Text

Description automatically generated

**סמל מוסד - 270538**

**פרויקט גמר**

**הנדסאי מערכות תקשוב**

**התמחות: מערכות תקשוב**

**הנושא: הקמה, תכנון ופיתוח לחברת BlueSpace**

**מנחת הפרויקט : סיון שי**

**המגיש:**

שם: סוקולוב אדוארד ת.ז. 213761711

תאריך ההגשה: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

[מבוא](#_heading=h.30j0zll) **4**

[אודות החברה](#_heading=h.1fob9te) **5**

[תרשים ארגוני](#_heading=h.3znysh7) **6**

[מבנה החברה](#_heading=h.2et92p0) **7**

[מדיניות הקצאת שמות לרכיבי הרשת](#_heading=h.tyjcwt) **9**

[טבלת כתובות של סניפים](#_heading=h.3dy6vkm) **12**

[הגדרת שמות וMTOD לרכיבים](#_heading=h.1t3h5sf) **13**

[הגדרת שמות לרכיבים בסניף הראשון:](#_heading=h.4d34og8) 13

[הגדרת שמות לרכיבים בסניף השני:](#_heading=h.2s8eyo1) 14

[הגדרת שמות לרכיבים בסניף השלישי:](#_heading=h.17dp8vu) 15

[VTP](#_heading=h.3rdcrjn) **17**

[VTP Domain](#_heading=h.26in1rg) 17

[VTP Modes](#_heading=h.lnxbz9) 17

[VLAN’s](#_heading=h.35nkun2) **20**

[כיצד פועלים רשתות VLAN](#_heading=h.1ksv4uv) 20

[הגדרת vlan’s](#_heading=h.44sinio) **20**

[Trunk](#_heading=h.2jxsxqh) **27**

[HSRP](#_heading=h.z337ya) **29**

[מהו HSRP](#_heading=h.3j2qqm3) 29

[כיצד מורכבת פקטת HSRP](#_heading=h.1y810tw) 29

[הגדרת HSRP](#_heading=h.4i7ojhp) **31**

[STP](#_heading=h.2xcytpi) **33**

[בעיות שהSTP פותר](#_heading=h.1ci93xb) 33

[Broadcast storm](#_heading=h.3whwml4) 33

[STP port states](#_heading=h.2bn6wsx) 34

[Root Bridge](#_heading=h.qsh70q) 35

[Root Port](#_heading=h.3as4poj) 35

[Designated Port](#_heading=h.1pxezwc) 35

[Alternative Port](#_heading=h.49x2ik5) 35

[Path Cost](#_heading=h.2p2csry) 35

[Bridge Protocol Data Units (BPDU)](#_heading=h.147n2zr) 36

[סוגים נוספים של STP](#_heading=h.3o7alnk) 36

[RSTP](#_heading=h.23ckvvd) 36

[PVST](#_heading=h.ihv636) 36

[Rapid-PVST](#_heading=h.32hioqz) 36

[הגדרות בסיסיות ואבטחתיות של STP](#_heading=h.1hmsyys) 36

[Portfast](#_heading=h.41mghml) 36

[BPDU Guard](#_heading=h.2grqrue) 36

[Root Guard](#_heading=h.vx1227) 36

[Router On a Stick (Dot1Q)](#_heading=h.3fwokq0) **41**

[Sub-interfaces](#_heading=h.1v1yuxt) 41

[**DHCP**](#_heading=h.4f1mdlm) **42**

[יתרונות הDHCP](#_heading=h.2u6wntf) 43

[תצורת כתובת IP אמינה](#_heading=h.19c6y18) 43

[ניידות](#_heading=h.3tbugp1) 43

[שרת DHCP](#_heading=h.28h4qwu) 43

[לקוח DHCP](#_heading=h.nmf14n) 43

[מאגר כתובות IP](#_heading=h.37m2jsg) 43

[Subnet](#_heading=h.1mrcu09) 43

[תהליך הקצאת כתובות IP](#_heading=h.46r0co2) 44

[**DHCP Snooping**](#_heading=h.2lwamvv) **44**

[הגדרת DHCP](#_heading=h.111kx3o) 44

[Port-Security](#_heading=h.3l18frh) **48**

[איך זה עובד?](#_heading=h.206ipza) 50

[הגדרת port security](#_heading=h.4k668n3) 50

[IP Routing](#_heading=h.2zbgiuw) **52**

[EIGRP](#_heading=h.1egqt2p) **53**

[Distance vector vs Link State](#_heading=h.3ygebqi) 53

[יחסי שכנות](#_heading=h.2kx4qcwq1u0c) 54

[Feasible Distance vs Advertised Distance](#_heading=h.sqyw64) 55

[Variance command](#_heading=h.pz9ad5va0k05) 55

[ערכי K](#_heading=h.vcrf29ky9u4d) 55

[**OSPF**](#_heading=h.o0cfpxrp4qrx) **60**

[יחסי שכנות](#_heading=h.x78fgrjvxj4l) 60

[**ביבליוגרפיה**](#_heading=h.g6fmqq6xthmt) **66**

[איתור תקלות ופתרונן](#_heading=h.4bvk7pj) **66**

# מבוא

קוראים לי אדוארד סוקולוב, אני בן 19 וגר בחיפה.

למדתי המדעי המחשב 10 יחל בתיכון חוגים,חיפה במשך 3 שנים וכרגע אני לומד להנדסאי תקשוב במרום תקשוב במכללת כנרת.

הגעתי ללימודים האלה על מנת לצבור כלים חדשים בתחום המחשוב לקריירה העתידית שלי ובעיקר לפרויקטים האישיים שלי שיעזרו לי לממש את הרעיונות שלי בין אם זה בתוכנה או בחומרה.

השאיפה שלי היא לצאת לאזרחות עם כלים שיעזרו לי לנווט את הסביבה שלי לניצול מקסימאלי של הטכנולוגיה למען איכות חיים טובה יותר, בין אם זה משהו קטן שיעשה את חיים נוחים יותר או משהו הרבה יותר מזה.

נושאים שלמדתי בשנת יג + יד:

א. בסיסי נתונים: שליפת נתונים ויצירת טבלאות על ידי פקודות.

ב. רשתות תקשורת: רשתות LAN - כתובות IP וsubnet, פרוטוקול Ethernet, ,VLAN

TRUNK, VTP ,DOT1Q

.OSPF, EIGRP,RIP,STATIC ROUTE :ניתוב פרוטוקולי – WAN רשתות

.ועודDNS, DHCP,STP,NAT :ושרתים פרוטוקולים

ג.JS, CSS, HTML - בניית אתרים, ותכנות שלהם.

ד. שפת תכנות כמו: c,c# וpython

Linux: DNS, MAIL,FTP,SQL,WEB ב שרתים הקמת. ה

ו. חשמל ואלקטרוניקה. הכרה של רכיבים חשמליים ואת פעולותיהם.

ז.מערכות ספרתיות. הכרות עם שערים לוגים,אוגרים, מקודדים ועוד.

ח.HACKERU : למידה על מתקפות שונות ואיך למנוע.

ט. אבטחת מידע: למידה על כל מיני סוגי התקפות, דרכים להגן מפניהם, סוגי נוזקות ועוד.

# אודות החברה

BlueSpace היא חברה המתעסקת בפיתוח, תחזוקה, ותמיכה טכנית למוצרים שלה אשר מתרכזות בכל הקשור למערכות ניהול עסק מתקדם מרחוק.

לחברה יש שלושה סניפים ברחבי הארץ כאשר כל אחד מהם מתמקד בחלק שבו הוא הכי מתמחה. הסניף העיקרי של החברה נמצא בתל אביב, ושם נמצא הדירקטוריון של החברה.

המוצרים שהחברה מציאה ללקוחות שלה הן פתרונות אוטומציה ובקרה לבתי עסק גדולים וקטנים המאפשרים לבעלי העסק יותר שליטה על העסק, ובעלויות קטנות יותר ממה שיציע מנהל אנושי.

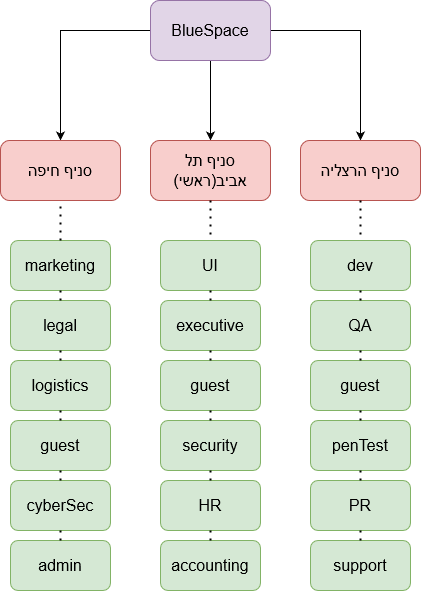
בסופו של היום כל עלות נוספת למעסיק עובר ללקוחות, וכאשר למדינה יש בעיה של יוקר מחייה, לא יזיק לקצץ בעלויות בכל מקום שאפשר, ובמיוחד במשרה שגם ככה מקבלת שכר גבוה בהרבה מהממוצע.

כרגע החברה נמצאת בשלב של סטארט-אפ בה היא מנסה לגייס הון דרך השקעות וחובות. על מנת למתן סיכונים, דירקטוריון החברה הוציא הנחייה לא להוציא כספים על מוצרים מיותרים שאינם תורמים לתפעול החברה, וכתוצאה מכך נדרש מהמנהלי רשת לתכנן רשת אופטימלית ככל הניתן מבלי לשבש את פעילותה ואבטחתה התקינה (יש לקחת בחשבון שיתירות מינימלית נדרשת על מנת להבטיח את פעילותה התקינה של הרשת).

כאשר החברה תהיה רווחית היא תדרוש יותר משאבים מהרשת. על מנת לצמצם עלויות נוספות חשוב למנהלי רשת לקחת בחשבון גדילה עתידית של הרשת ולתכנן אותה בהתאם.

ניתן גם להוסיף שככל שהחברה תגדל בשווי והרווחיות שלה היא תידרש להתמודד עם איומים חדשים ומתוחכמים יותר, לכן חשוב לתת לה בסיס אבטחתי טוב כך שלפחות איומים בסיסים יותר יחסמו ישר. המטרה של מנהלי הרשת בחברה היא לתקן את כל פרצות האבטחה הידועות בפרוטוקולי ורכיבי התקשורת הקיימים ברשת הארגונית.

# תרשים ארגוני



# מבנה החברה

חברת Bluespace מתפרסת לשלושה סניפים ברחבי הארץ: סניף ראשי בתל אביב , ושני סניפי משנה

בחיפה והרצליה. המחלקות השונות ותפקידן :

|  |  |  |
| --- | --- | --- |
| שם המחלקה | קיצור | תפקיד |
| Admin | Adm | תחזוק מערכות מחשוב ותקשורת אשר מתעסקת בתחזוק כל הרשת הארגונית. זקוקה לגישה מלאה לרשת הארגונית |
| Executive | Exc | מחלקה למנהלי החברה מתעסקת בניהול כל החברה והיא כוללת מסמכים רגישים מאוד, כך שנדרשת אבטחה גבוהה מן הרגיל |
| Guest | Gst | מחלקה לאורחים ופנאי אשר ממוקמת בכל סניף. מחלקה זו פתוחה באופן מוחלט וחייבת בידוד מלא מהרשת הארגונית |
| Security | Sec | מחלקה למאבטחי הבניין. מחלקה זו מתעסקת בקישור מצלמות האבטחה למסכים של המאבטחים |
| HR | Hr | מחלקת משאבי אנוש. מחלקה המתעסקת במשאבי אנוש ומחייבת קישור למסד נתונים בו יש מידע על כל העובדים. מחלקה זו זקוקה לאבטחת מידע מוגברת |
| Accounting | Act | מחלקת חשבונאות לניהול תקציבי החברה. מחלקה זו זקוקה לקישור למסד נתונים בו נשמר המידע הפיננסי, והיא זקוקה לאבטחה מוגברת |
| dev | Dev | מחלקת פיתוח מוצרים. מחלקה זו מצריכה גישה לאינטרנט על מנת שהעובדים יוכלו להיעזר בו, אך היא גם זקוקה לאבטחה מוגברת כיוון שהsource code נשמר בו |
| QA | Qa | מחלקת בדיקת איכות. מחלקה זו אחראית לבדיקת איכות המוצר |
| penTest | Pnt | רשת נפרדת לבדיקת חדירויות. רשת זו מחייבת בידוד מוחלט מהרשת האירגונית כיוון שנעשים בו בדיקות חדירויות אשר עשויות לפגוע במכשירים. |
| PR | Pr | מחלקת יחסי ציבור. מחלקה זו מתעסקת ביחסי ציבור והיא זקוקה לחיבור לאינטרנט |
| support | Sup | מחלקת שירות לקוחות ותמיכה טכנית. מחלקה זו אחראית על מתן עזרה ללקוחות שלה והיא מצריכה חיבור לאינטרנט, קישור למסד נתונים השומר נתוני לקוחות, והבטחה מוגברת על מנת שהמידע לא יזלוג החוצה |
| Marketing | Mkt | מחלקת שיווק. במחלקה זו נדרש חיבור לאינטרנט |
| Legal | Lgl | מחלקת משפטים. זקוקה לחיבור לאינטרנט |
| Logistics | Lgs | מחלקת לוגיסטיקה. מחייבת קישור למסד נתונים המתעסק בלוגיסטיקה |
| cyberSec | Cys | מחלקת אבטחת מידע. מחלקה שמחפשת ניסיונות פריצה בזמן אמת.זקוקה לגישה מלאה לרשת הארגונית. |
|  |  |  |

# מדיניות הקצאת שמות לרכיבי הרשת

**סניף ראשון – הרצליה (HZL)**

|  |  |  |  |
| --- | --- | --- | --- |
| **הסבר קיצור** | **קיצור** | **מיקום**  **בטופולוגיה** | **שם הרכיב** |
| RT = Router  C = Core | RT01-C-HZL RT02-C- HZL | Core | Router 1941 |
| SE = Switch Empty  D = Distribution | SE1-D- HZL  SE2-D- HZL | Distribution | Switch-PT-Empty |
| SW = Switch A = Access | SW01-A- HZL  SW02-A- HZL | Access | Switch 2960 |
| <n> = Number in the  department  <d> = Department name | <d>-HZL-<n> | - | PC |
| <s> = server type  <n> = number | HZL-<s>-<n> | - | server |

**דוגמאות לשמות התקני הקצה בסניף הרצליה (HZL)**

|  |  |  |
| --- | --- | --- |
| **שם ההתקן** | **קיצור מחלקה** | **מחלקה** |
| dev-HZL-01  dev-HZL-02 | dev | Development |
| PR-HZL-05 | PR | PR |
| QA-HZL-05 | QA | QA |

**סניף שני – תל אביב (TLV)**

|  |  |  |  |
| --- | --- | --- | --- |
| **הסבר קיצור** | **קיצור** | **מיקום**  **בטופולוגיה** | **שם הרכיב** |
| RT = Router  C = Core | RT01-C- TLV RT02-C- TLV | Core | Router 1941 |
| SE = Switch Empty  D = Distribution | SE1-D- TLV  SE2-D- TLV | Distribution | Switch-PT-Empty |
| SW = Switch A = Access | SW01-A- TLV  SW02-A- TLV | Access | Switch 2960 |
| <n> = Number in the  department  <d> = Department name | <d>- TLV -<n> | - | PC |
| <s> = server type  <n> = number | TLV-<s>-<n> | - | server |

**דוגמאות לשמות התקני הקצה בסניף תל אביב (TLV)**

|  |  |  |
| --- | --- | --- |
| **שם ההתקן** | **קיצור מחלקה** | **מחלקה** |
| UI-TLV-02  UI-TLV-03 | UI | UI |
| exec-TLV-02 | exec | Executive |

**סניף שלישי – חיפה (HFA)**

|  |  |  |  |
| --- | --- | --- | --- |
| **הסבר קיצור** | **קיצור** | **מיקום**  **בטופולוגיה** | **שם הרכיב** |
| RT = Router  C = Core | RT01-C-HZL RT02-C- HFA | Core | Router 1941 |
| SE = Switch Empty  D = Distribution | SE1-D- HFA  SE2-D- HFA | Distribution | Switch-PT-Empty |
| SW = Switch A = Access | SW01-A- HFA  SW02-A- HFA | Access | Switch 2960 |
| <n> = Number in the  department  <d> = Department name | <d>- HFA -<n> | - | PC |
| <s> = server type  <n> = number | HFA-<s>-<n> | - | server |

**דוגמאות לשמות התקני הקצה בסניף חיפה (HFA)**

|  |  |  |
| --- | --- | --- |
| **שם ההתקן** | **קיצור מחלקה** | **מחלקה** |
| admin-HFA-02  admin -HFA-03 | Adm | admin |
| legal -HFA-03 | lgl | legal |

# טבלת כתובות של סניפים

הקצאת כתובות הip נעשת כך שהאוקטטה הראשונה היא 10 מכיוון שמדובר בכתובת רשת פנימית ממחלקה A, האוקטטה השניה מציינת את מספר הסניף:1 – הרצליה, 2 – תל אביב, 3 – חיפה, האוקטטה השלישית מציינת את הvlan Id של המחלקה, והאוקטטה הרביעית שייכת לhosts.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| סניף | מחלקה | כמות שרתים | כמות מחשבים | VLAN ID | NETID | Subnet Mask | Default Gateway |
| הרצליה | Dev |  | 8 | 10 | 10.1.10.0 | 255.255.255.0 | 10.1.10.254 |
| QA |  | 6 | 20 | 10.1.20.0 | 255.255.255.0 | 10.1.20.254 |
| Guest |  | 8 | 30 | 10.1.30.0 | 255.255.255.0 | 10.1.30.254 |
| penTest |  | 6 | 40 | 10.1.40.0 | 255.255.255.0 | 10.1.40.254 |
| PR |  | 6 | 50 | 10.1.50.0 | 255.255.255.0 | 10.1.50.254 |
| Support |  | 6 | 60 | 10.1.60.0 | 255.255.255.0 | 10.1.60.254 |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| סניף | מחלקה | כמות שרתים | כמות מחשבים | VLAN ID | NETID | Subnet Mask | Default Gateway |
| תל אביב | UI |  | 8 | 10 | 10.2.10.0 | 255.255.255.0 | 10.2.10.254 |
| Executive |  | 6 | 20 | 10.2.20.0 | 255.255.255.0 | 10.2.20.254 |
| Guest |  | 8 | 30 | 10.2.30.0 | 255.255.255.0 | 10.2.30.254 |
| Security |  | 8 | 40 | 10.2.40.0 | 255.255.255.0 | 10.2.40.254 |
| HR |  | 8 | 50 | 10.2.50.0 | 255.255.255.0 | 10.2.50.254 |
| accounting |  | 8 | 60 | 10.2.60.0 | 255.255.255.0 | 10.2.60.254 |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| סניף | מחלקה | כמות שרתים | כמות מחשבים | VLAN ID | NETID | Subnet Mask | Default Gateway |
| חיפה | marketing |  | 6 | 10 | 10.3.10.0 | 255.255.255.0 | 10.3.10.254 |
| legal |  | 8 | 20 | 10.3.20.0 | 255.255.255.0 | 10.3.20.254 |
| Guest |  | 6 | 30 | 10.3.30.0 | 255.255.255.0 | 10.3.30.254 |
| logistics |  | 9 | 40 | 10.3.40.0 | 255.255.255.0 | 10.3.40.254 |
| cyberSec |  | 6 | 50 | 10.3.50.0 | 255.255.255.0 | 10.3.50.254 |
| admin |  | 6 | 60 | 10.3.60.0 | 255.255.255.0 | 10.3.60.254 |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

# הגדרת שמות וMTOD לרכיבים

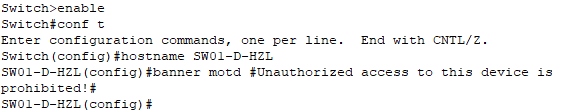
על מנת ליצור סדר ברשת ניתן לתת שם רכיב לכל רכיב רשת בעזרת הפקודה hostname. לאחר הגדרת הפקודה ניתן יהיה לראות את שם הרכיב המעודכן בפתיחה של כל שורת פקודה.

לפני כניסה למשתמש חשוב להראות למשתמש הצהרה שכניסה לא מורשת לרכיבים אסורה.

על מנת להגדיר שם לרכיב יש להכנס תחילה למצב קונפיגורציה ואז לרשום את הפקודה hostname ומיד לאחריו לרשום את השם

## הגדרת שמות לרכיבים בסניף הראשון:

הגדרת שם לSW01-D-HZL :



הגדרת שם לSW02-D-HZL :

Text

Description automatically generated

הגדרת שם לSW01-A-HZL :

Text

Description automatically generated

הגדרת שם לSW02-A-HZL :

Text

Description automatically generated

הגדרת שם לSW03-A-HZL :

Text, letter

Description automatically generated

הגדרת שם לRT01-C-HZL :

Text

Description automatically generated

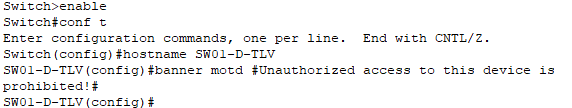
הגדרת שם לRT02-C-HZL :

Text

Description automatically generated

## הגדרת שמות לרכיבים בסניף השני:

הגדרת שם לSW01-D-TLV :

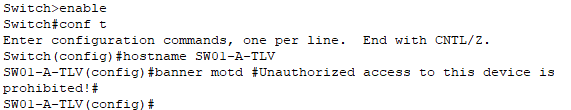


הגדרת שם ל SW02-D- TLV:

Text

Description automatically generated

הגדרת שם ל SW01-A- TLV:



הגדרת שם ל SW02-A- TLV:

Text

Description automatically generated

הגדרת שם ל SW03-A- TLV:

Text

Description automatically generated

הגדרת שם ל RT01-C- TLV:

Text

Description automatically generated

הגדרת שם ל RT02-C- TLV:

Text, letter

Description automatically generated

## הגדרת שמות לרכיבים בסניף השלישי:

הגדרת שם לSW01-D-HFA :

Text

Description automatically generated

הגדרת שם ל SW02-D- HFA:

Text

Description automatically generated

הגדרת שם ל SW01-A- HFA:

Text, letter

Description automatically generated

הגדרת שם ל SW02-A- HFA:

Text, letter

Description automatically generated

הגדרת שם ל SW03-A- HFA:

Text

Description automatically generated

# VTP

VTP הוא פרוטוקול רשתות בשכבה 2 שנועד לנהל את היצירה והמחיקה של רשתות VLAN ולשמור על סנכרון מסדי נתונים של VLAN ברחבי הרשת. באמצעות פרוטוקול זה, מנהל רשת יכול להוסיף או למחוק רשתות VLAN ולהעביר את השינויים הללו באופן אוטומטי לכל שאר המתגים ברשת. ללא VTP, מתגים אינם מחליפים מידע על רשתות VLAN ומנהל הרשת היה צריך להגדיר VLAN ידנית על כל אחד מהמתגים.

הפרוטוקול עוצב סביב רעיון הניהול הריכוזי. למתג אחד או יותר מוקצה התפקיד של שרת VTP. כל עדכונים שנעשו על מתגים אלה נשלחים דרך VTP למתגים האחרים, שהם לקוחות VTP.

## VTP Domain

VTP Domain מוגדר על ידי כל המתגים שחולקים את אותו שם תחום VTP. מתג יכול להיות בדומיין אחד בלבד.

כברירת מחדל, למתגים אין שם תחום VTP שהוקצה. כאשר הם מקבלים VTP advertisement דרך חיבור Trunk, הם יורשים את שם הדומיין ומספר גרסת ה-VTP שנמצא בהודעה. ניתן להגדיר את השם באופן ידני באמצעות הפקודה vtp name.

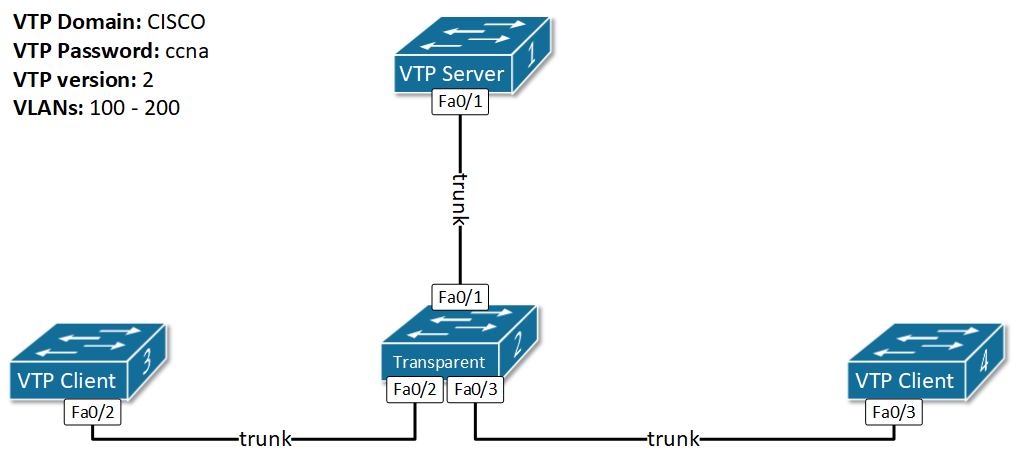
כל השינויים שנעשו בשרת ה-VTP מופצים רק למתגים בתחום ה-VTP. אם יש מתג שמוגדר עם שם תחום VTP אחר, הוא לא יקבל את הפרסומות ולא יעדכן את מסד הנתונים של ה-VLAN שלו.

## VTP Modes

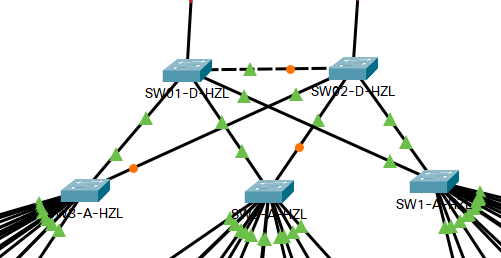
**VTP Server -** שרת VTP הוא מצב ברירת המחדל במתגי סיסקו. במצב Server, המתג מפרסם את כל השינויים במסד הנתונים של VLAN לכל שאר המתגים באותו VTP Domain. הודעות broadcast אלו נשלחות דרך קישורי trunk

**VTP Client -** לקוחות VTP משדרים ומקבלים עדכוני VTP אך אינם יכולים ליצור או למחוק רשתות VLAN. כל השינויים ב-VLAN נעשים על המתג במצב שרת.

**VTP Transparent -** מתגים במצב VTP Transparent משדרים ומקבלים עדכוני VTP אך אינם מעדכנים את מסד הנתונים של ה-VLAN שלהם. הם אינם משתתפים כלל ב-VTP ורק מעבירים הודעות VTP.



יש להגדיר חיבורי TRUNK בין כל המתגים שמשתמשים בVTP לפני הגדרת הVTP.



הגדרת VTP במתג SW01-D-HZL בתור השרת:

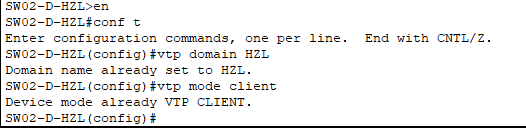
תחילה ניכנס למצב הקונפיגורציה. נגדיר למתג שיעבוד במצב שרת, ולאחר מכן נגדיר לו שם דומיין.

Text

Description automatically generated

הגדרת VTP במתג SW02-D-HZL בתור לקוח:

תחילה ניכנס למצב הקונפיגורציה. נגדיר למתג שיעבוד במצב לקוח, ולאחר מכן נגדיר לו שם דומיין.



פקודת show vtp status בתור שרת:

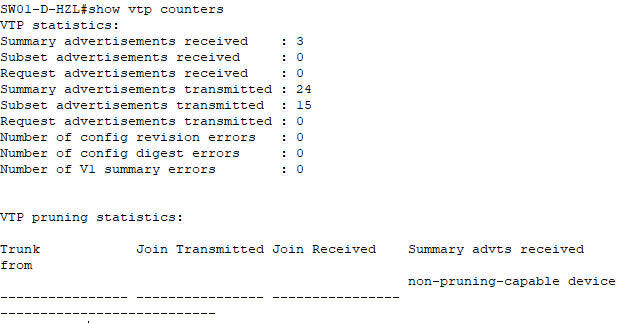
בפקודה זו ניתן למצוא פרטים כללים על הVTP כמו הגרסה שלו, הConfiguration Revision, באיזה מצב המתג עובד, באיזה דומיין הוא נמצא

Text

Description automatically generated

פקודת show counters:

בפקודה זו אנו מקבלים נתונים סטטיסטים על המידע שעובר בפרוטוקול VTP.



# VLAN’s

רשת LAN וירטואלית (VLAN) היא רשת לוגית שמקבצת יחד תת-קבוצה של התקנים החולקים רשת LAN פיזית, ומבודדת את התעבורה עבור כל קבוצה.

LAN היא קבוצה של מחשבים או התקנים אחרים באותו מקום - למשל, באותו בניין או קמפוס - שחולקים את אותה רשת פיזית. LAN משויך בדרך כלל לתחום שידור בשכבה 2, שהוא קבוצת התקני הרשת שחבילת שידור Ethernet יכולה להגיע אליה.

מחשבים ברשת ה-LAN מתחברים לאותו מתג רשת, ישירות או דרך נקודות גישה אלחוטיות (APs) המחוברות לאותו מתג. מחשבים יכולים גם להתחבר לאחד מתוך קבוצה של מתגים המחוברים זה לזה דרך חיבור trunk.

VLAN, כמו ה-LAN שהוא יושב עליו, פועל בשכבה 2 של הרשת, ברמת ה-Ethernet. רשתות VLAN מחלקות רשת מיתוג יחידה לקבוצה של רשתות וירטואליות מכוסות על גבי, שיכולות לעמוד בדרישות תפקודיות ואבטחה שונות. חלוקה זו מונעת את הצורך להחזיק במספר רשתות פיזיות שונות עבור מקרי שימוש שונים ובכך מאפשרת לחברה לחסוך לא מעט כסף על ציוד רשת.

מהנדסי רשת משתמשים ב-VLAN מסיבות מרובות, כולל הבאות:

לשיפור הביצועים

להדק את האבטחה

כדי להקל על הניהול

## כיצד פועלים רשתות VLAN

VLAN מזוהה במתגי רשת על ידי ID VLAN. לכל יציאה במתג ניתן להקצות מזהה VLAN אחד או יותר והוא יעבור ל-VLAN ברירת מחדל אם לא יוקצה עוד אחד. כל VLAN מספק גישה לקישור נתונים לכל hosts המחוברים ליציאות מתג המוגדרות עם מזהה ה-VLAN שלו.

מזהה VLAN מתורגם לתג VLAN, שדה של 12 סיביות בנתוני הכותרת של כל מסגרת Ethernet שנשלחת לאותו VLAN. מכיוון שתג הוא באורך 12 סיביות, ניתן להגדיר עד 4,096 רשתות VLAN לכל תחום מיתוג.

כאשר מתקבלת frame Ethernet מhost ​​מחובר, אין לה תג VLAN. המתג מוסיף את תג VLAN.

הוא מעביר קדימה frames מתויגות לעבר כתובת mac של היעד, ומעביר רק ליציאות שאיתן משויך ה-VLAN. תעבורת broadcast, unicast ו-multicast לא ידועה מועברת לכל היציאות ב-VLAN. קישורי trunk בין מתגים יודעים אילו רשתות VLAN נמצאות על המתגים, מקבלים ומעבירים לאורך כל התעבורה עבור כל VLAN שנמצא בשימוש משני צידי הtrunk.

# הגדרת vlan’s

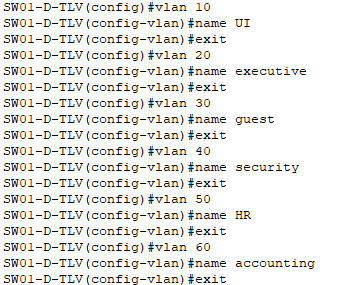
על מנת להגדיר vlan נכתוב תחילה vlan ואז המספר שלו. לאחר הפקודה הזו אנו נכנסים למצב קונפיגורציה של הVLAN ובו נגדיר תשם שלו על ידי הפקודה name.

הגדרת VLAN במתג SW01-D-HZL:

Text, table

Description automatically generated

הגדרת VLAN במתג SW01-D-TLV (תקף לכל המתגים ברשת TLV):



הגדרת VLAN במתג SW01-D-HFA (תקף לכל המתגים ברשת HFA):

Text

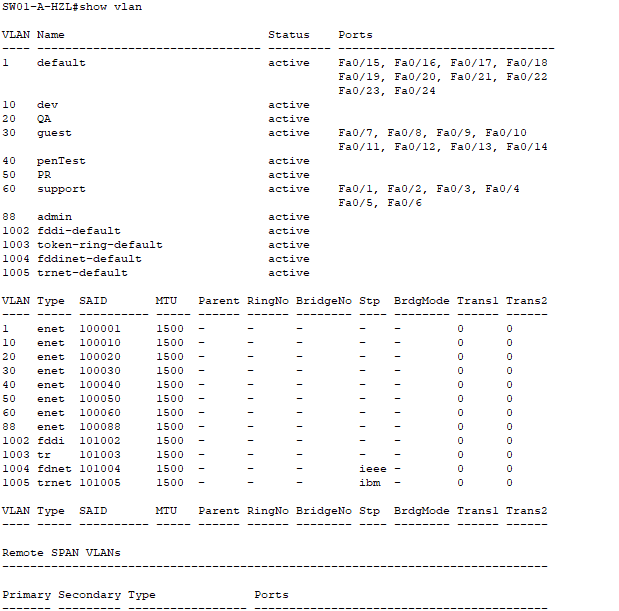
Description automatically generated

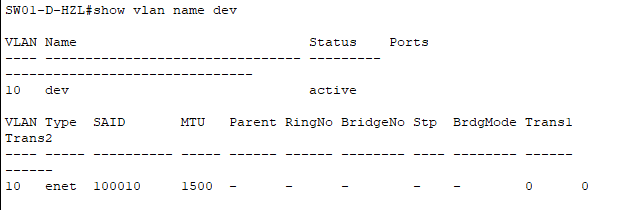
פקודת show vlan brief בסניף הראשון:

Table

Description automatically generated

פקודת show vlan בסניף הראשון:



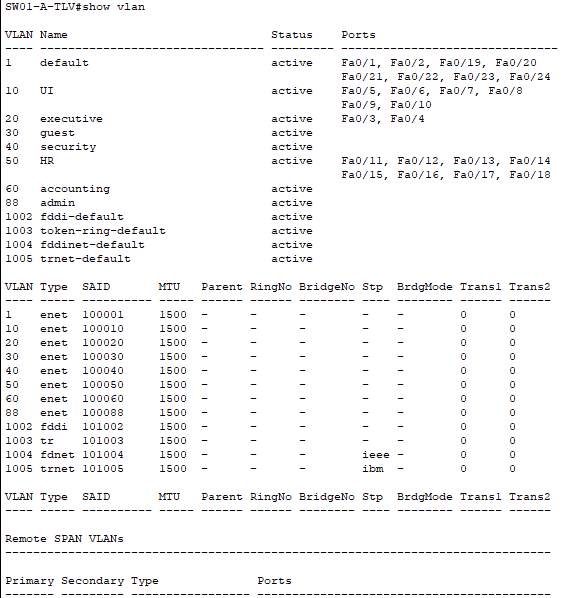


פקודת show vlan brief בסניף השני:

Table

Description automatically generated with low confidence

פקודת show vlan בסניף השני:

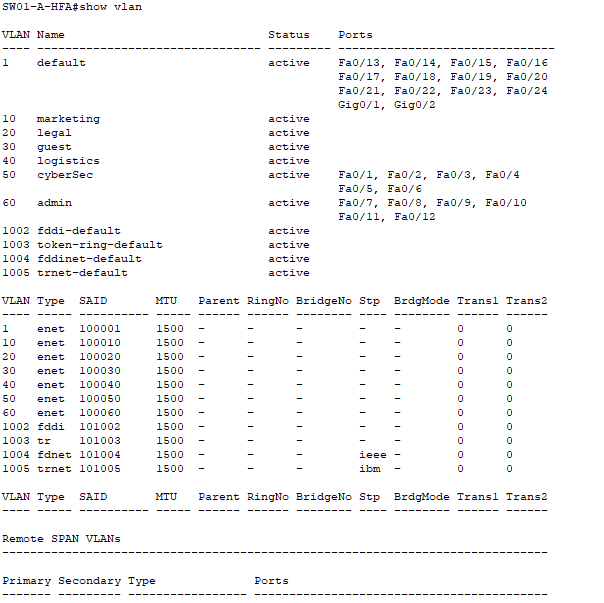


פקודת show vlan brief בסניף השלישי:

Table

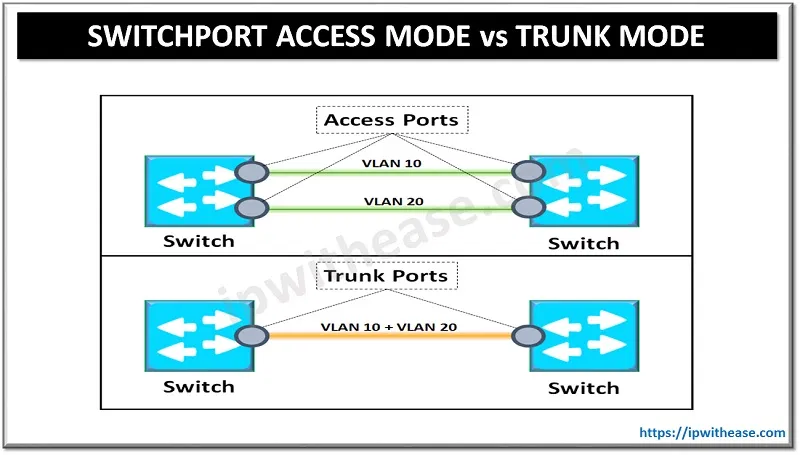
Description automatically generated with medium confidence

פקודת show vlan בסניף השלישי:



# Trunk

Trunk הוא הגדרה של ממשק שדרכו ניתן להעביר יותר מvlan אחד. בחיבור הזה משתמשים כמעט בכל רשת מקומית בין מתגים או מתג לנתב. על מנת להגדיר ממשק כtrunk יש להיכנס תחילה להגדרות הממשק דרך הפקודה interface ואז לכתוב switchport mode trunk.



הגדרת ממשק כ trunk במתג שכבה 2 בסניף הראשון:

נכנסים למצב הגדרת ממשק

מגדירים את הswitchport לעבוד כtrunk

Text, letter

Description automatically generated

הגדרת ממשק כ trunk במתג שכבה 3 בסניף השני:

במתגי שכבה 3 יש להוסיף את הפקודהswitchport trunk encapsulation dot1q על מנת שהנתב יוכל לעבוד בtrunk.



כדי לבדוק האם הגדרנו את הממשק כtrunk, נשתמש בפקודה show interfaces trunk:

Table

Description automatically generated

# HSRP

רשת מודרנית דורשת זמינות תמידית בכל רגע נתון גם אם ישנה תקלה באחד ממכשירי הרשת עם הבנה זו, פרוטוקולי יתירות First Hop פותחו והופעלו ברוב הרשתות כדי לספק גמישות, זמינות ויתירות. מנקודת המבט של הלקוח אם השער בריירת מחדל יירד, הגישה לרשת שלמה תרד. פרוטוקולי First Hop Redundancy protocol )FHRP) יאפשרו יתירות שער ברירת מחדל, דרך יותר מנתב אחד.

במקרה של תקלה בנתב, יש מכשיר גיבוי שיכנס וימשיך להעביר תעבורה לרשתות מרוחקות ובכך ימנע מצב של ניתוק מרשתות אחרות. להלן 3 סוגי טכנולוגיית FHRP:

פרוטוקול נתב המתנה חם (HSRP)

פרוטוקול יתירות נתב וירטואלי (VRRP)

פרוטוקול איזון עומסים של שער (GLBP)

## מהו HSRP

Hot Standby Router Protocol (HSRP) הוא פרוטוקול בבעלות של CISCO, המספק מנגנון שנועד לתמוך בכשל של תעבורת רשת בנסיבות מסוימות. במקרה זה, שני נתבים או יותר נותנים אשליה של נתב וירטואלי. HSRP מאפשר לך להגדיר נתבים עם כתובת IP משותפת שמשמשת כ default gateway ורק נתב בודד כפעיל בכל פעם. כל הנתבים בקבוצת HSRP חולקים כתובת MAC וכתובת IP אחת, המשמשת כשער ברירת מחדל לרשת המקומית. הנתב הפעיל מעביר את התנועה. אם הנתב הפעיל נכשל, נתב הגיבוי לוקח על עצמו את כל האחריות של הנתב הפעיל ומעביר את התעבורה.

## כיצד מורכבת פקטת HSRP

Opcode הוא 8 סיביות.

0 Opcode - ה-HSRP פועל ומסוגל להפוך לנתב הפעיל או המתנה.

Opcode 1 - הפיכה. הנתב הופך לנתב הפעיל.

Opcode 2 - התפטר. הנתב אינו עוד הנתב הפעיל.

HSRP States הוא 8 סיביות.

1. פעיל - זהו המצב של המכשיר שמעביר תנועה באופן פעיל.

2. Init או Disabled - זהו המצב של מכשיר שעדיין לא מוכן או מסוגל להשתתף ב-HSRP.

3. למד - זהו המצב של מכשיר שעדיין לא קבע את כתובת ה-IP הוירטואלית ועדיין לא ראה הודעת hello ממכשיר פעיל.

4. הקשב - זהו המצב של מכשיר שמקבל הודעות hello.

5. דיבור - זהו המצב של מכשיר ששולח ומקבל הודעות hello.

6. המתנה – זהו מצבו של מכשיר שמוכן לקחת על עצמו את חובות העברת התנועה מהמכשיר הפעיל.

hello time הוא 8 ביטים. המרווח בין הודעות HSRP helloעוקבות מנתב נתון הוא 3 שניות.

החזק זמן את המרווח בין קבלת הודעת hello לבין ההנחה שהנתב השולח כשל לאחר 10 שניות.

העדיפות היא 8 ביטים.

עדיפות ברירת המחדל היא 100. נתב עם עדיפות גבוהה יותר זוכה. שדה עדיפות משמש בתהליך הבחירות הנתבים הפעילים וההמתנה. במצב של שבירת שוויון, כתובת ה-IP הגבוהה ביותר מנצחת.

הקבוצה היא 8 ביט.

שדה זה מזהה את קבוצת ההמתנה בין 0 ל-255.

השמור הוא 8 ביט.

נתוני אימות הם 64 סיביות.

שדה זה מכיל טקסט ברור של 8 תווים בסיסמה בשימוש חוזר. אם לא מוגדרים נתוני אימות, ערך ברירת המחדל המומלץ הוא 0x63 0x69 0x73 0x63 0x6F 0x00 0x00 0x00.

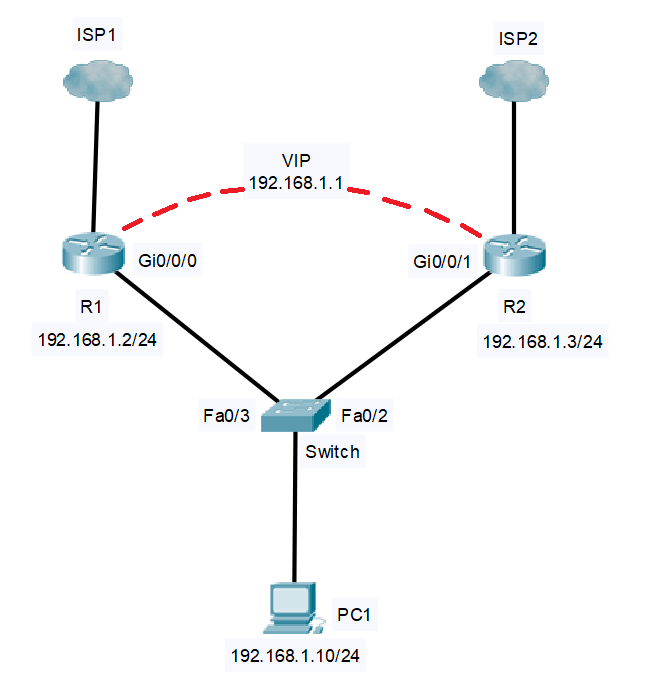
כתובת IP וירטואלית היא 32 סיביות.

כתובת ה-IP הוירטואלית שבה משתמשת קבוצה זו. אם כתובת ה-IP הוירטואלית אינה מוגדרת בנתב, ייתכן שהיא נלמדת מהודעת hello מהנתב הפעיל. יש ללמוד כתובת רק אם לא הוגדרה כתובת והודעת Hello מאומתת.

הראוטר שהוא Active עונה לבקשות ARP של Router על ידי ה Virtual Mac Adress, והוא יודע את כתובת ה IP הוירטואלית שלו.

הראוטר שבמצב Active שולח הודעת Hello לראוטר במצב Passive כל 3 שניות. במידה והראוטר Passive לא קיבל את ההודעה הוא יהפוך את עצמו לActive, וזאת כי הוא הניח שהרשת נפלה.

כאשר הראוטר הקודם שהיה Active חוזר לרשת הוא עובר לעבוד במצב Passive. במידה ואנחנו מעדיפים שהוא יעבוד כActive בכל רגע שהוא פועל, נשתמש בפקודה preempt שתדרוש מראוטרים אחרים לתת לו להיות Active במקומם.



# הגדרת HSRP

הגדרת HSRP בנתב הראשי בסניף הראשון על VLAN 10:

תחילה אנו נכנסים לתת ממשק ומגדירים שיעבוד בvlan 10

לאחר מכן נגדיר כתובת IP לממשק (חשוב שהכתובת תהיה ייחודית לממשק הזה ברשת!).

בפקודה הבאה אנו מגדירים את כתובת הIP הוירטואלית שלו(הכתובת ברירת מחדל).

נגדיר את הפקודה preempt כך שהתעבורה תעבור תמיד דרך הנתב הראשי כל עוד הוא עובד.

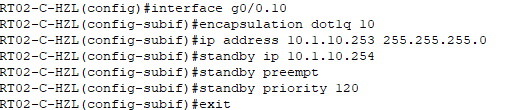
ניתן לנתבpriority הכי גבוה כך שהוא יהיה הנתב הראשי

Text, letter

Description automatically generated

הגדרת HSRP בנתב המשני בסניף הראשון על VLAN 10:

בנתב המשני חוזרים על אותם הפעולות רק מגדירים priority נמוך יותר כך שיחשב למשני.

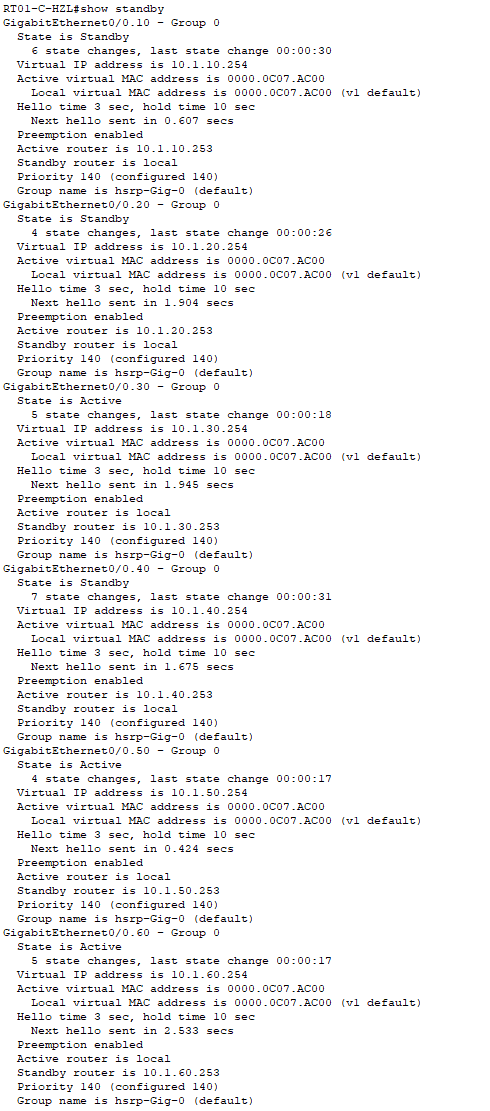


הגדרת HSRP במתג שכבה 3 בסניף השני:

במתג שכבה 3 חוזרים על אותם פעולות כמו בנתב רק שמגדירים HSRP על ממשק VLAN ולא תת ממשק.

Text

Description automatically generated



# STP

stp או spanning tree protocol הוא פרוטוקול בשכבה 2 במודל הOSI האחראי על מניעת לולאות בתוך רשת מחשבים.

STP נוצר כדי למנוע את הבעיות המתעוררות כאשר מחשבים מחליפים נתונים ברשת מקומית (LAN) המכילה נתיבי גיבוי. אם זרימת התעבורה אינה מפוקחת ונשלטת בקפידה, הנתונים יכולים להיתפס בלולאה שמסתובבת סביב מקטעי רשת, משפיעה על הביצועים ועלולה אף לגרום לעצירה כמעט מוחלטת של התעבורה.

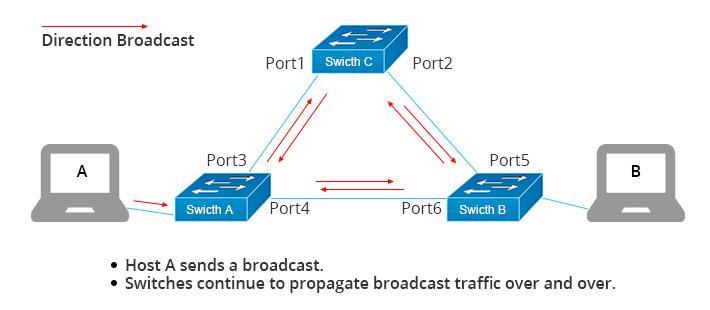
ברשתות תקשורת של ארגונים רבים ניתן למצוא יתירות בכבלים ורכיבי רשת אשר מאפשרים גלישה גם במידה ואחד הרכיבים לא פועל כמו שצריך. יתירות זו יוצרת מספר נתיבים דרכם המידע יכול לעבור מה שעלול ליצור מצב של לולאה.

STP יכול לסייע במניעת לולאת ברשתות LAN הכוללות חיבורים מיותרים. ללא STP, יהיה קשה ליישם את היתירות הזו ועדיין להימנע מלולאת רשת. STP מנטר את כל חיבורי הרשת, מזהה חיבורים מיותרים ומשבית את היציאות שעלולות להוביל ללולאה.

## בעיות שהSTP פותר

### Broadcast storm

הבעיה הראשונה והגדולה מכולם נקראת broadcast storm והיא למעשה נגרמת כאשר יש הודעת broadcast שנשלחת לרשת. מכיוון שמדובר בהודעת broadcast המתגים מעבירים אותה לכל הממשקים שיש לה הנמצאים באותו vlan או במצב trunk , וזה גורם לבעיה מכיוון שכל מתג יעביר את ההודעה למתג הבא ואם הם מחוברים במעגל זה יצור לולאה אינסופית אשר בקלות יכולה לתקוע את הרשת.



**unstable mac table**

הבעיה השניה נקראת unstable mac table והיא נגרמת מזה שהמתג לומד על הmac באמצעות הפריימים שעוברים דרכה, וכאשר נוצרת לולאה הפריימים מבלבלים את המתג.

Diagram

Description automatically generated

**Duplicate Frame**

הבעיה השלישית והאחרונה היא duplicate frame והיא נוצרת כאשר אותו פריים עובר דרך כמה נתיבים בו זמנית מה שהופך את העברת המידע ללא יעילה.

## STP port states

כאשר STP מופעל על מתג ברשת, כל יציאה מוגדרת לאחד מחמישה מצבים כדי לשלוט בהעברת frames:

1. Disabled – הממשק אינו משתתף בהעברת frames.
2. Blocking - הממשק אינו משתתף בהעברת frames ומבטל frames שהתקבלו מקטע הרשת המצורף. עם זאת, הממשק ממשיך להאזין ולעבד BPDUs.
3. Listening - ממצב החסימה, הממשק עובר למצב האזנה. הממשק משליך frames. עם זאת, הוא מקבל BPDUs ומעבד אותם.
4. Learning - הממשק עובר ממצב האזנה למצב הלמידה. הוא מאזין ומעבד BPDUs אך משליך frames. הוא גם מתחיל לעדכן את טבלת הכתובות עם המידע שנלמד. בנוסף, הוא מעבד frames אך אינו מעביר את אותם.
5. Forwarding - הממשק עובר ממצב הלמידה למצב ההעברה ומתחיל להעביר פריימים על פני מקטעי הרשת. זה כולל frames מקטע הרשת המצורפת ואלו שהועברו מיציאה אחרת. היציאה גם ממשיכה לקבל ולעבד BPDUs, וטבלת הכתובות ממשיכה להתעדכן.

### Root Bridge

זה המתג דרכו חייבת לעבור כל התעבורה. לכל הגשרים ברשת LAN יש נתיב לroot bridgh. STP מאפשר לך לבחור את גשר השורש באופן אוטומטי. עם זאת, אם מנהל רשת STP רוצה, הוא או היא יכולים לשנות את ה-RB בהתאם לרשת.

כיצד נבחר הroot bridge ברשת?

כל המתגים ברשת מכריזים על עצמם כroot bridge ומתחילים להחליף BPDUs משלהם. ה-BPDU עם הpriority הנמוך ביותר ייחשב כעליון. כעת המתג המקבל את ה-BPDU העליון מבצע שינויים ב-BPDU שלו ומעביר הלאה למתגים השכנים שלו. הוא משנה את הערך של root Bridge I'd עם הroot ID של ה-BPDU העליון שלו. תהליך זה נמשך עד שכל המתגים יודעים למי יש את הpriority הנמוך ביותר והמתג הזה יוכרז כroot bridge .

### Root Port

ממשק הroot הוא ממשק המוביל לroot bridgh.

## Designated Port

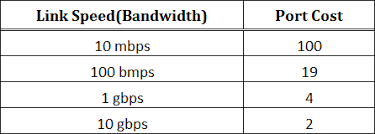
ממשק זה מקבל את המידע ומעביר אותו לroot port

### Alternative Port

ממשק זה משמש כממשק חלופי לממשק root והוא נמצא במצב חסימה.

### Path Cost

עוזר להחליט על הטופולוגיה הטובה ביותר ביחס למהירות ההעברה. ככל שמהירות ההעברה של החיבור גדולה יותר כך הcost שלה נמוך יותר. הרשת תמיד תעדיף את הנתיב שסכום הcosts שלו הוא הנמוך ביותר.



## Bridge Protocol Data Units (BPDU)

פרוטוקול Spanning Tree דורש התקן רשת STP להחליף הודעות כדי לעזור מטופולוגיה לוגית נטולת לולאות. הודעות אלו נקראות BPDUS (Bridge Protocol Data Units). כל התקן רשת שולח BPDUs, שעוזר להחליף מידע טופולוגי.

BPDUS עוזר למתגים לבצע את הפעולות הבאות:

* לבחור מתג בודד שיפעל כroot
* BPDU גם עוזר לחשב את הנתיב הקצר ביותר מעצמו למתג הroot.
* לחסום alternative ports

## סוגים נוספים של STP

### RSTP

Stp אשר עולה מהר יותר אך גם צורך יותר משאבים.

### PVST

Stp אשר פועל על כמה vlans במקביל.

### Rapid-PVST

Rapid-pvst הוא pvst אשר עולה מהר יותר אך גם צורך יותר משאבים.

## הגדרות בסיסיות ואבטחתיות של STP

### Portfast

Switchport המחובר למכשיר קצה מאפשר לאותו מכשיר להתחבר לרשת באופן מידי, במקום לחכות שהממשק יעבור ממצבי האזנה ולמידה למצב העברה. לא רצוי להגדיר portfast לחיבור אשר לא מתחבר למכשיר קצה כיוון שהוא מדלג על שלבים חשובים אשר עוזרים למנוע לולאות.

### BPDU Guard

BPDU Guard היא תכונה המגנה על הטופולוגיה של Layer 2 Spanning Tree Protocol (STP) מפני איומים הקשורים ל-BPDU ונועדה להגן על רשת המיתוג. יש להפעיל את תכונת ה-BPDU guard ביציאות שלא אמורות לקבל BPDUs מהתקנים מחוברים.

### Root Guard

Root guard הוא תכונת STP המופעלת על הroot bridgh אשר מתעלמת מפאקטות BPDU עליונות. זה מונע מממשק מוגדר להפוך לroot port. Root guard מונע ממתג במורד הזרם (לעתים קרובות מוגדר שגוי או זדוני) להפוך לroot bridgh בטופולוגיה

הגדרת STP במתג שכבה 2 בשכבת ה Distribution בסניף הראשון SW01-D-HZL מוגדר כRoot bridgh:

Text, letter

Description automatically generated

הגדרת STP במתג שכבה 2 בשכבת ה Distribution בסניף הראשון SW02-D-HZL מוגדר כBackup:

Text

Description automatically generated

הגדרת STP במתג שכבה 2 בשכבת ה Access בסניף הראשון SW01-A-HZL:

A screenshot of a computer

Description automatically generated with medium confidence



פקודות show: Table

Description automatically generated

Text, letter

Description automatically generatedA picture containing text

Description automatically generatedTable

Description automatically generated

# Router On a Stick (Dot1Q)

נתב על מקל הוא אחת הדרכים לאפשר ניתוב בין רשתות VLAN. סוג זה של התקנה מורכב מנתב ומתג המחוברים דרך חיבור אחד המוגדר כחיבור trunk 802.1q. תצורה כזו אופיינית ברשתות שבהן לא קיים מתג שכבה-3.

בעת קביעת תצורה של רשתות VLAN חדשות ביציאות מתג, חילקנו את תחום השידור שבו הוא הוצב לחלקים קטנים יותר מוגבלים ל-VLAN. המתג הבסיסי אינו יכול לנתב מידע בין תחומי שידור. ידוע שהנתב הוא מכשיר שמנתב פאקטות ומפריד בין תחומי שידור.

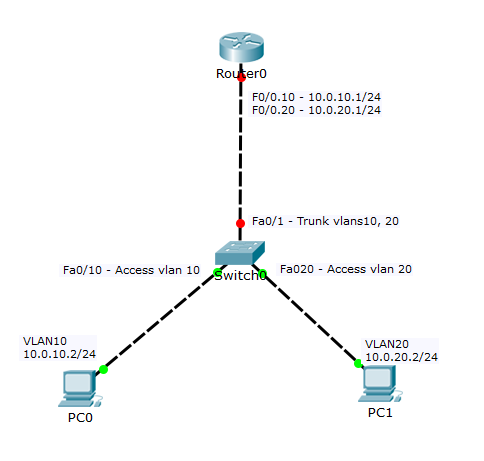
ניתן להשתמש ב שכבה 3 כדי לאפשר תקשורת בין התקנים מ-VLAN שונים. ראוטר על מקל היא דרך אחת בה נוכל ליישם זאת.

## Sub-interfaces

Sub interface הוא תכונה מרכזית של הנתב על מקל. הוא מוגדר עבור כל רשת בנתב.

ממשקי משנה הם המרכיבים הלוגיים של ממשק פיזי. הודות לגישה זו אין צורך להשתמש ב-X ממשקים פיזיים של הנתב ב-Y VLANs. במקום זאת, קיים קישור trunk בין מתג לנתב, כך שמנות מסומנות עוברות לנתב. לאחר מכן, הנתב מסיר את התוויות ומציג תצוגה מקדימה של טבלת הניתוב.

חשוב להגדיר encapsulation 802.1Q על כל תת ממשק ותג VLAN תואם גם כן.



**הגדרת dot1q**

**יש להיכנס לתת ממשק של הממשק המחובר כTRUNK ואז ולהגדיר את פקודת ה encapsulation dot1q ואת מספר מזהה הvlan.**

****

# **DHCP**

DHCP הוא מנגנון הקצאת כתובות IP למארחים קבועים וניידים המחוברים קווית או אלחוטית.

אם פרוטוקול DHCP לא היה קיים, מנהלי רשת יצטרכו לחלק באופן ידני כתובות IP מהמאגר הזמין, דבר שיצרך זמן רב, לא יעיל ונוטה לשגיאות.

כאשר מכשיר רוצה גישה לרשת המשתמשת ב-DHCP, הוא שולח בקשה לכתובת IP שנקלטת על ידי שרת DHCP. השרת מגיב במסירת כתובת IP למכשיר, ואז עוקב אחר השימוש בכתובת ומחזיר אותה לאחר זמן מוגדר או כשהמכשיר נכבה. לאחר מכן, כתובת ה-IP מוחזרת למאגר הכתובות המנוהל על ידי שרת ה-DHCP כדי להיות מוקצה מחדש למכשיר אחר בזמן שהוא מחפש גישה לרשת.

בעוד שהקצבת כתובות IP היא הפונקציה המרכזית של הפרוטוקול, DHCP גם מקצה מגוון פרמטרים קשורים לרשת, כולל כתובת שער ברירת מחדל וכתובת שרת DNS.

## יתרונות הDHCP

DHCP מספק מגוון יתרונות למנהלי רשת:

### תצורת כתובת IP אמינה

לא יכולים להיות שני משתמשים עם אותה כתובת IP מכיוון שזה יוצר התנגשות שבה אחד או שני המכשירים לא יכולים להתחבר לרשת. DHCP מבטל שגיאות אנוש כך שהתנגשות כתובות או שגיאות הקלדה פשוטות ממוזערות.

### ניידות

DHCP מטפל ביעילות בשינויי כתובת IP עבור משתמשים במכשירים ניידים שעוברים למיקומים שונים ברשתות קוויות או אלחוטיות.

### שרת DHCP

זהו מכשיר רשת המריץ את שירות DHCP שמכיל כתובות IP. זה לרוב שרת או נתב אבל יכול להיות כל דבר שמתפקד כhost

### לקוח DHCP

תוכנת נקודת קצה זו מבקשת ומקבלת מידע משרת DHCP. ניתן להתקין זאת על מחשב, מכשיר נייד, IoT או כל דבר אחר שדורש קישוריות לרשת. רובם מוגדרים לקבל מידע DHCP כברירת מחדל.

### מאגר כתובות IP

טווח כתובות ה-IP הזמינות ללקוחות. כתובות בדרך כלל מחולקות ברצף מהנמוך לגבוה ביותר.

### Subnet

ניתן לחלק רשתות IP למקטעים המכונים subnet.

**Lease**

משך הזמן שבו לקוח DHCP מחזיק את פרטי כתובת ה-IP. כאשר חוזה שכירות מסתיים, על הלקוח לחדשו.

בשרתות המאפשרות גישה למכשירים ניידים רצוי לקבוע זמן נמוך על מנת לשחרר את הכתובת ip כמה שיותר מוקדם לאחר שהמכשיר התנתק מהרשת.

**relay DHCP**

נתב או מארח המאזין להודעות לקוח המשודרות ברשת זו ולאחר מכן מעביר אותן לשרת מוגדר המוגדר כ-DHCP. לאחר מכן השרת שולח תגובות בחזרה לrelay שמעביר אותן ללקוח. זה יכול לשמש כדי לרכז שרתי DHCP במקום שיהיה שרת בכל תת רשת.

## תהליך הקצאת כתובות IP

השאלה הקיומית הקשורה ל-DHCP היא כיצד משתמש קצה מתחבר לרשת מלכתחילה מבלי שיש לו כתובת IP?

ראשית, כל מערכות ההפעלה המודרניות של המכשיר כוללות לקוח DHCP, המופעל בדרך כלל כברירת מחדל. על מנת לבקש כתובת IP, מכשיר הלקוח שולח הודעת שידור –DISCOVER. הרשת מפנה את הבקשה לשרת ה-DHCP המתאים.

לאחר מכן השרת קובע את כתובת ה-IP המתאימה ושולח חבילת OFFER ללקוח, אשר מגיב בחבילת REQUEST. בשלב האחרון בתהליך, השרת שולח חבילת ACK המאשרת שהלקוח קיבל כתובת IP.

כל זה נעשה במהירות ובאופן אוטומטי וללא צורך בפעולה כלשהי של משתמש.

**אבטחת DHCP**

עם DHCP, ההקצאה הראשונית של כתובת IP מתוכננת להיות מהירה ויעילה. הפשרה היא שפרוטוקול DHCP אינו דורש אימות.

תוקף עלול להשתלט או לזייף את שרת ה-DHCP ולחלק מידע גרוע למשתמשי קצה לגיטימיים, לשלוח אותם לאתר מזויף. או שהוא יכול לחלק כתובות IP לגיטימיות למשתמשים לא מורשים. זה עלול להוביל להתקפות "man in the middle" ולהתקפות של מניעת שירות.

# **DHCP Snooping**

DHCP Snooping היא טכנולוגיית אבטחה על מתג רשת שכבה 2 שיכולה למנוע משרתי DHCP לא מורשים לגשת לרשת שלך. זוהי הגנה מפני hosts שרוצים להפוך לשרתי DHCP Snooping. DHCP פועל כהגנה מפני התקפות " man in the middle ". DHCP עצמו פועל על שכבה 3 של שכבת OSI בעוד DHCP snooping פועלת על התקני שכבה 2 כדי לסנן את התעבורה שמגיעה מלקוחות DHCP.

במתגים של Cisco, DHCP Snooping מופעלת באופן ידני. יש להגדיר trusted ports באופן ידני ושאר היציאות לא מוגדרות נחשבות ליציאות untrusted. רוב המכשירים המחוברים ל trusted ports הם נתבים, מתגים ושרתים. לקוחות DHCP כמו PC ומחשבים ניידים מחוברים בדרך כלל ליציאה untrusted.

## הגדרת DHCP

הגדרת שרת DHCP בנתב cisco בסניף הראשון RT01-C-HZL:

נכנסים למצב קונפיגורצית הנתב.

משתמשים בפקודת ip dhcp excluded-address לכל הכתובות שאנו לא רוצים שהdhcp יחלק.

נכנסים למצב הגדרת DHCP pool לכל pool בנפרד על ידי הפקודה ip dhcp pool ואז השם שלה.

בתוך למצב הגדרת DHCP pool נגדיר את שם הרשת וה subnet mask שלה.

נגדיר את הdefault gateway של הרשת על מנת לשלוח אותה לרכיבים שידעו אותה.

Text

Description automatically generated

הגדרת שרת DHCP על שרת קצה בPacket tracer בסניף השני:

לאיזה ממשק נגדיר DHCP פועל או כבוי.

כאשר אנו רוצים ליצור pool חדש נמלא את הפרטים הבאים ונלחץ על הכפתור save.

Graphical user interface, application

Description automatically generated

הגדרת מתגי שכבה 3 בשכבת הDistribution כך שיעבירו DHCP לשרת DHCP בשרת אחרת:

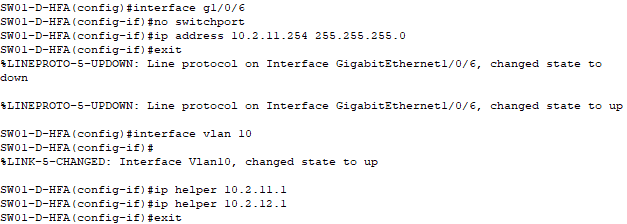
תחילה ניכנס לממשק המחובר לרשת בה נמצא שרת הDHCP.

נגדיר שהממשק יעבוד בשכבה 3 ולא כswitchport.

ניתן לו כתובת IP על מנת שיוכל לתקשר ברשת של הDHCP.

נכנס למצב הגדרת VLAN לכל אחד מהVLANים.

נגדיר את פקודת הip helper אשר יעביר DHCP לכתובת המוגדרת.



פקודות show

פקודת show ip dhcp pool:

מראה פרטים על כל POOL בנפרד.

Diagram

Description automatically generated with medium confidence

פקודת show ip dhcp binding:

פקודה המציגה את כל הקצאות הכתובות IP לכתובות mac, לכמה זמן ,ואיזה סוג.

Table

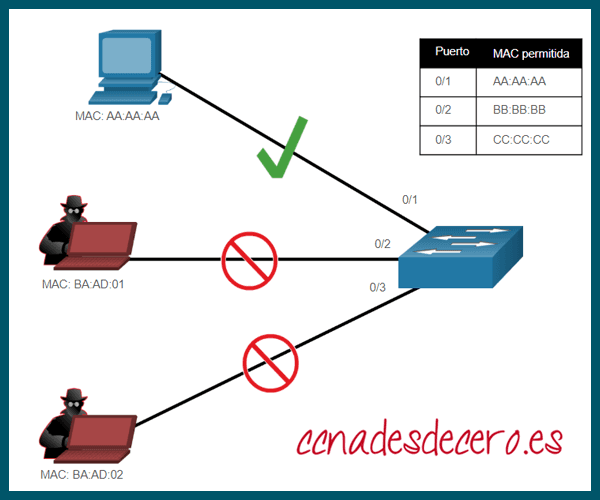
Description automatically generated

# Port-Security

רשתות LAN פגיעות מאוד להתקפה מכיוון שיציאות המתג פתוחות לשימוש כברירת מחדל. התקפות שונות כגון התקפת Dos בשכבה 2, זיוף כתובות יכול להתרחש. על מנת לפתור את הבעיה זו, נוצר מנגנון port security אשר מונע גישה לרשת לרכיבים אשר לא מורשים להיכנס.

## איך זה עובד?

מתגים לומדים כתובות MAC כאשר ה frame מועבר דרך switchport. על ידי שימוש בport security, משתמשים יכולים להגביל את מספר כתובות MAC שניתן ללמוד ליציאה, להגדיר כתובות MAC סטטיות ולהגדיר violation עבור אותה יציאה אם רכיב לא מורשה מנסה להתחבר דרכו . משתמשים יכולים להשתמש בפקודות restrict, shutdown או protect על port security.



* Protect – מצב זה מפיל את החבילות עם כתובות mac מקור לא ידועות עד שתסיר מספיק כתובות mac מאובטחות כדי לרדת מתחת לערך המקסימלי.
* Restrict – מצב זה מבצע את אותה פונקציה כמו הגנה, כלומר מוריד מנות עד שיוסרו מספיק כתובות Mac מאובטחות כדי לרדת מתחת לערך המקסימלי. בנוסף לכך, הוא יפיק log.
* Shutdown – מצב זה מועדף בעיקר בהשוואה למצבים אחרים מכיוון שהוא מכבה את הממשק מיד אם מתבצעת גישה לא מורשית ויפיק log. יציאה זו תישאר במצב כיבוי עד שמנהל המערכת יבצע את הפקודה "ללא כיבוי".
* Sticky – על ידי שימוש בפקודה sticky, המשתמש מספק אבטחת כתובת Mac סטטית מבלי להקליד את כתובת MAC המוחלטת. לדוגמה, אם המשתמש מספק מגבלה מקסימלית של 2 אז 2 כתובות ה-Mac הראשונות שנלמדו ביציאה זו יוצבו בתצורה הפועלת. לאחר כתובת ה-Mac השניה שנלמדה, אם המשתמש השלישי רוצה לגשת, המערכת תיצור violation.

הערה – port security יעבוד על access port בלבד, כלומר כדי לאפשר port security, המשתמש צריך קודם כל להפוך אותה לaccess port.

## הגדרת port security

כאשר אנו רוצים להגדיר port security ניכנס תחילה למצב קונפיגורצית ממשק.

נהפוך את הswitchport לעבוד במצב access.

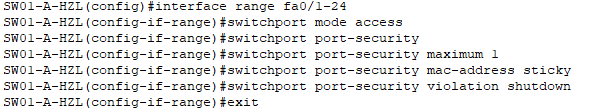
נפעיל את הport security.

נגדיר לו שילמד מקסימום כתובת mac אחת.

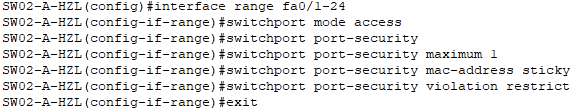
נגדיר שילמד את הכתובת mac בצורה אוטומטית.

נגדיר לו violation mode ל shutdown.

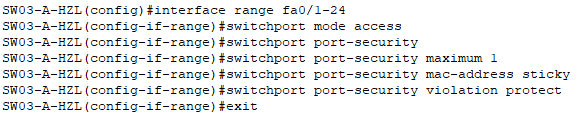
הגדרת port security במתג sw01-A-HZL violation shutdown:



הגדרת port security במתג sw02-A-HZL violation restrict:



הגדרת port security במתג sw03-A-HZL violation protect:



פקודות SHOW

פקודת show port-security:

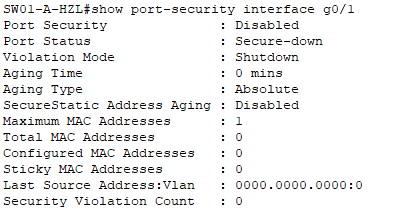
בפקודה זו ניתן לראות טבלה אשר מראה לכל ממשק כמה מקסימום כתובות היא יכולה לשמור, כמה כתובות היא שומרת כרגע, כמה פעמים מכשיר לא מורשה ניסה להתחבר, ובאיזה פעולה יש לנקוט כאשר מכשיר לא מזוהה מתחבר.

Table

Description automatically generated

פקודת show port security interface:

פקודה זו מראה port security על ממשק ספציפי.



# IP Routing

Routing IP הוא תהליך של שליחת packets מhost ברשת אחת לhost ברשת אחרת. תהליך זה נעשה בדרך כלל על ידי נתבים אך גם על ידי מתגים בשכבה 3 במידה ומגדירים את הפקודה ip routing.

# EIGRP

Enhanced Interior Gateway Routing Protocol (EIGRP) הוא פרוטוקול ניתוב דינמי המשמש למציאת הנתיב הטוב ביותר בין מכשירים העובדים על שכבה 3 במודל הOSI. הפרוטוקול פותח על ידי חברת cisco ובמקור עבד רק על המכשירים שלהם, ובהמשך הוא הפך להיות סטנדרט פתוח . EIGRP עובד על פרוטוקול בשכבת הרשת ומשתמש בפרוטוקול מספר 88. הוא משתמש במדדים כדי לגלות את הנתיב הטוב ביותר בין שני התקני שכבה 3 (נתב או מתגי שכבה 3) הפועלים על EIGRP. EIGRP מגיע ממשפחת פרוטוקולי ניתוב הנקראת Distance vector אך יש המחשיבים אותה כהיברידית מפני שיש לה גם תכונות ממשפחת הLink State.

אלגורידם שהEIGRP משתמש בו נקרא: DUAL

ערך הAD של EIGRP Summarization הוא: 5

ערך הAD של EIGRP Internal הוא: 90

ערך הAD של EIGRP External הוא: 170

היתרונות של EIGRP הם:

* שימוש נמוך מאוד במשאבי רשת במהלך פעולה רגילה; רק הודעות HELLO מועברות ברשת יציבה.
* כאשר מתרחש שינוי בטופולוגיה, רק שינויים בטבלת הניתוב מופצים, לא כל טבלת הניתוב; זה מפחית את העומס שפרוטוקול הניתוב עצמו מטיל על הרשת.
* זמני התכנסות מהירים לשינויים בטופולוגית הרשת (במצבים מסוימים ההתכנסות יכולה להיות כמעט מיידית).

## Distance vector vs Link State

|  | **Distance vector** | **Link state** |
| --- | --- | --- |
| פרוטוקולים | RIP, EIGRP | OSPF |
| אלגורידם | DUAL | Dijsktra |
| Network view | מידע על נתיבים שהשכנים מכירים | מידע מלא על הטופולוגיה |
| בחירת נתיב הכי טוב | Based on the least number of hops | על בסיס cost |
| עדכונים | Full routing table | Link state updates |
| תדירות עדכונים | כל פרק זמן קבוע | כאשר יש שינוי בטופולוגיה |
| משאבים | לוקח מעט משאבים | לוקח הרבה משאבים |
| מורכבות | מורכבות נמוכה | מורכבות גבוהה |
| Convergence time | Moderate | Fast |
| עדכונים באמצעות | broadcast | multicast |
| Hierarchical structure | No | Yes |
| Intermediate Nodes | No | Yes |

## יחסי שכנות

על מנת ליצור יחסי שכנות הנתבים משתמשים בהודעות Hello. הודעות Hello אחראיות על מציאת ווידוי הימצאותם אחרי כל כמה זמן. הנתב שולח כל כמה זמן הודעת Hello לנתבים המחוברים אליו ישירות ובמידה ויש התאמה בכל הקריטריונים הוא אמור לקבל הודעת Hello בחזרה.

* הודעותhello נשלחות בMulticast דרך הכתובת 224.0.0.10

קריטריונים כדי ליצור שכנות EIGRP, יש לדאוג לקריטריונים הבאים:

1. ערכי הK חייבים להיות תואמים.
2. מספר הAS חייב להיות תואם.
3. במידה והגדרנו Authentication, הוא חייב להיות תואם.(ID וpassword זהה)
4. הsubnet mask חייב להיות תואם.

* מספר המזהה של הנתב לא חייב להיות ייחודי ברשת(בניגוד לפרוטוקולים אחרים). אך הוא לא יוכל לקבל עדכונים מרשת חיצונית.

Timers:

Hello timer - המרווח זמן שבו EIGRP שולח הודעת HELLO בממשק. זה 5 שניות כברירת מחדל.

Hold timer - המרווח זמן שבו השכן יוכרז כלא פעיל אם לא יוכל לשלוח את חבילת HELLO. זה 15 שניות כברירת מחדל.

נ.ב: נהוג שזמן ה Dead timerחייב להיות גדול פי 3 מזמן ה Hello timer.

**עדכון טבלאות הניתוב**

* **מתרחש רק כאשר יש שינוי בטופולוגיה.**
* **מחיקת נתיבים אשר עוברים דרך נתב שכן שעבר את זמן ה** Hold timerמבלי לענות.
* כאשר נתיב לרשת מסוימת נקטע ואין נתיב גיבוי, הנתיב יכנס למצב Active. במצב הזה הנתב ישלח חבילת Quary לכל השכנים שלו כדי לשאול האם מישהו מכיר נתיב אחר לרשת שהוא מחפש. השכן יענה באמצעות חבילת Reply ויתן לו מסלול חלופי במידה ויש.
* מידע על ניתוב עובר באמצעות חבילת הUpdate. כאשר הנתב מקבל את חבילת ה הUpdate הוא עונה בחזרה עם חבילת Ack לנתב ממנו קיבל את ה הUpdate.

**כתובות multicast:**

**הכתובת 224.0.0.10 משמשת את EIGRP על מנת לאפשר לנתבי EIGRP לשתף מידע ניתוב עם נתבי EIGRP אחרים בקטע רשת.**

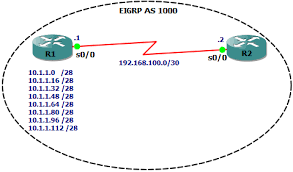
**EIGRP Summarization**

**סכימת רשתות הוא תהליך דרכו ניתן להכליל מספר רשתות תקשורת בטבלת הניתוב לרשת תקשורת אחת גדולה יותר.**

**היתרון של סכימת רשתות הוא צמצום טבלאות ניתוב אשר מקלות על משאבי רשת ותיעדוף מסלול על ידי הAD הנמוך יותר שיש לו.**

**החיסרון בסכימה הוא במידה ויש רשת שלא נמצאת בתוך הסכימה הנתב עלול לפספס אותה.**

**ניתן להגדיר סכימה בצורה ידנית או אוטומתית על נתבי cisco.**

****

## Feasible Distance vs Advertised Distance

Advertised Distance (AD) – הmetric של הנתיב כפי שהנתב השכן פרסם אותו.

Feasible Distance (FD) – הmetric של הנתיב הכולל גם את הmetric של הנתב הנוכחי מהשכן דרכו הנתיב עובר.

**Successor vs Feasible Successor**

* Successor – הנתיב עם הmetric הטוב ביותר ליעד. הנתיב הטוב ביותר הוא הנתיב שהFD שלו הוא הכי קטן.בברירת מחדל יש מקסימום 4 נתיבי successor. ניתן לפתוח עד 32 נתיבי successors.
* Feasible Successor – נתיב גיבוי אם ה Successorנופל. ל Feasible Successor חייב להיות AD קטן יותר מהFD של ה Successor. אין הגבלה לכמות נתיבי הfeasible successor .

במידה ולא נמצא נתיב גיבוי המערכת תיכנס למצב של לולאה.

## Variance command

* פקודה היוצרת load balancing.
* על מנת לאפשר מספר נתיבי successors נכפיל את הFD של הsuccessor בערך הvariance.
* ערך ברירת המחדל של הvariance הוא 1.
* כל הנתיבי feasible successors עם FD שווה או נמוך יותר מתוצאת המכפלה של הsuccessor עם הvariance יהפוך להיות successor בנוסף.

## ערכי K

חישוב הmetric ב EIGRP משתמש בשילוב של 5 משתנים, אך רק 2 משמשים כברירת מחדל (K1 ו-K3). הערכים המשולבים הם:

K1 (bandwidth) **= 1**  
K2 (load)  = 0  
K3 (delay)  = 1  
K4 (reliability)  = 0  
K5 (MTU - maximum transmit unit) = 0

רוחב הפס, העומס, ההשהיה, האמינות, ה-MTU הנמוכים ביותר לאורך הנתיב בין המקור ליעד נלקח בחשבון בmetric על מנת לחשב את העלות.

נוסחה לחישוב הmetric:

metric = ([K1 \* bandwidth + (K2 \* bandwidth) / (256 - load) + K3 \* delay] \* [K5 / (reliability + K4)]) \* 256

passive-interface

הגדרת ממשק פסיבי מונעת מהממשק לשלוח עדכוני ניתוב

**הגדרת EIGRP**

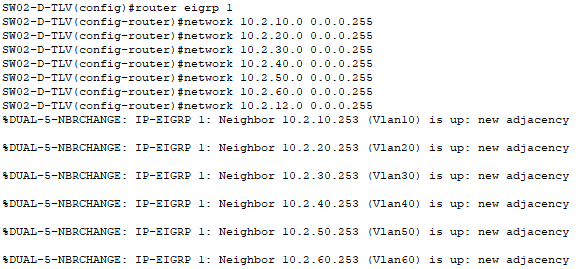
הגדרת פרוטוקול EIGRP במתג שכבה 3 בסניף השני:

תחילה נגדיר את כל הממשקים דרכם אנו רוצים לבצע ניתוב כממשקים בשכבה 3 על ידי הפקודה no switchport.

נגדיר כתובת Ip לכל אחד מהממשקים.

ניכנס למצב קונפיגורצית EIGRP בAS 80.

נגדיר לכל הרשתות המחוברות אלינו פיזית אשר נרצה לעשות ניתוב דרכן את הפקודה network אשר מבצעת יחסי שכנות עם כל הנתבים באותה רשת.



**פקודות SHOW**

show ip eigrp interfaces:

מראה סטטיסטיקה של כל הממשקים העובדים עם EIGRP

Interface – שם הממשק

Peers – כמות השכנים שהנתב מכיר בממשק

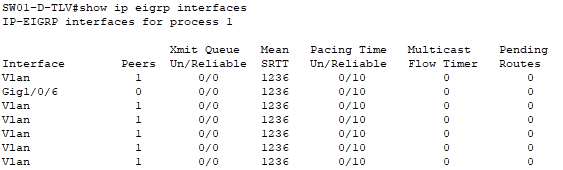
Xmit Queue Un/Reliable -

Mean SRTT-

Pacing Time Un/Reliable -

Multicast Flow Timer -

Pending Routes -



show ip eigrp neighbors:

בפקודה זו ניתן לראות טבלה של כל השכנים אשר הנתב מכיר ומידע נוסף עליהם כמו כתובת הIP שלהם, מאיזה ממשק הם הגיעו, מספר השניות שנותר להם לעדכן את נוכחותם.

Address – כתובת הIP של השכן.

Interface – הממשק המקומי דרכו מחובר השכן.

Hold – ספירה לאחור לפני מחיקת הנתב מהטבלה.

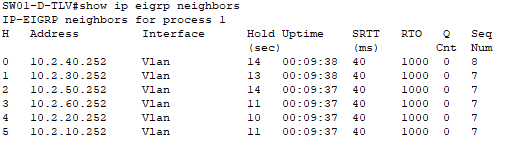
Up time – כמה זמן השכן בטבלה.

SRTT – הזמן שלקח לחבילת EIGRP להגיע לשכן ולחזור בחזרה.

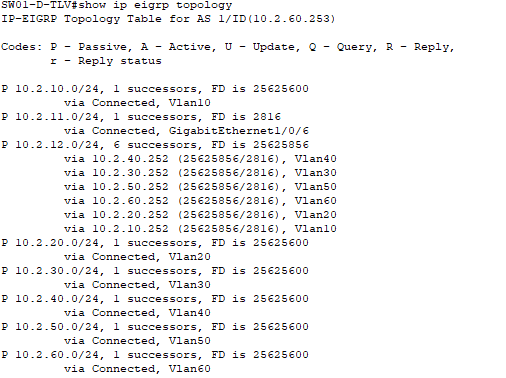
RTO – הזמן שהנתב יחכה לפני שישלח הודעת unicast במקרה ולא התקבלה חבילה מהשכן.

Q count - כמות החבילות שממתינות לשליחה.

Seq num – המספר הסידורי של החבילה האחרונה שהתקבלה.



בפקודה זו אנו רואים את כל הרשתות שהeigrp מכיר בין אם הן מחוברות ישירות או דרך שכן. לכל נתיב יש אות אשר אומרת לנו את המצב בו נמצא הנתיב. כשרואים את האות P הנתיב נמצא במצב Passive מה שאומר שהוא תקין ולא נדרש לחפש דרך להגיע ליעד. כאשר רואים את האות A הנתב מחפש בצורה אקטיבית דרך להגיע ליעד ולכן הנתיב אינו זמין. במידה ויש מספר נתיבים שאפשר לעבור דרכן לרשת מסוימת יהיה אפשר לראות אותם כfeasible successors. המספר הראשון בסוגריים הוא הFD והמספר השני הוא הAD.



באמצעות פקודה זו ניתן לראות מידע סטטיסטי על packets הרלוונטיים לeigrp.

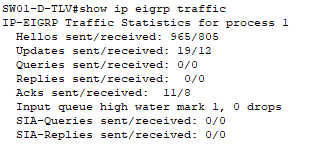
כמות חבילות הHELLO שנשלחו והתקבלו

כמות חבירות הUPDATES שנשלחו והתקבלו

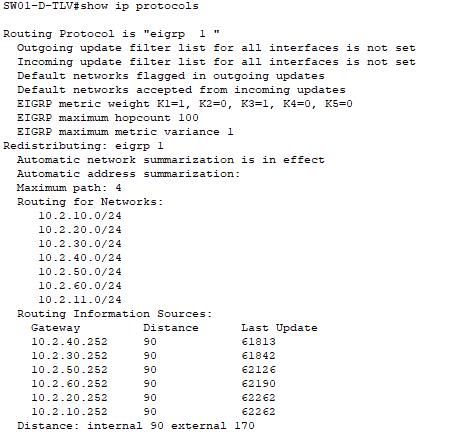
כמות חבירות QUERIES שנשלחו והתקבלו

כמות חבירות REPLIES שנשלחו והתקבלו

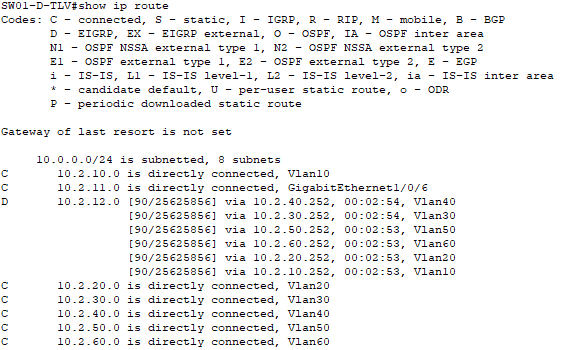
כמות חבירות ACK שנשלחו והתקבלו



Show ip protocols:



Show ip route:



# OSPF

Open Shortest Path First (OSPF) הוא פרוטוקול Link State המשמש למציאת הנתיב הטוב ביותר בין המקור לנתב היעד תוך שימוש ב-Shortest Path First שלו. OSPF פותח על ידי Internet Engineering Task Force (IETF) כאחד מפרוטוקול Interior Gateway (IGP), כלומר, הפרוטוקול שמטרתו להעביר את החבילה בתוך מערכת אוטונומית גדולה או תחום ניתוב. זהו פרוטוקול שכבת רשת שעובד על פרוטוקול מספר 89 ומשתמש בערך AD 110 . OSPF משתמש בכתובת multicast 224.0.0.5 לתקשורת רגילה ו-224.0.0.6 לעדכון לנתב ייעודי (DR)/נתב ייעודי גיבוי (BDR).

ערך הAD שלOSPF הוא: 110

OFPF הוא:

* OSPF הוא פרוטוקול שער פנימי (IGP).
* הוא פועל בתוך תחום ניתוב יחיד, כגון מערכת אוטונומית (AS).
* הוא משתמש במושג שנקרא areas, כדי לייעל את תעבורת הרשת ולפשט את הניהול.
* הוא פועל על פרוטוקול IP אך אינו משתמש בפרוטוקול תחבורה (כגון TCP או UDP) כדי להקיף את הנתונים שלו.

אלגורידם שהEIGRP משתמש בו נקרא: Dijsktra

היתרונות של OSPF הם:

* זה מבוסס על תקן פתוח. זה יכול לפעול ברוב הנתבים.
* הוא משתמש באלגוריתם SPF כדי לספק טופולוגיה נטולת לולאות.
* זה תומך בשתי הגרסאות של IP. פרוטוקול OSPFv2 תומך ב-IPv4 ו-OSPFv3 תומך ב-IPv6.
* הוא תומך באיזון עומסים עם מסלולים בעלות שווה לאותו יעד.

החסרונות של OSPF הם:

* הוא זקוק למידע רב כדי לחשב את המסלול הטוב ביותר עבור כל יעד. כדי לאחסן מידע זה, OSPF צורך יותר זיכרון מאשר פרוטוקולי ניתוב אחרים.
* כדי לחשב את המסלול הטוב ביותר, הוא מריץ את אלגוריתם SPF הדורש עיבוד CPU רב.
* זה מורכב להגדרה וקשה לפתרון בעיות. ברשת גדולה, רק מנהלי רשת מנוסים יכולים להגדיר אותה.

## סיכום של תהליך OSPF

* + - 1. כל נתב לומד על הרשתות שמחוברות אליו ישירות
      2. כל נתב אחראי לשלוח חבירות HELLO לשכניו שמחוברים ישירות.
      3. כל נתב בונה חבילת LSP שמכילה את מצב (State) של כל ה Links
      4. כל נתב מציף Floods את חבילות הLSP לכל שכניו, אשר אוגרים את כל החבילות שהתקבלו בתוך הDatabase
      5. כל נתב משתמש בDatabase , בכדי לבנות "מפה כוללת" של טופולוגית הרשת , ומחשב את המסלול הטוב ביותר ליעדים.

שימוש בOSPF:

בדרך כלל, OSPF משמש ברשתות ארגוניות גדולות המשתמשות בציוד ניתוב של ספקים שונים. OSPF משמש גם בחברות שיש להן מדיניות להשתמש בפרוטוקול סטנדרטי פתוח לניתוב המעניק להן גמישות כאשר הן צריכות להחליף נתב קיים או להוסיף נתב חדש.

## יחסי שכנות

הודעות Hello של OSPF משמשות כדי לאפשר לנתב לגלות נתבים סמוכים אחרים בקישורים וברשתות המקומיות שלו. ההודעות מייצרות קשרים בין התקנים שכנים (הנקראים adjacencies) ומעבירות פרמטרים מרכזיים לגבי אופן השימוש ב-OSPF במערכת או בAS. בזמן פעולה רגילה, נתבים שולחים הודעות hello לשכניהם במרווחי זמן קבועים (hello interval); אם נתב יפסיק לקבל הודעות שלום מהשכן, לאחר תקופה מוגדרת (dead interval), הנתב יניח שהשכן ירד.

תנאים ליצירת יחסי שכנות:

Hello ו- Hold זהים

Area זהה

Password

subnet

שלבים עד ליצירת יחסי שכנות:

* Down
* Init
* Two-way
* Extart – מצב בו נבחר הDR והBDR
* Exchange – הנתבים מחליפים ביניהם description packet
* Loading – מעבירים ביניהם request packet וDBD
* Full - יחסי שכנות מלאים

ניתן למצוא את המצב הנוכחי באמצעות פקודת show ip ospf neighbor

timers:

Hello timer - המרווח זמן שבו EIGRP שולח הודעת HELLO בממשק. זה 10 שניות כברירת מחדל.

dead timer - המרווח זמן שבו השכן יוכרז כלא פעיל אם לא יוכל לשלוח את חבילת HELLO. זה 30 שניות כברירת מחדל.

נ.ב: נהוג שזמן ה Dead timerחייב להיות גדול פי 3 מזמן ה Hello timer.

מושגים בOSPF:

* Router I’d - זוהי כתובת ה-IP הפעילה הגבוהה ביותר הקיימת בנתב. ראשית, הכתובת הloopback ביותר נחשבת. אם לא מוגדרת loopback, כתובת ה-IP הפעילה הגבוהה ביותר בממשק של הנתב נחשבת.
* Router priority - זהו ערך של 8 סיביות המוקצה לנתב המפעיל OSPF, המשמש לבחירת DR ו-BDR ברשת שידור.
* Designated Router (DR) - זה נבחר כדי למזער את מספר הסמיכות שנוצרו. DR מפיץ את ה-LSAs לכל שאר הנתבים. DR נבחר ברשת שידור שאליה כל שאר הנתבים חולקים את ה-DBD שלהם. ברשת שידור, הנתב מבקש עדכון ל-DR, ו-DR יגיב לבקשה זו בעדכון.
* Backup Designated Router (BDR) - BDR הוא גיבוי ל-DR ברשת שידור. כאשר DR יורד, BDR הופך ל-DR ומבצע את תפקידיו.
* DBD –

Link state packets:

* Link state request (LSR) - הודעות LSR משמשות נתב אחד כדי לבקש מידע מעודכן על חלק מה-LSDB מנתב אחר. ההודעה מציינת את הקישור/ים שעבורם המכשיר המבקש רוצה מידע עדכני יותר.
* Link state update (LSU) - הודעות LSU מכילות מידע מעודכן לגבי מצבם של קישורים מסוימים ב-LSDB. הם נשלחים בתגובה להודעת LSR, וגם משודרים או משודרים בmulticast על ידי נתבים על בסיס קבוע. התוכן שלהם משמש לעדכון המידע ב-LSDBs של נתבים שמקבלים אותם.
* Link state acknowledgment (LSAck) - הודעות LSAck מספקות מהימנות לתהליך Link state update, על ידי אישור מפורש על קבלת הודעת LSU.

תהליך בחירת הDR - בחירות DR ו-BDR מתקיימות ברשת broadcast. להלן הקריטריונים לבחירות:

1. נתב בעל Router priority הגבוהה ביותר יוכרז כ-DR.
2. אם יש שוויון בעדיפות הנתב אז הנתב עם ה Router I’d הגבוה יותר יוכרז כ-DR.

Path Cost

OSPF משתמש Path Cost כמדד הניתוב הבסיסי שלו, שהוגדר על ידי התקן שלא ישווה לשום ערך סטנדרטי כגון מהירות, כך שמעצב הרשת יכול לבחור מדד חשוב לתכנון. בפועל, זה נקבע על ידי השוואת מהירות הממשק לרוחב פס ייחוס עבור תהליך OSPF. העלות נקבעת על ידי חלוקת רוחב הפס הייחוס במהירות הממשק (אם כי ניתן לעקוף את העלות עבור כל ממשק באופן ידני). אם רוחב פס ייחוס מוגדר ל-'10000', קישור של 10 ג'יגה-ביט/שניה יהיה בעל עלות של 1. כל מהירויות הנמוכות מ-1 מעוגלות כלפי מעלה ל-1.

**הגדרת OSPF**

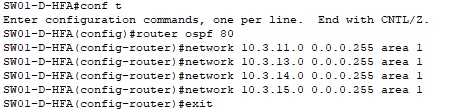
הגדרת פרוטוקול OSPF במתג שכבה 3 בסניף השלישי:

תחילה נגדיר את כל הממשקים דרכם אנו רוצים לבצע ניתוב כממשקים בשכבה 3 על ידי הפקודה no switchport.

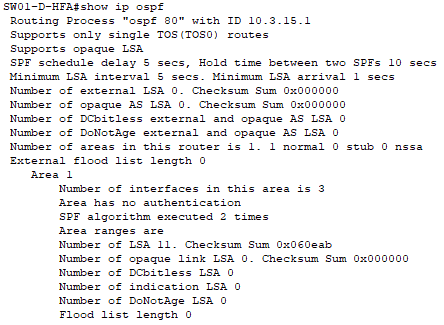
נגדיר כתובת Ip לכל אחד מהממשקים.

ניכנס למצב קונפיגורצית EIGRP בprocces ID 80.

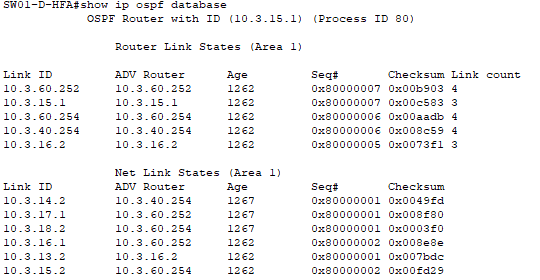
נגדיר לכל הרשתות המחוברות אלינו פיזית אשר נרצה לעשות ניתוב דרכן את הפקודה network אשר מבצעת יחסי שכנות עם כל הנתבים באותה רשת.



show ip ospf:



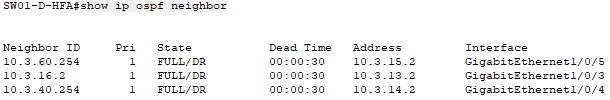
show ip ospf database:

****

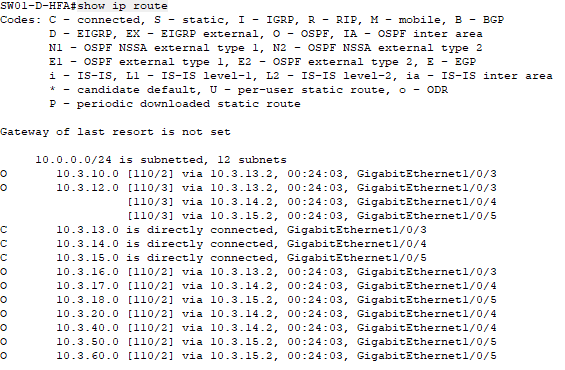
show ip ospf interface :



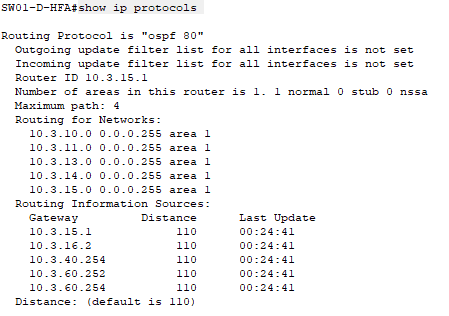
show ip ospf neighbor:



show ip route:



show ip protocols:



# ביבליוגרפיה

<https://www.cisco.com/c/en/us/support/docs/lan-switching/vtp/10558-21.html>

<https://en.wikipedia.org/wiki/VLAN>

<https://en.wikipedia.org/wiki/Trunking>

<https://networklessons.com/cisco/ccie-routing-switching/hsrp-hot-standby-routing-protocol>

<https://www.guru99.com/stp-spanning-tree-protocol-examples.html>

<https://www.networkacademy.io/ccna/ethernet/router-on-a-stick>

<https://en.wikipedia.org/wiki/Dynamic_Host_Configuration_Protocol>

<https://www.cisco.com/c/en/us/support/docs/ip/enhanced-interior-gateway-routing-protocol-eigrp/16406-eigrp-toc.html>

<https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_ospf/configuration/xe-16/iro-xe-16-book/iro-cfg.html>

# איתור תקלות ופתרונן

במהלך קונפיגורציה פרוטוקול STP בסניף הרצליה נתקלתי בבעיה חיבוריות בכמה vlans ספציפיים, שלא יכלו לתקשר אחד עם השני או עם מכשירים בתוך אותה הרשת. הבעיה נבעה מהגדרה ידנית של priority על המתגים שאמורים להיות מוגדרים כהroot bridgh והגיבוי שלו. כאשר אחד המתגים האחרים נתן לעצמו priority נמוך יותר בצורה אוטומתית זה שיבש למעשה את כל התעבורה. הפתרון לבעיה הזו הוא להגדיר את הroot bridgh והגיבוי שלו באמצעות הפקודות root primary ו root secondary.