ACL

Access Control Lists

Overview

- Access Control List או בתרגום לעברית: רשימת גישה, הוא פיצ'ר אבטחה פשוט וחשוב מאוד שחייב להכיר. הנתב (Router) אותו הכרנו היטב במהלך הקורס מנתב חבילות מידע לכל יעד ברחבי הרשת ללא הבחנה. ACL מאפשר למנהל הרשת לחסום או לאפשר ניתוב חבילות על נתבי הרשת, מנהל הרשת משיג שליטה מלאה על תעבורת המידע ברשת ויכול בקלות למנוע תקשורת בין רשתות, מחשבים ואפילו פרוטוקולים (לדוג' HTTP).
- הגנת הרשת בפני איומים תבוא לרוב מרכיבי Firewall, למרות הדמיון הרב בין ACL לחומת האש, ACL אינו תחליף ל-Firewall בשום פנים ואופן, אלא תוספת.
- לימודי ה-CCNA אינם מתמקדים בעולם אבטחת המידע אבל, חלק מאחריות מנהל הרשת היא להגן על הרשת בעזרת הכלים שנתנו לו. במהלך הקורס הכרנו כלי אבטחה בסיסיים וכיצד לאבטח היבטים שונים ברשת שלנו. (לדוג': אבטחת עדכוני ניתוב, Port Security ו-SSH) עלינו ליישם אותם עד כמה שניתן.

ACL Concept

- Access Control List היא רשימה פשוטה שכוללת משפטים משני סוגים:
  - **Permit**-משפט תנאי **שמאפשר** מעבר חבילות מידע.
  - **Deny**-משפט תנאי **שחוסם** מעבר חבילות מידע.
- משפטי התנאי מתבססים על מספר פרמטרים פשוטים:
  - **כתובות לוגיות** (IP) של רכיב המקור והיעד. לדוג': 10.1.0.5
  - **כתובות פורטים לוגים** של רכיב המקור והיעד. לדוג' 80, 23, 443.
- הנתב (Router) משווה את הנתונים של כל חבילה שמגיעה/יוצאת מהפורטים לתנאים ב-ACL ופועל על פיהם וכך מבצע את רצון מנהל הרשת. אפשר לתאר את פעולת הפיצ'ר במילה אחת: **פילטר**.

ACL Uses

- אופן הפעולה של ACL הוא פשוט מאוד וגמיש מאוד, לכן נוכל להשתמש בו במקרים שונים. שימושי ACL:
  - **מידור והוספת אבטחה בסיסית לרשת** - בעזרת ACL נוכל למדר חלקים שונים ברשת. לדוגמה: נוכל למנוע גישה ממחשב ספציפי לשרת קריטי, בעקבות חשש לדליפת מידע רגיש או פעולות זדוניות מצד המשתמש.
  - **חסכון במשאבים והגברת ביצועי הרשת** - בעזרת ACL נוכל לחסום מידע מיותר ברשת שאינו קשור למשימות הארגון. לדוגמה: לחסום תוכנת הורדת סרטים שגוזלת רוחב-פס ומשאבים נוספים (לדוג' ביטורנט ונטפליקס).
  - **חסימת שירותים לפי סוג** - ACL חוסם פורטים לוגיים, מה שאומר גם פרוטוקולים. בעזרת ACL נוכל להגביל את הגישה לשירותי מייל (IMAP, POP3, SMTP) או שירותי אחסון קבצים (FTP) ברחבי הרשת.

ACL Uses

- **חסימת פרוטוקולי ניתוב ופרוטוקולי אבטחה** - יכולת נוספת של ACL היא חסימת עדכוני פרוטוקולי ניתוב כמו: RIP, OSPF, EIGRP ועוד.. וחסימה של חבילות פרוטוקולי אבטחה כמו: AHP ו-ESP ופרוטוקולים נוספים כמו GRE ו-ICMP. בצורה כזו יש למנהל הרשת שליטה מלאה על זרימת המידע וסוג המידע ברחבי הרשת.

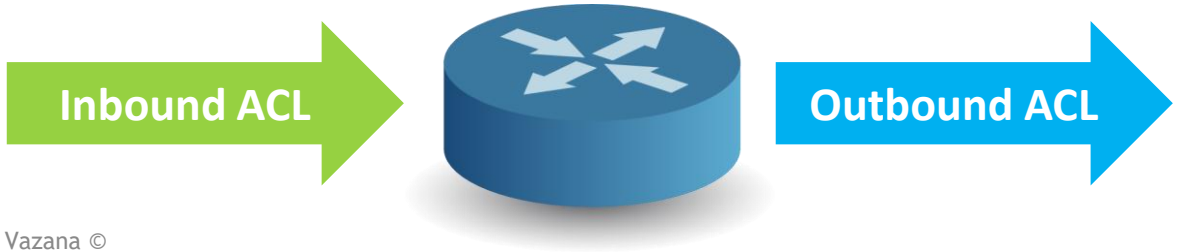
- ACL משמש לשירותים נוספים, עליהם נרחיב בהמשך הקורס:
  - Network Address Translation (NAT)
  - Quality of Service (QoS)

ACLs Types

- קיימים שני סוגים של ACL ההבדל בינהם בא לידי ביטוי בפרמטרים אותם הם בודקים:
  - **Standard ACLs** - סוג זה של ACL חוסם או מאפשר חבילות מידע על פי פרמטר אחד בלבד: כתובת המקור הלוגית (IP). זאת אומרת שלא משנה לנתב מהו היעד של החבילה, עצם העבודה שהיא הגיעה מרשת או מחשב ספציפי אסורים, החבלה נחסמת.
    - היתרון-לא גוזל משאבים רבים מהנתב בעת בדיקת חבילות.
    - חיסרון-לא מאפשר גמישות בתכנון אבטחה ברשת.
  - **Extended ACLs** - סוג זה של ACL הוא יותר מתקדם וגם מורכב. Extended ACL יכול לחסום או לאפשר חבילות מידע על פי מספר פרמטרים:
    - כתובות לוגיות (IP) של המקור והיעד.
    - כתובות מספרי פורטים של המקור והיעד.
    - לפי פרוטוקול.
    - היתרון-מאפשר גמישות רבה בתכנון אבטחת הרשת.
    - חיסרון-גוזל משאבים רבים מהנתב בעת בדיקת החבילות.

ACL Operation

- הגדרת ACL אינה מסתיימת ביצירת הרשימה. לנתב (Router) בדרך כלל יש מספר פורטים פיזיים (Interface) דרכם מגיעות ויוצאות חבילות מכל רחבי הרשת, אנו חייבים למקד את פעולת הסינון של ה-ACL **לפורט פיזי ספציפי בכיוון ספציפי**.
- כיוון ספציפי? מנקודת המבט של מנהל הרשת מקור ויעד החבילות הן דבר ברור. אנו רוצים שגם הנתב יכיר מנק' המבט שלו מהי רשת המקור ורשת היעד שמופיעות ב-ACL. לכן אנו חייבים לבחור באיזה כיוון הנתב יאכוף את משפטי התנאי:
  - **Inbound** - הגדרת ה-ACL בכיוון IN (פנימה) אומרת, שהנתב יבדוק את כל החבילות **שנכנסות** אל הפורט. לדוג' Interface Gig 0/1
  - **Outbound** - הגדרת ה-ACL בכיוון OUT (החוצה) אומרת, שהנתב יבדוק את כל החבילות **שיוצאות** מהפורט.
- **תמיד נזכור! שלב ראשון:** יצירת ה-ACL **שלב שני**: שיוך ה-ACL לפורט המתאים בכיוון המתאים.



Implicit Deny

- Implicit Deny הוא תנאי קבוע שנמצא בסופו של כל ACL! צמד מילים אלו אומר **חסימה מוחלטת!**
- סדר המשפטים ב-ACL הוא מאוד חשוב ויש לתכנן אותו בקפידה. הנתב בודק כל חבילת מידע מול כל המשפטים (ACEs) **בסדר כרונולוגי** עד להתאמה, במידה ולא נמצאה התאמה לאף משפט, הנתב מגיע למשפט התנאי **Implicit Deny** וחוסם את החבילה.
- חוק! בכל ACL חייב להיות לפחות משפט Permit אחד!

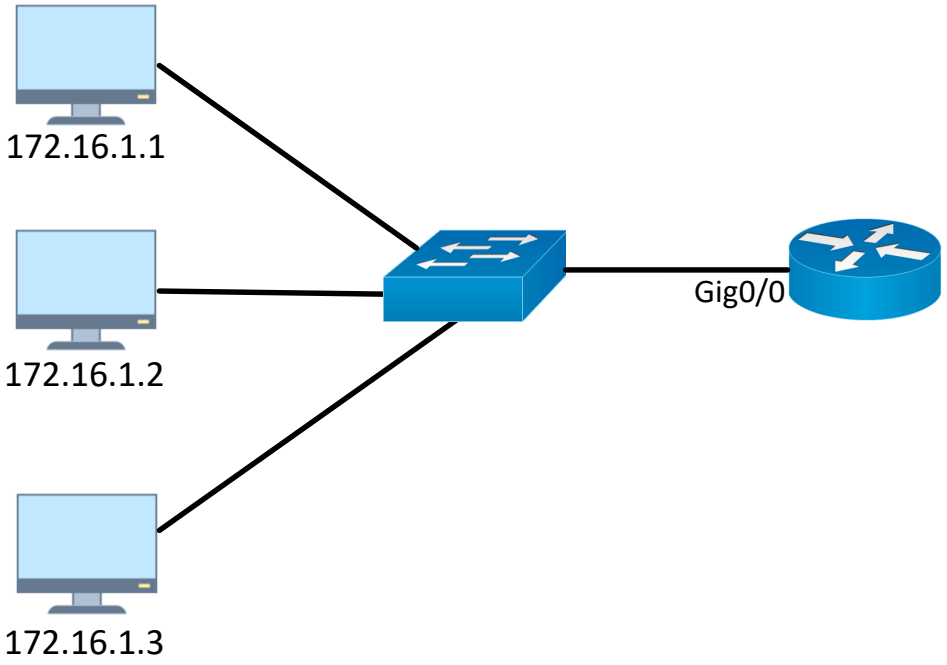
- בדוגמה משמאל נוכל לראות את ACL מספר 1:
- 2 משפטי התנאי הראשונים **חוסמים** כתובות מחשבים ספציפיים.
- משפט התנאי השלישי **מאשר** מעבר לכל החבילות שלא התאימו לשני משפטי התנאי הראשונים.
- משפט ה-Implicit deny **חוסם** את כל החבילות שלא התאימו לאף תנאי (נוצר אוטומטית).

Standard Access list 1

Deny host 192.168.1.1
Deny host 192.168.1.3
Permit any
Implicit Deny

Implicit Deny

- דוגמה למשפט התנאי בטופולוגיה פשוטה:
  - בדוגמה נוכל לראות שהוגדר ACL מסוג סטנדרט על פורט Gig 0/0 של הנתב בכיוון IN.



Standard access list 1

Deny host 172.16.1.2
Permit any
Implicit Deny

Wild Card Address

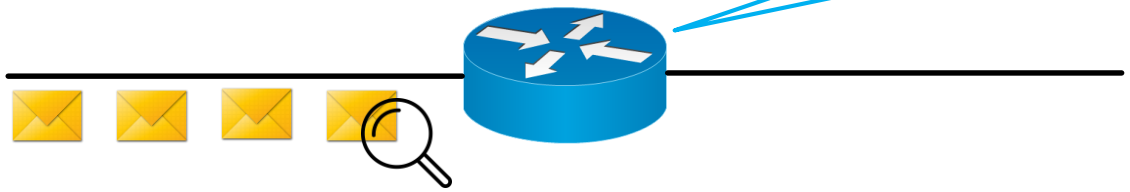
- כתובת Wild Card כבר ראינו בעבר, כשהגדרנו OSPF. כתובת זו מוצגת בצורה הפוכה לגמרי מכתובת ה-Subnet Mask, זאת אומרת שביטים ששווים אפס (0) מייצגים את חלק הרשת וביטים ששווים אחד (1) מייצגים את חלק המשתמש. מפה מגיעה המשמעות Wild "פרוע" כי מנוגד לחוקים.
- מטרת כתובת זו היא לשמש לנתב **מנגנון התאמה** בעזרת כתובת זו נוכל להגדיר לנתב בדיוק אילו כתובות לחפש ברמת ה-"אוקטטה" ואפילו ברמת ה-"ביט" הבודד. כתובת שהופך את ACL לכלי יעיל ומדויק מאוד.
- דוגמה לכתובות ה-Wildcard הנפוצות:
  - **0.255.255.255** הנתב יחפש התאמה **לאוקטטה הראשונה** של כתובת ה-IP בלבד! ושאר האוקטטות יכולות להיות כל דבר.
  - **0.0.255.255** הנתב יחפש התאמה **לשני האוקטטות** הראשונות של כתובת ה-IP.
  - **0.0.0.255** הנתב יחפש התאמה **לשלושת האוקטטות** הראשונות של כתובת ה-IP.
  - **0.0.0.0** הנתב יחפש **התאמה מדוייקת** לכתובת ה-IP.

Wildcard Match!

- דוגמה לאופן הפעולה של נתב יחד עם Wildcard מסויים.
- משפט ה-ACL:

```
Access-list 1 deny 192.168.0.0 0.0.255.255
```

"ממם...מחפש חבילות מידע שכתובת המקור שלהן מתחילה ב-**192.168**"



Standard ACL Syntax

- מבנה הפקודה של Standard ACL הינו פשוט:

R1(config)# access-list *number action source-address*

- *number* - עלינו להגדיר מספר ששמור לסוג זה, כדי שהנתב יבין שמדובר ב-Standard ACL טווח המספרים הוא **1-99**
- *action* -במקום action נגדיר את מטרת ה-ACL חסימה: deny או אישור: permit
- *source-address* - כתובת המקור הלוגית, במקום פרמטר זה נוכל להגדיר שלוש אפשריות:
  - כתובת רשת וכתובת wildcard מתאימה.
  - כתובת IP ספציפית של רכיב ברשת. נשתמש בפקודה: **host**
  - כולם. נשתמש בפקודה: **any**

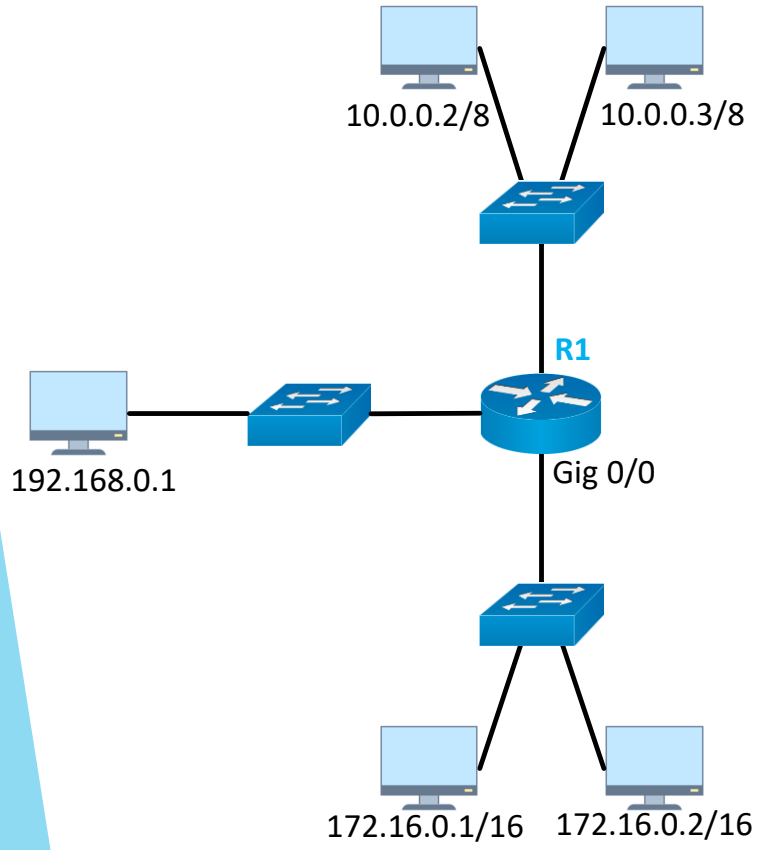
Extended ACL Syntax

- מבנה הפקודה של Extended ACL הוא מעט מורכב, כי הוא כולל מספר רב של פרמטרים:
  - בדוגמה נראה רק את הפרמטרים שחובה עלינו להגדיר בעת יצירת ACL מסוג זה, נוכל להוסיף פרמטרים כמו **פורטים לוגים** במידת הצורך.

R1(config)# access-list *number action protocol source-address destination-address*

- *number* - עלינו להגדיר מספר ששמור לסוג זה, טווח המספרים הוא **100-199**
- *action* -במקום action נגדיר את מטרת ה-ACL חסימה: deny או אישור: permit
- *protocol* - באיזה פרוטוקול יתמקד ה-ACL, במקום פרמטר זה נוכל להגדיר 3 אופציות עיקריות:
  - ip - פקודה זו מתכוונת לכל התעבורה ללא כל הבחנה בין סוגי המידע.
  - tcp - פקודה זו מתמקדת בפרוטוקולי TCP בלבד כמו HTTP.
  - udp - פקודה זו מתמקדת בפרוטוקולי UDP בלבד כמו DNS.
- *source-address* & *destination-address* - כתובות המקור והיעד הלוגיות, במקום פרמטר זה נוכל להגדיר שלוש אפשריות:
  - **כתובת רשת** וכתובת wildcard מתאימה.
  - כתובת IP ספציפית של רכיב ברשת. נשתמש בפקודה: **host**
  - כולם. נשתמש בפקודה: **any**

Standard ACL Example



- דוגמה ליצירת והגדרת ACL מסוג Standard:
- דרישה: מניעת קישוריות בין רשת 10.0.0.3/8 לרשת 172.16.0.0/16

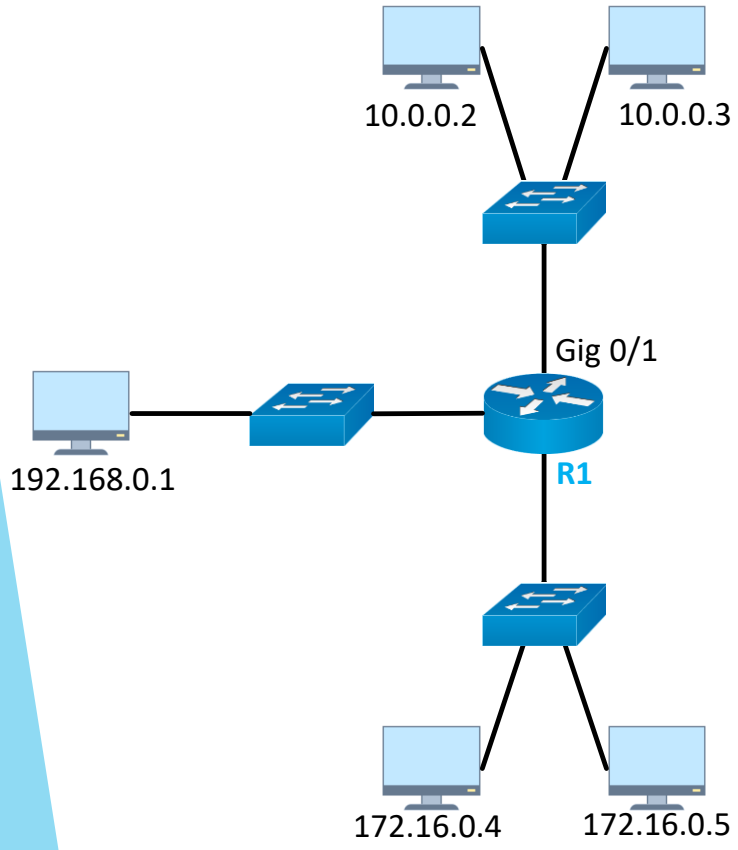
  - טווח המספרים **1-99** שמור ליצירת Standard ACLs.

```
R1(config)#access-list 1 deny 10.0.0.0 0.255.255.255
R1(config)#access-list 1 permit any
```

  - כלל זהב: הצבת Standard ACL לרוב יהיה בפורט הכי **קרוב אל היעד**.

```
R1(config)#interface gigabitEthernet 0/0
R1(config-if)#ip access-group 1 out
```

Extended ACL Example



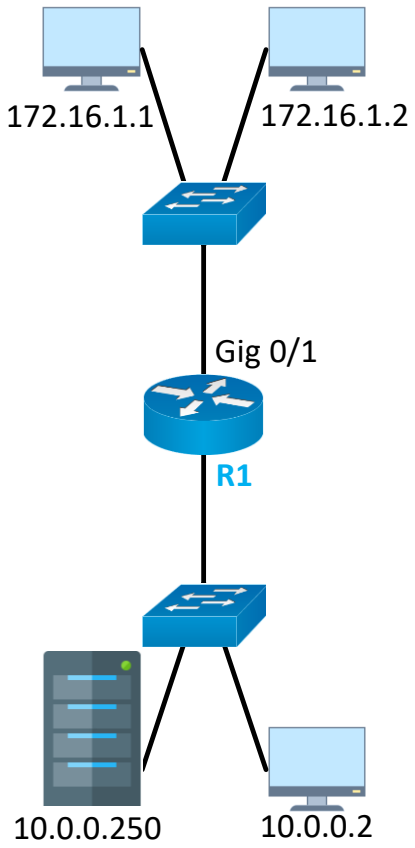
- דוגמה ליצירת והגדרת Extended ACLs:
- דרישה: מניעת קישוריות בין מחשב 10.0.0.2 למחשב 172.16.0.5
  - טווח המספרים **100-199** שמור ליצירת Extended ACLs.

```
R1(config)#access-list 100 deny ip host 10.0.0.2 host 172.16.0.5
R1(config)#access-list 100 permit ip any any
```

  - כלל זהב: הצבת Extended ACL לרוב יהיה בפורט הכי **קרוב אל המקור**.

```
R1(config)#interface gigabitEthernet 0/1
R1(config-if)#ip access-group 100 in
```

Extended ACL Example 2



- דוגמה ליצירת והגדרת Extended ACLs:
- דרישה: חסימת **שירות העברת קבצים** (FTP-21) בין **רשת** 172.16.1.0/24 **לשרת** 10.0.0.250
  - טווח המספרים **100-199** שמור ליצירת Extended ACLs.

```
R1(config)#access-list 100 deny tcp 172.16.1.0 0.0.0.255 host 10.0.0.250 eq 21
R1(config)#access-list 100 permit ip any any
```

  - כלל זהב: הצבת Extended ACL לרוב יהיה בפורט הכי **קרוב אל המקור**.

```
R1(config)#interface gigabitEthernet 0/1
R1(config-if)#ip access-group 100 in
```

Command Page

- רשימת הפקודות המלאה והסבר, נמצאת בקובץ Command Page ACL.