מודול 3 — יסודות רשתות

חומר לימוד

1 יחידות

תוכן העניינים

1. מודול 3 — יסודות רשתות (Networking Fundamentals)

מודול 3 — יסודות רשתות (Networking Fundamentals)

מטרות המודול: להבין לעומק כיצד מחשבים מתקשרים ברשת — מודל OSI ו-TCP/IP, TCP מול UDP ולחיצת היד, כתובות MAC, IP (IPv4/IPv6) ו-ARP, פורטים ופרוטוקולים, Subnetting, ופרוטוקולי התשתית (DNS, DHCP, NAT, ICMP). כל זה עם דוגמאות מעשיות ופקודות.

קבצים: `README.md` · `missions.md` · `solutions.md` · `practice.md` · `slides.md`.

מבוסס על שיעור המקור #07 ("Networking — 3 hours").

למה זה חשוב להאקר? כל תקיפה עוברת דרך הרשת. בלי להבין פורטים, פרוטוקולים, כתובות ולחיצת יד — אי אפשר להבין מה סורקים, מה מוצאים, וכיצד מנצלים. זה הבסיס למודולים 5–6 (סריקה וניצול).

תוכן העניינים

1. מהי רשת ומסע החבילה
2. מודל OSI
3. מודל TCP/IP
4. TCP מול UDP ולחיצת היד
5. כתובות IP
6. MAC ו-ARP
7. פורטים ופרוטוקולים
8. Subnetting
9. פרוטוקולי תשתית: DNS, DHCP, NAT
10. כלים מעשיים

3.1 מהי רשת ומסע החבילה

רשת היא אוסף מכשירים המתקשרים ביניהם. המידע נשלח ב**חבילות (Packets)** — יחידות קטנות של נתונים העוברות מהמקור ליעד, לעיתים דרך צמתים רבים (Routers).

מסע חבילה בקצרה — מה קורה כשאתה גולש ל- `example.com`

1. **DNS:** המחשב שואל את שרת ה-DNS "מה הכתובת של `example.com`?", ומקבל בחזרה כתובת `IP` — למשל `93.184.216.34`.
2. **ARP:** כדי לשלוח ברשת המקומית, המחשב מוצא את כתובת ה- `MAC` של ה-Gateway.
3. **TCP Handshake:** נפתח חיבור לפורט `80` או `443` (הרצף `SYN → SYN-ACK → ACK`).
4. **HTTP:** נשלחת בקשה `GET /`, והשרת מחזיר את הדף.

> 💡 כל שלב כאן שייך לשכבה אחרת במודל OSI — וכל שלב הוא גם **נקודת תקיפה אפשרית**: זיוף `DNS`, הרעלת `ARP`, סריקת פורטים, וחולשות Web.

3.2 מודל OSI

מודל **OSI** מתאר תקשורת בשבע שכבות. כל שכבה מדברת עם המקבילה בצד השני, ומוסיפה/מסירה "עטיפה" משלה.

#	שכבה	תפקיד	דוגמאות	תקיפה אופיינית
7	Application	ממשק לאפליקציה	HTTP, DNS, FTP, SSH	חולשות Web, פישינג
6	Presentation	הצפנה/קידוד	TLS/SSL	פענוח תעבורה
5	Session	ניהול חיבורים	RPC, NetBIOS	חטיפת Session
4	Transport	תעבורה מקצה-לקצה	TCP, UDP	סריקת פורטים
3	Network	ניתוב וכתובות לוגיות	IP, ICMP	Spoofing, ניתוב
2	Data Link	כתובות פיזיות	Ethernet, ARP, MAC	ARP Poisoning
1	Physical	אותות וכבלים	Wi-Fi, כבל	האזנה פיזית

> 💡 **טריק לזכירה (מלמעלה למטה):** *All People Seem To Need Data Processing*.

דוגמה להמחשה: כשאתה שולח מייל — שכבה 7 יוצרת את ההודעה, שכבה 4 מחלקת אותה לחבילות TCP, שכבה 3 מוסיפה כתובות IP, שכבה 2 מוסיפה כתובות MAC, ושכבה 1 שולחת את הביטים. בצד השני התהליך הפוך.

3.3 מודל TCP/IP

TCP/IP הוא הגרסה המעשית (4 שכבות) שרשת האינטרנט משתמשת בה בפועל. הוא "מקפל" את שכבות OSI:

TCP/IP	מקביל ל-OSI	דוגמאות
Application	`5–7`	HTTP, DNS, SSH
Transport	4	TCP, UDP
Internet	3	IP, ICMP
Network Access	`1–2`	Ethernet, ARP

3.4 TCP מול UDP ולחיצת היד

בשכבת התעבורה שני פרוטוקולים עיקריים:

	TCP	UDP
אמינות	אמין — מוודא הגעה	לא אמין — "יורה ושוכח"
חיבור	Connection-oriented	Connectionless
מנגנון	לחיצת יד תלת-שלבית	ללא
מהירות	איטי יותר	מהיר יותר
שימוש	web, מייל, קבצים	DNS, סטרימינג, VoIP

לחיצת היד התלת-שלבית (3-Way Handshake)

כך נפתח **כל** חיבור TCP — וזהו הבסיס לסריקת פורטים:

```
לקוח  --- SYN --->      (בקשת חיבור)       שרת
לקוח  <-- SYN-ACK --    (אישור + בקשה חוזרת)      שרת
לקוח  --- ACK --->      (החיבור פתוח)      שרת
```

🔍 **קשר ישיר לתקיפה — כך nmap קובע אם פורט פתוח:**

מסקנה	תשובת השרת ל-SYN	
מסונן (Filtered — חומת אש)	אין תשובה	
סגור	הפורט (Reset) `RST`	
פתוח	הפורט `SYN-ACK`	

סריקת "SYN Scan" (`nmap -sS`) שולחת SYN ולא משלימה את הלחיצה — מהירה ו"שקטה" יותר.

3.5 כתובות IP

כתובת IP מזהה מכשיר ברשת. שתי גרסאות: - **IPv4:** ארבעה מספרים `(0–255)`, למשל `192.168.1.10`. 32 סיביות, בפורמט עשרוני (Decimal). - **IPv6:** פורמט חדש, 128 סיביות, למשל `2001:0db8:85a3::8a2e:0370:7334`. בפורמט הקסדצימלי — נועד לפתור את מחסור הכתובות.

פרטי מול ציבורי

- **פרטי (Private)** — לשימוש פנימי, אינו מנותב באינטרנט:
  - `10.0.0.0 – 10.255.255.255` (/8)
  - `172.16.0.0 – 172.31.255.255` (/12)
  - `192.168.0.0 – 192.168.255.255` (/16)
- **ציבורי (Public)** — ייחודי וגלוי באינטרנט.
- **Loopback** — `127.0.0.1` = המכונה עצמה (`localhost`).

💡 כשתראה `192.168.x.x` או `10.x.x.x` בבדיקה — אתה ברשת פנימית. זה מכתיב את הצעד הבא (Pivoting — מודול 12).

3.6 MAC ו-ARP

MAC Address

כתובת פיזית קבועה של כרטיס הרשת, בשכבה 2.

- **פורמט:** `00:1A:2B:3C:4D:5E`
- "צרובה" בחומרה — אך **ניתנת לזיוף** (טכניקת `MAC Spoofing`).

ARP — Address Resolution Protocol

פרוטוקול שמתרגם כתובת `IP` לכתובת `MAC` ברשת המקומית.

איך זה עובד: כשמכונה רוצה לשלוח מידע לכתובת מקומית, היא "צועקת" ברשת — *"מי מחזיק את הכתובת הזו?"* — והבעלים עונה בחזרה עם כתובת ה-`MAC` שלו.

ARP Poisoning

מתקפת "איש בתווך" (**Man-in-the-Middle**) בשכבה 2.

התוקף עונה לבקשות `ARP` בשקר, וגורם לכל התעבורה לעבור דרכו — וכך הוא מאזין לתקשורת. זו מתקפה קלאסית ברשת מקומית.

פקודות לצפייה בטבלת ה-ARP:

```
ip neigh      # ידוע IP <-> MAC מיפוי
arp -a        # תחביר ישן
```

3.7 פורטים ופרוטוקולים נפוצים

פורט (Port) הוא "דלת" ממוספרת `(0–65535)` שדרכה שירות מקבל תקשורת. זיהוי הפורטים הפתוחים = זיהוי השירותים = **משטח התקיפה**.

פורט	פרוטוקול	שירות	הערה לתקיפה
21	TCP	FTP	העברת קבצים, לרוב לא מוצפן
22	TCP	SSH	גישה מרוחקת מאובטחת; יעד Brute Force
23	TCP	Telnet	גישה **לא** מוצפנת — מסוכן
25	TCP	SMTP	דואר יוצא
53	TCP/UDP	DNS	תרגום שמות
80	TCP	HTTP	web (לא מוצפן)
110	TCP	POP3	קבלת דואר
139/445	TCP	SMB	שיתוף קבצים Windows — יעד מרכזי
143	TCP	IMAP	קבלת דואר
443	TCP	HTTPS	web מוצפן (TLS)
3306	TCP	MySQL	בסיס נתונים
3389	TCP	RDP	שולחן עבודה מרוחק Windows

טווחי פורטים: **Dynamic** `(49152–65535)`, **Registered** `(1024–49151)`, **Well-Known** `(0–1023)`.

> 💡 שנן לפחות: 21, 22, 80, 443, 445, 3389 — הם חוזרים כמעט בכל בדיקה.

3.8 Subnetting (חלוקת רשת)

Subnetting הוא חלוקת רשת לתת-רשתות. עבור בודק חדירות, המטרה המעשית: **להבין את גבולות הרשת** — אילו כתובות שייכות לאותה תת-רשת (ולכן ייסרקו יחד).

מסכת רשת ו-CIDR

- **מסכת רשת (Subnet Mask)** מפרידה בין חלק הרשת לחלק המארח (Host).
- **CIDR** מקצר זאת: המספר אחרי `/` = כמה סיביות שייכות לרשת.

CIDR	מסכה	מס' כתובות	מארחים שמישים	דוגמה
/24	255.255.255.0	256	254	`192.168.1.0/24`
/16	255.255.0.0	65,536	65,534	`172.16.0.0/16`
/8	255.0.0.0	~16.7M	~16.7M	`10.0.0.0/8`

🔍 **דוגמה מעשית:** `nmap 192.168.1.0/24` = "סרוק את כל 256 הכתובות בתת-הרשת". `ipcalc` `192.168.1.0/24` יראה לך את הכתובת הראשונה, האחרונה וה-Broadcast.

3.9 פרוטוקולי תשתית

DNS — Domain Name System

מתרגם שם דומיין (למשל `example.com`) לכתובת `IP`.

- עובד בעיקר מעל `UDP`, פורט `53`.
- יעד מרכזי ל-OSINT ולתקיפות כמו `DNS Spoofing` ו-`Zone Transfer`.

DHCP — Dynamic Host Configuration Protocol

מחלק כתובות `IP` אוטומטית למכשירים חדשים ברשת — יחד עם כתובת ה-Gateway ושרת ה-DNS.

NAT — Network Address Translation

הראוטר "מתרגם" כתובות פרטיות רבות לכתובת ציבורית אחת. זו הסיבה שלכל המכשירים בבית יש כתובת `192.168.x.x`, אך כלפי האינטרנט מופיעה כתובת ציבורית אחת בלבד.

ICMP — Internet Control Message Protocol

פרוטוקול בקרה. הפקודה `ping` משתמשת בו כדי לבדוק אם מכונה "חיה".

🔍 **Zone Transfer** (הפקודה `AXFR`): אם שרת ה-DNS מוגדר שגוי, אפשר "לשאוב" ממנו את כל רשומות הדומיין — פרס ב-Recon:

```
dig axfr @nsserver target.com
```

3.10 כלים מעשיים

```
ip a                        # כתובות IP ו-MAC
ip route                    # ה-Gateway
ping -c 4 8.8.8.8           # קישוריות (ICMP)
traceroute google.com      # מסלול החבילה (צמתים)
ss -tuln                    # פורטים מאזינים מקומית
nslookup google.com        # תרגום DNS
dig google.com             # מפורטת DNS שאילתת
ipcalc 192.168.1.0/24      # חישוב תת-רשת
```

Wireshark ו-tcpdump — לראות את החבילות

- **Wireshark** — כלי גרפי ללכידת וניתוח תעבורה. מסננים שימושיים: `tcp.flags.syn==1`, `http`, `dns`.
- **tcpdump** — גרסת שורת הפקודה: `sudo tcpdump -i eth0 port 80`.

> 💡 **תרגיל מנטלי:** פתח Wireshark, גלוש לאתר, וזהה את רצף ה-SYN → SYN-ACK → ACK. לראות את הלחיצה "בחיים" מקבע את ההבנה.

סיכום המודול

- **מסע חבילה**: DNS → ARP → TCP Handshake → HTTP; כל שלב הוא שכבה ונקודת תקיפה.
- **OSI (7)** ו-**TCP/IP (4)** מתארים את התקשורת; שכבה 4 = פורטים, שכבה 3 = IP, שכבה 2 = MAC.
- **TCP** אמין (Handshake) · **UDP** מהיר; תשובת SYN קובעת אם פורט פתוח/סגור/מסונן.
- **IP** (פרטי/ציבורי), **MAC/ARP**, **פורטים** ו-**Subnetting/CIDR** מגדירים את משטח התקיפה.
- **DNS/DHCP/NAT/ICMP** הם התשתית — וגם יעדים (Spoofing, Zone Transfer).

מה הלאה?

➡️ **תרגילים** — `missions.md` · **פתרונות** — `solutions.md` · **תרגול ותרחישים** — `practice.md` ➡️

מודול 4 — סקריפטינג: Bash ו-Python

⬅️ חזרה למפת הקורס