



Computer Networks Advanced Course

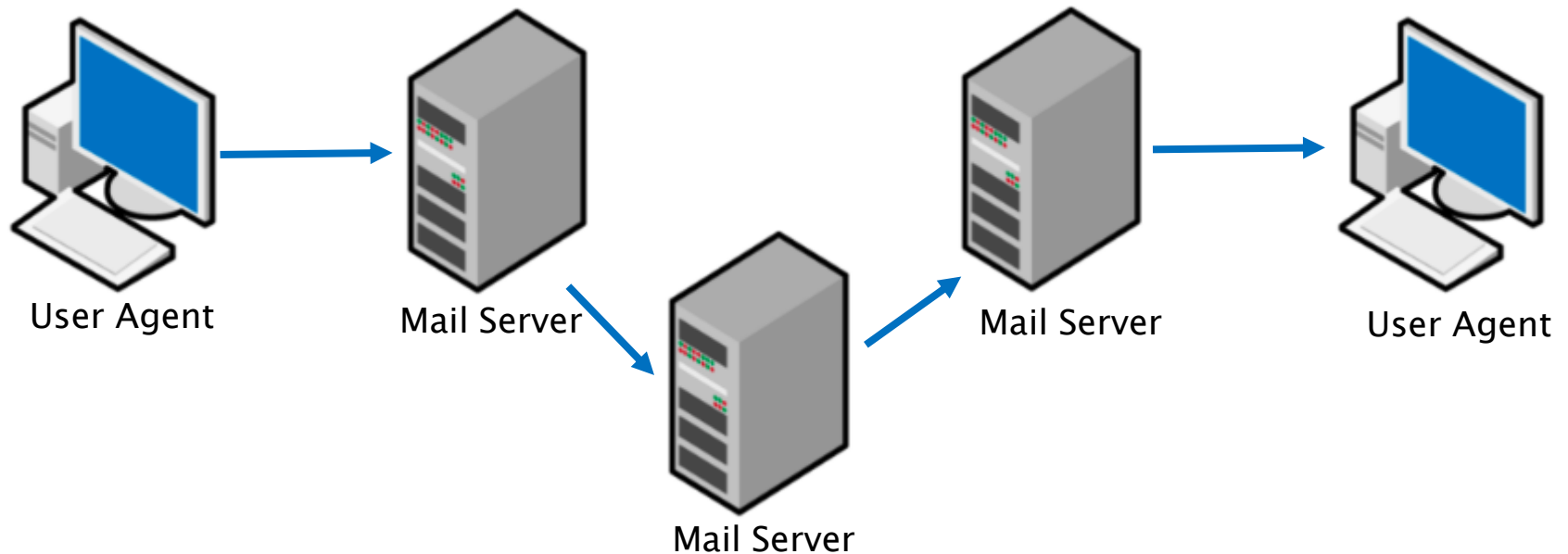
Email

Barak Gonen

Lesson Topics

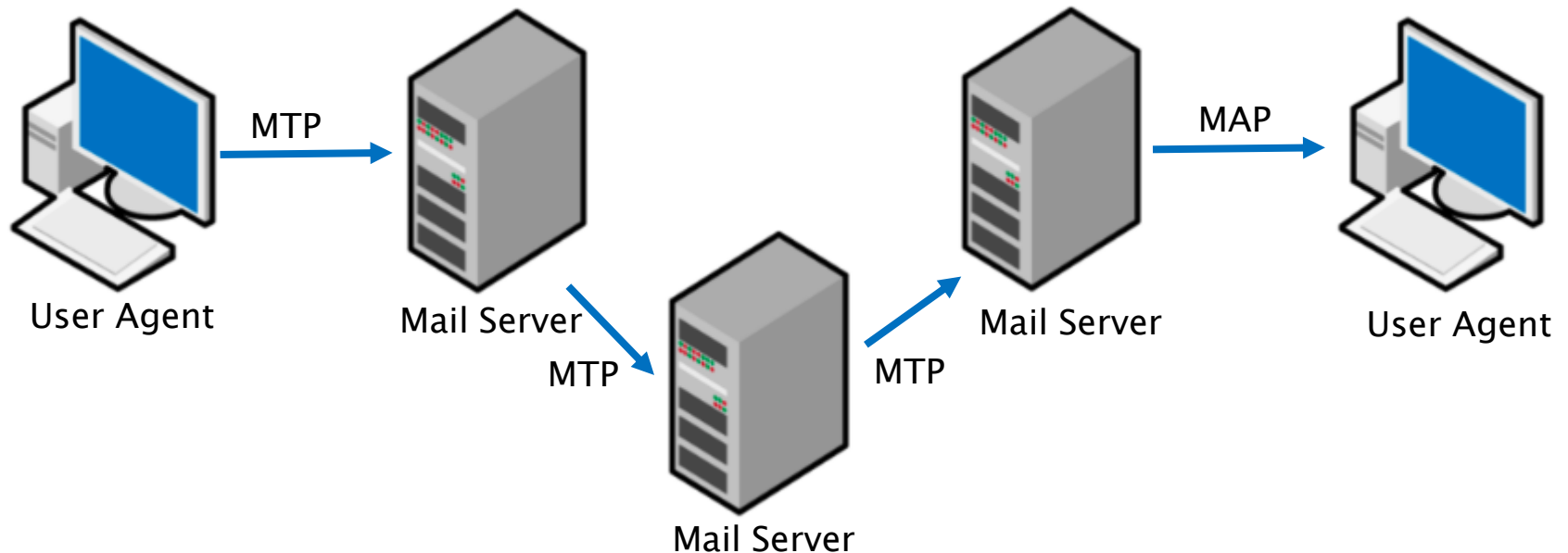
- ▶ Email architecture – sending, retrieving
 - SMTP
 - POP3
 - IMAP
 - HTTP/S
- ▶ Hands on with Wireshark
- ▶ SMTP programming

Email Architecture



- ▶ User agents (clients) – provide users with email functionality (read, reply, delete etc)
- ▶ Mail servers – store mail boxes, communicate with servers and agents

Email Architecture



- ▶ Mail transfer protocol: to send emails
 - SMTP – Simple Mail Transfer Protocol
- ▶ Mail access protocols: to retrieve emails
 - POP3
 - IMAP
 - HTTP/S

Access Protocols – POP3

- ▶ Post Office Protocol
- ▶ RFC 1939, May 1996
- ▶ Modes of operation
 - Download and delete
 - Retrieve messages and stores locally, server deletes
 - Download and keep
 - Same, but server keeps
 - Why is it troublesome? Consider office/home user

Access Protocols – IMAP

- ▶ Internet Message Access Protocol
- ▶ RFC 9051, Aug 2021
- ▶ Mail kept on the server
- ▶ Built to manage multiple devices for the same client
- ▶ Client can manage folders on the server
- ▶ Solves POP3 sync problem

About POP3

▶ Pros:

- Once downloaded, no need for Internet connection
- Saves storage on mail server

▶ Cons:

- Need for backup
- May be more vulnerable to viruses since mails are fully downloaded

About IMAP

▶ Pros:

- Emails stored on the server → adapted to multiple devices
- Thus, all is saved: sent, draft, and deleted mails or created folders
 - All devices are synchronized

▶ Cons:

- Requires internet connection

Access Protocols – HTTP/S

- ▶ Yahoo mail, Hotmail, Gmail, etc.
- ▶ IMAP special client no longer required
- ▶ User agent is browser
- ▶ HTTP/S
 - GET method– pull emails from server
 - POST method – push email to server

SMTP

- ▶ Simple Mail Transfer Protocol
- ▶ RFC 5321, Oct 2008
- ▶ From client to server and server to server
- ▶ Use the RFC to answer:
 - What is code 220?
 - Which code should be sent when requested mail action completed?

Base64

- ▶ Used to convert binary data to text
 - Some protocols, such as HTTP and SMTP, are textual

Encoding of the source string (Man) in Base64

Source	Character	M				a				n			
ASCII text	Octets	77 (0x4d)				97 (0x61)				110 (0x6e)			
Bits		0	1	0	0	1	1	0	1	0	1	1	0
Base64 encoded	Sextets	19				22				5			
	Character	T				W				F			
	Octets	84 (0x54)				87 (0x57)				70 (0x46)			

Base64 alphabet defined in RFC 4648.

Index	Binary	Char.	Index	Binary	Char.	Index	Binary	Char.	Index	Binary	Char.
0	000000	A	16	010000	Q	32	100000	g	48	110000	w
1	000001	B	17	010001	R	33	100001	h	49	110001	x
2	000010	C	18	010010	S	34	100010	i	50	110010	y
3	000011	D	19	010011	T	35	100011	j	51	110011	z
4	000100	E	20	010100	U	36	100100	k	52	110100	0
5	000101	F	21	010101	V	37	100101	l	53	110101	1
6	000110	G	22	010110	W	38	100110	m	54	110110	2
7	000111	H	23	010111	X	39	100111	n	55	110111	3
8	001000	I	24	011000	Y	40	101000	o	56	111000	4
9	001001	J	25	011001	Z	41	101001	p	57	111001	5
10	001010	K	26	011010	a	42	101010	q	58	111010	6
11	001011	L	27	011011	b	43	101011	r	59	111011	7
12	001100	M	28	011100	c	44	101100	s	60	111100	8
13	001101	N	29	011101	d	45	101101	t	61	111101	9
14	001110	O	30	011110	e	46	101110	u	62	111110	+
15	001111	P	31	011111	f	47	101111	v	63	111111	/
										Padding	=

Base64 Class Exercise

- ▶ Convert “Lev!” to base64
 - Note that padding is required if the number of encoded characters is not divided by 4

SMTP

- ▶ Simple Mail Transfer Protocol
- ▶ RFC 5321, Oct 2008
- ▶ From client to server and server to server
- ▶ Study hands on using wireshark
- ▶ Questions:
 - Client IP, Server IP
 - What does the client send on the first message? Server?
 - Extract username, password
 - What is the message?
 - How is disconnection done?

Lessons Learnt

- ▶ Client – server model operation
- ▶ Using socket module
- ▶ Programming client and server
- ▶ Basic communication protocol

