**חוברת לימוד · רשתות תקשורת · שכבה יג תשפ"ז**

# מעבדה: כתיבת ACL ואימותן (על רשימות גישה)

שבוע 16 · 07/01–03/01 · **לופו משה**

שלוש דרישות ממנהל הרשת. כל אחת — רשימה אחת, מקום אחד, כיוון אחד. היום כותבים סטנדרטית (מספר ובשם), מורחבת (לפי פורט), נופלים בכוונה במלכודת של `permit` חסר, ומוכיחים שהכול עובד עם מוני ההתאמות.

## 1. לפני `PT` — שלוש שורות על הנייר

| הדרישה | סוג | איפה | כיוון |
|---|---|---|---|
| רשת 192.168.10.0/24 לא תגיע לשרת 192.168.30.0/24 (הכול אחר — כן) | סטנדרטית | ממשק מול 30.0. | `out` |
| מהמחלקה: `DNS` ו-`DHCP` כן, `HTTP` לא | מורחבת | ממשק מול המחלקה | `in` |
| רק `HTTPS` לשרת 10.0.1.5 מכולם | מורחבת | ממשק מול השרת | `out` |

**טיפ** · 'מקור בלבד?' → סטנדרטית → קרוב ליעד. 'פרוטוקול/פורט?' → מורחבת → קרוב למקור.

## 2. סטנדרטית — מספר ובשם

```
! numbered (5.1.8)
R1(config)# access-list 10 deny 192.168.10.0 0.0.0.255
R1(config)# access-list 10 permit any
R1(config)# interface g0/2
R1(config-if)# ip access-group 10 out

! named (5.1.9) — same thing, readable
R1(config)# ip access-list standard BLOCK_10
R1(config-std-nacl)# deny 192.168.10.0 0.0.0.255
R1(config-std-nacl)# permit any
R1(config)# interface g0/2
R1(config-if)# ip access-group BLOCK_10 out

R1# show access-lists
Standard IP access list BLOCK_10
    10 deny   192.168.10.0, wildcard bits 0.0.0.255 (8 matches)   <- proof
    20 permit any (42 matches)
```

`deny` ואז `permit any. out` על הממשק מול היעד. המונים מוכיחים שהחבילות עוברות בשורה הנכונה.

## 3. מורחבת — פרוטוקול ופורט

```
! 5.4.12: block FTP and ICMP from LAN to server, allow the rest
R1(config)# access-list 101 deny tcp 172.22.34.64 0.0.0.31 host 172.22.34.62 eq ftp
R1(config)# access-list 101 deny icmp 172.22.34.64 0.0.0.31 host 172.22.34.62
R1(config)# access-list 101 permit ip any any
R1(config)# interface g0/0
R1(config-if)# ip access-group 101 in           <- near the SOURCE

  test: ping server  -> fails (icmp denied)
        ftp server   -> fails
        http server  -> works  (permit ip any any)
```

`tcp/udp/icmp` · `host` · `eq` ‹פורט› · `permit ip any any` בסוף · `in` קרוב למקור.

## 4. המלכודת — ורק אז ההוכחה

```
R1(config)# access-list 20 deny 10.0.0.0 0.0.0.255     <- and nothing else
R1(config-if)# ip access-group 20 in
  -> EVERYTHING from that interface is blocked (implicit deny any)
R1(config)# access-list 20 permit any                  <- fixed

  order matters:
  access-list 30 permit any
  access-list 30 deny 10.0.0.0 0.0.0.255   <- never reached: permit any caught everything
```

בלי `permit` — הכול חסום. `permit` לפני `deny` — ה-`deny` מת. הסדר והסיום הם 90% מהטעויות.

# דוגמה פתורה

### השאלה

הדרישה: 'מהמחלקה 192.168.20.0/24 — לאפשר `DNS` ו-`DHCP`, לחסום `WEB`, לאפשר השאר'. כתבו ושייכו.

### הפתרון צעד אחר צעד

1. **סוג**: פרוטוקול ופורט → מורחבת → קרוב למקור → `in` על הממשק מול .20.0.
2. **`DNS`**: `permit udp 192.168.20.0 0.0.0.255` `.any eq 53`
3. **`DHCP`**: `permit udp 192.168.20.0 0.0.0.255` `.any eq 67`
4. **`WEB`**: `deny tcp 192.168.20.0 0.0.0.255` `any eq 80` (ו-443 אם רוצים).
5. **השאר**: `.permit ip any any`
6. **שיוך**: `interface g0/0/1` → `.ip access-group 101 in`
7. **אימות**: `nslookup` ✓ · `ipconfig` / `renew` ✓ · דפדפן ✗ · `ping` ✓ · `show access-lists` — המונים.

---

**תשובה** · בדיוק שאלה 3ה מבחינת 2023: '`access-list 101 deny ip` | `host` ...' — התבנית. `permit ip any any` בסוף, `in` על הממשק של המחלקה.

## תרגילים — פתרו לבד לפני שאתם בודקים בכיתה

1. שייכתם סטנדרטית `in` על הממשק מול המקור. מה נשבר?
2. `ping` עובר, `HTTP` לשרת לא. איזו רשימה?
3. `permit any` לפני `deny` — מה קורה?
4. `show access-lists` מראה 0 `matches` על `deny`. שני הסברים?
5. `access-list 101 deny icmp` 172.22.34.64 0.0.0.31 `host 172.22.34.62` — מה נחסם?
6. `wildcard 0.0.0.31` = איזה `prefix`?
7. רק `HTTPS` לשרת 10.0.1.5 מכולם, השאר חסום — שורה אחת מספיקה?

## דף עבודה — מעבדה 16 — `ACL`: סטנדרטית · בשם · מורחבת · המלכודת

שם: _____ תאריך: _____ חוליה: _____

---

**המטרה:** 3 רשימות מאומתות עם מונים, ומלכודת אחת שנפלתם בה בכוונה.

### מהלך — סמנו כל שלב שהושלם

☐ מלאו את טבלת 'דרישה → סוג → מקום → כיוון' לשלוש הדרישות — לפני `PT`.

☐ פתחו 5.1.8. סטנדרטית ממוספרת. `ping` לפני/אחרי. `show access-lists` — העתיקו עם המונים. `Completion`.

☐ פתחו 5.1.9. אותו דבר בשם. `Completion`.

☐ פתחו 5.4.12. מורחבת: `FTP` ו-`ICMP` חסומים, השאר לא. בדקו `ping/ftp/http`. `.show access-lists. Completion`

☐ המלכודת: `access-list 20 deny` ‹רשת› בלבד → שיוך → `ping` לכל מקום: _____. הוסיפו `permit any` → _____. הפכו את הסדר (`permit` קודם) → מה עם ה-`deny`? _____.

☐ (רשות) 5.5.1 `Challenge`.

## רישום התוצאות

| # | מה למלא | חישוב · הערות · ראיות (פלט פקודה) |
|---|---|---|
| 1 | טבלה: _____ | _____ |
| 2 | 5.1.8: `show access-lists` _____ · `Completion` _____ % | _____ |
| 3 | 5.1.9: `Completion` _____ % | _____ |
| 4 | 5.4.12: `ping` _____ `ftp` _____ `http` _____ · `show access-lists` _____ · `Completion` _____ % | _____ |
| 5 | מלכודת: `deny` בלבד → _____ · + `permit any` → _____ · `permit` קודם → `deny`: _____ | _____ |

## רפלקציה — שתי שורות

מה היה הכי קשה, ומה פתר את זה?

_____

מה הייתי עושה אחרת בפעם הבאה?

_____

**אישור המורה:** ☐ Completion במעבדה (150) ☐ התוצר (קובץ + תיעוד) (50) ☐ בונוס: ההרחבה (50) · **XP** _____ · תג: ☐ שומר/ת הסף · חתימה _____

## כרטיס יציאה — 3 שאלות, 5 דקות, נמסר ביציאה

1. סטנדרטית — איפה ובאיזה כיוון (בדרך כלל)?

---

2. מורחבת — איפה?

---

3. מה בסוף כל רשימה?

---

## עוד לבדיקה עצמית — ענו בעל פה, בלי להסתכל אחורה

1. ההוכחה שהרשימה עובדת?

## מילון מונחים

| עברית | אנגלית | הסבר קצר |
|---|---|---|
| **רשימה בשם** | `Named ACL` | `ip access-list standard/extended NAME` |
| **מספור שורות** | `Sequence Numbers` | 10, 20, 30 — לעריכה בשם |
| **eq** | `eq ‹port›` | שווה לפורט — ב-`ACL` מורחבת |
| **אתגר** | `Challenge` | 5.5.1 — בלי הנחיות |

## להעמקה

### בספר הקורס

- **ENSA 3.8.6** — DHCP · עמוד 133
- **ITN 15.4.7** — DHCP Operation · עמוד 528
- **ENSA 5.1** — Configure Standard IPv4 ACLs · עמוד 188

### קבצים שנעבוד איתם

- 5.1.8 Packet Tracer - Configure Numbered Standard IPv4 ACLs.pka
- 5.1.9 Packet Tracer - Configure Named Standard IPv4 ACLs.pka
- 5.4.12 Packet Tracer - Configure Extended IPv4 ACLs - Scenario 1.pka
- 5.4.12 Packet Tracer - Configure Extended IPv4 ACLs - Scenario 1.docx
- 5.5.1 Packet Tracer - IPv4 ACL Implementation Challenge.pka
- Access Control Lists.docx

---