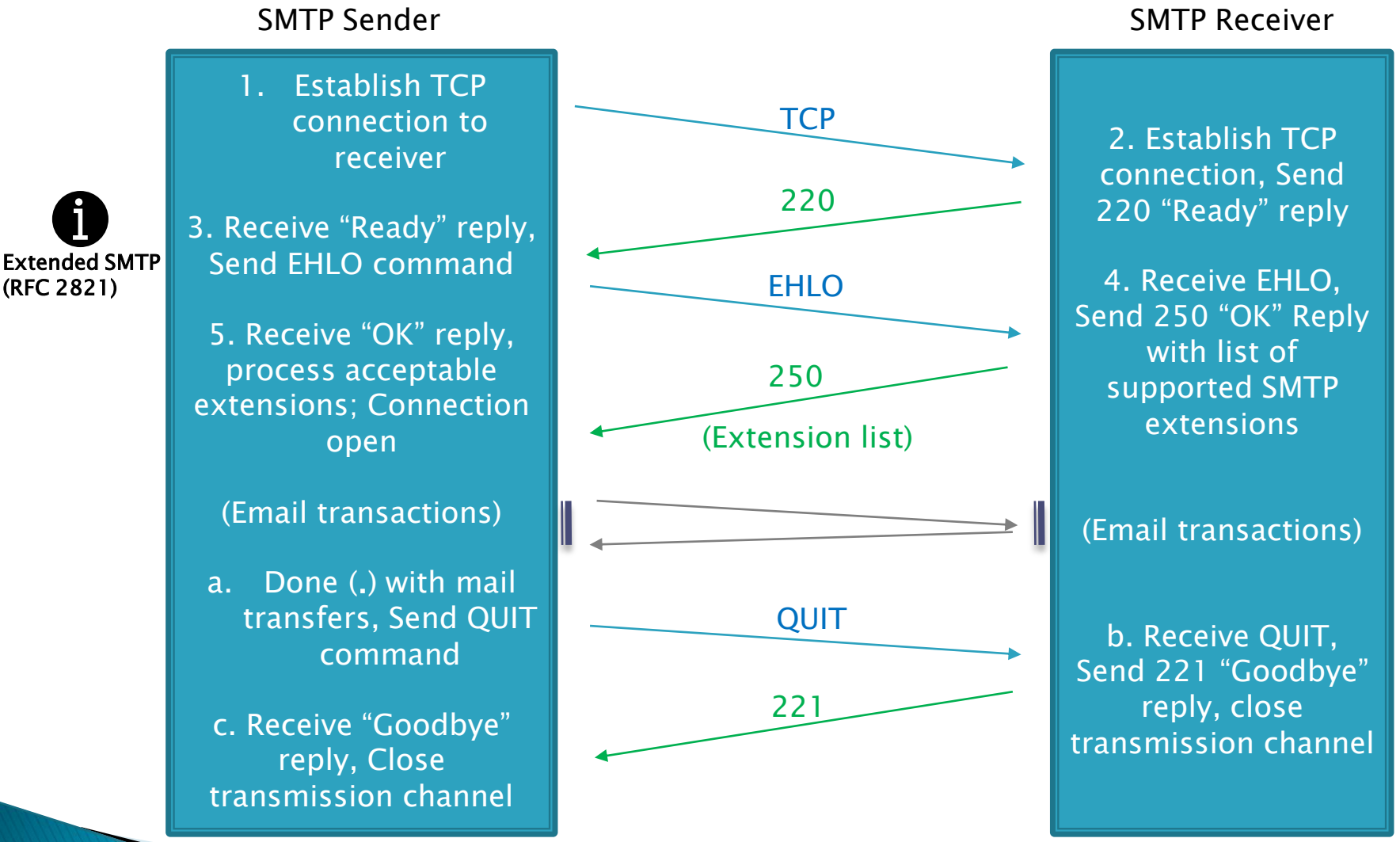
SMTP protocol: dialogue keywords

Client keyword	Arguments
HELO	Sender's Host Domain Name
MAIL FROM:	Email Address of sender
RCPT TO:	Email of Intended recipient
DATA	Body of the message
QUIT	

- The Server responds with 3 digit code that may be followed by text info
  - 2## – Success
  - 3## – Command can be accepted with more information
  - 4## – Command was rejected, but error condition is temporary
  - 5## – Command rejected, Bad User!
  - …

Ref.: www.iana.org/assignments/smtp-enhanced-status-codes/smtp-enhanced-status-codes.xhtml

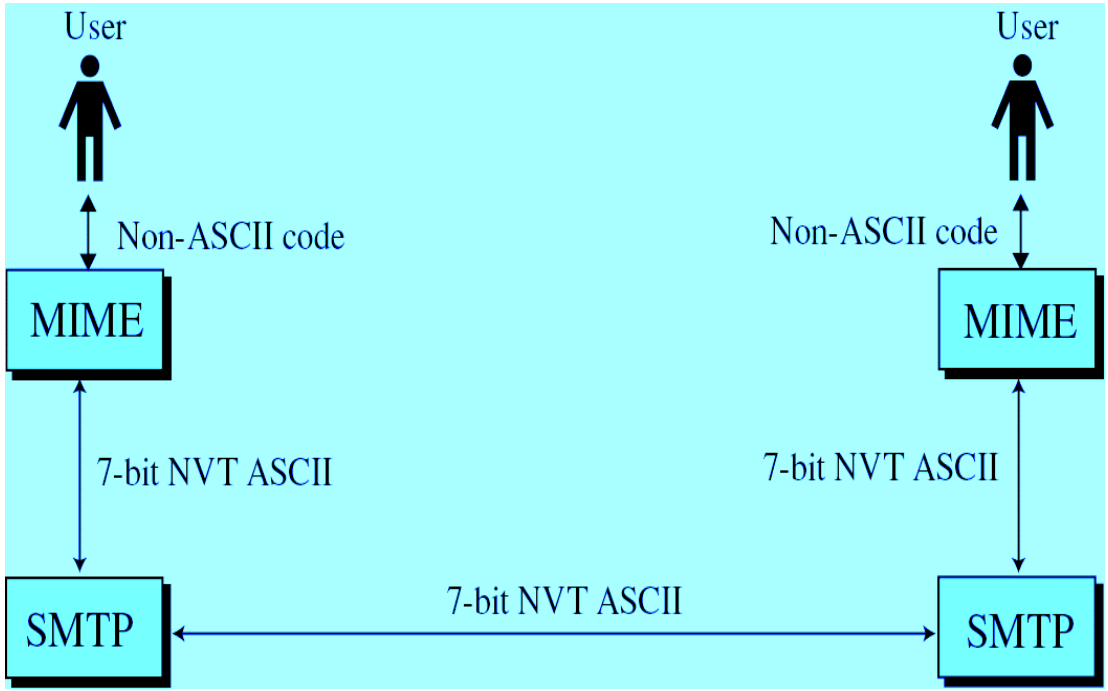
Message transaction progress



Extended SMTP (RFC 2821)

Extensions to SMTP

- MIME – Multipurpose Internet Mail Extensions.
  - Transforms non-ASCII data to NVT (Network Virtual Terminal) ASCII data by encoding it into chunks
    - Text
    - Application
    - Image
    - Audio
    - Video
    - …

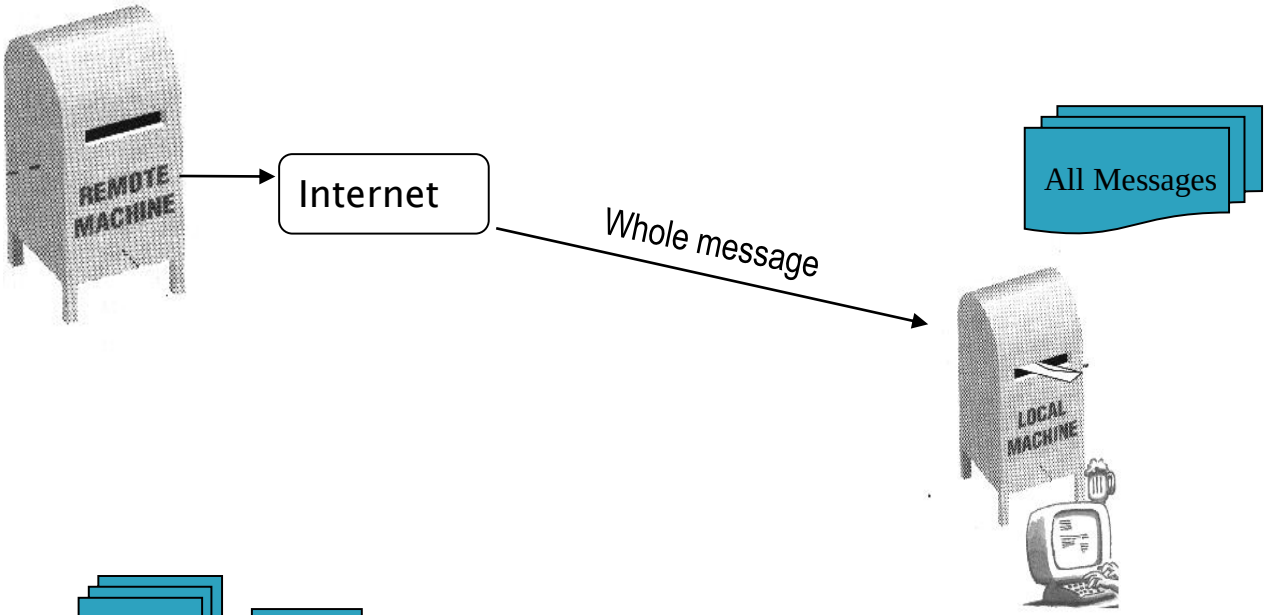


Mail Access Protocols

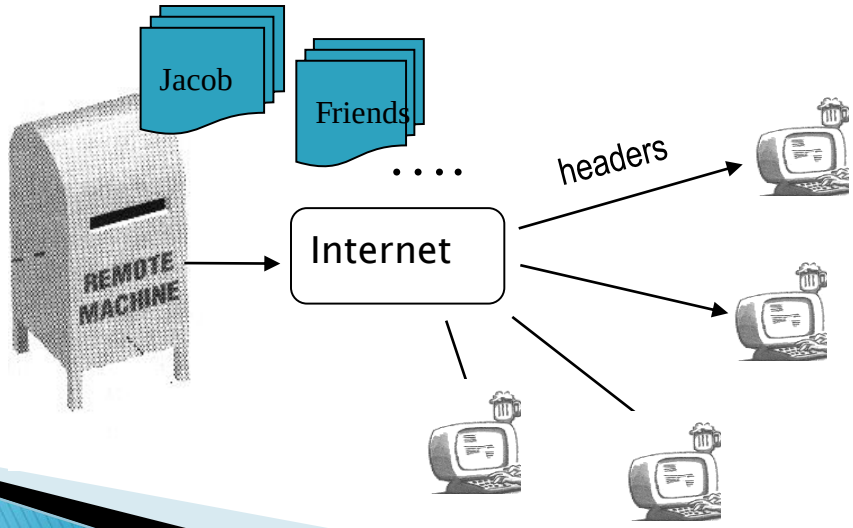
- The MTA delivers email to the recipients mailbox using Mail Delivery Agent (MDA)
- The Mail Access Protocols are used by the users to retrieve emails from local SMTP Server (or MDA)
  - Post Office Protocol version 3 (POP3)
    - POP3 (port: 110), POP3-S (port: 995)
  - Internet Message Access Protocol version 4 (IMAP)
    - IMAP4 (port: 143), IMAP4-S (port: 993)

POP vs IMAP

POP3:



IMAP4:



Simple SMTP limitations

- Only uses NVT (Network Virtual Terminal) 7-bit ASCII format
- Timeouts problem – If the Client and server have different timeouts, one of them may give up while the other is still busy, unexpectedly terminating the connection
- No authentication mechanisms
- Messages are sent un-encrypted
- Susceptible to misuse (Spamming, faking sender address, ...)
- …

PGP, SMIME and PEM

Available tools:

- GnuPG (command-line)
- Mailvelope (browser-extension)
- Mailfence/ Protonmail (Webmail based)
- …

	C	I	A
PGP: It incorporates mechanisms for authentication, confidentiality, compression, e-mail compatibility and segmentation & reassembly . MAIL & FILE	Symmetric encryption-CAST-128, 3-DES, IDEA	SHA	DSS+ SHA or RSA+SHA
SMIME: S/MIME provides the functionality of Enveloped data, signed data, clear signed data and signed and enveloped data. MIME	Diffe-Hellman (Key Exchange) Triple-DES or RC2/40	SHA-1/MD5	SHA-1/MD5 + DSS/RSA
PEM: Mechanism of key management for authentication purposes Text Based	DES	MD2/MD5	DES+MD5

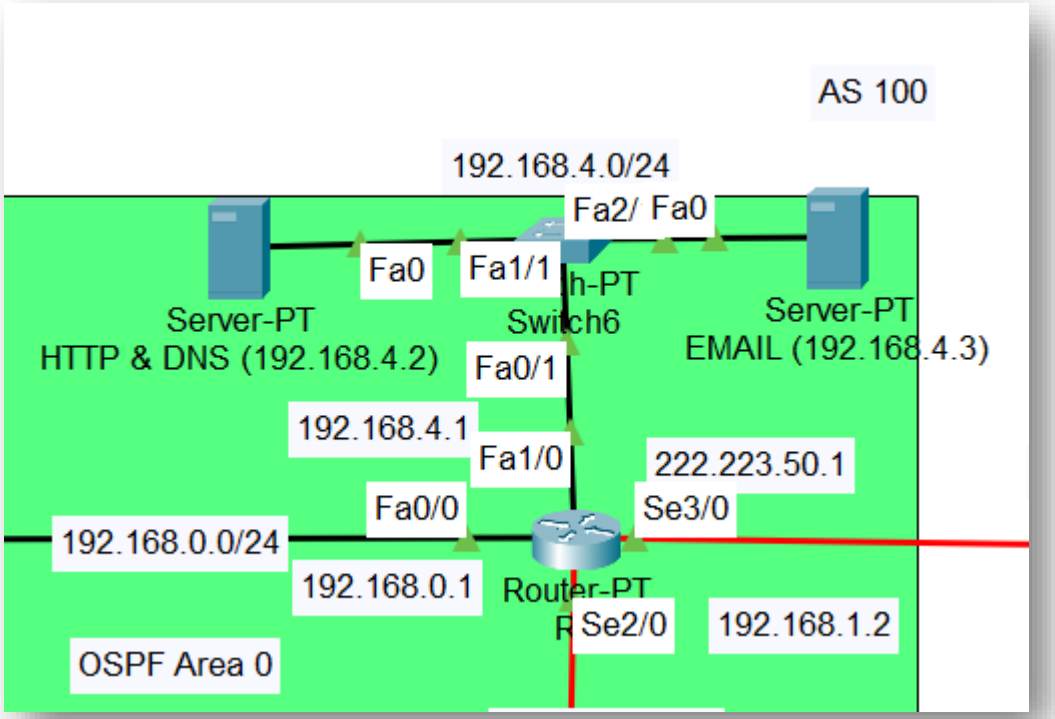
Note: SHA1, MD5, DES, 3DES algorithms are now considered as obsolete

Exercise 16: Practical work

- Using your last cisco packet tracer file:
  1. On the ASBR router (AS 100), setup an SMTP and HTTP server (in 192.168.4.0/25 network)
  2. For SMTP: Configure 2 users i.e., alice and bob
     1. From Bob's device, send an email to alice
  3. Verify using PING and TRACERT/TRACEROUTE, and by visiting the web-site
  4. **Study the packet transmission** in Simulation mode

Save your txt/doc file with your 'First_Last name'

Deadline: See 'Teams' Assignment section



Lecture 10 ends here

- Course Slides: Go to MS Teams:
  'Introduction to Computer Networks - Spring 2024 | BSc'
  -> Files section

- Send your questions by email:
  mohammad-salman.nadeem@epita.fr
  OR via direct message using MS Teams

- Thank You!