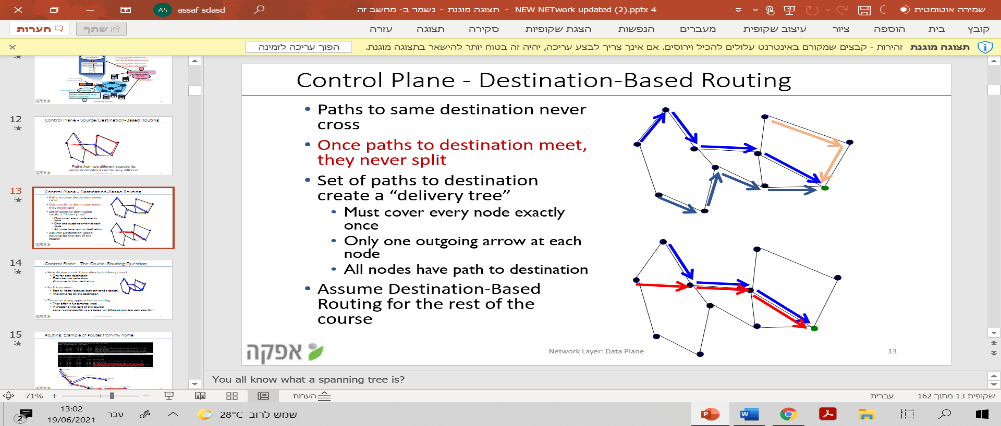
Ip

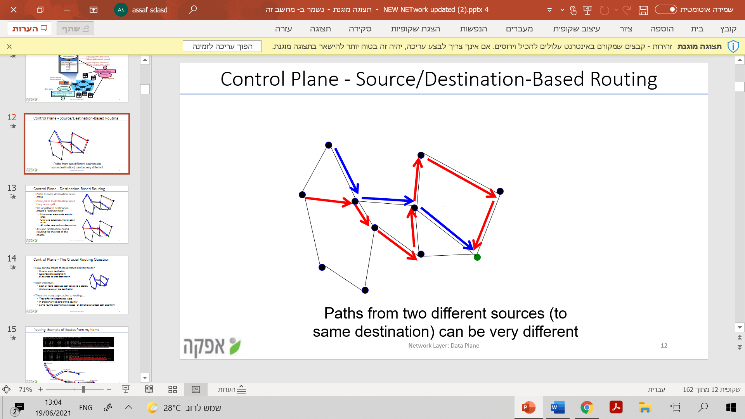
מטוס בקרה – control plane)) תהליך גלובלי ברחבי הרשת קובע כיצד מנותבת מנות בין נתבים לאורך נתיב קצה ממארח המקור למארח היעד

מישור נתונים – (- (forwarding functionפונקציה מקומית לכל נתב שמשתמש בטבלאות העברה לבחירת יציאת הפלט וכדי לקבוע כיצד מנות המגיעות ליציאת קלט הנתב מועברות ליציאת הפלט של הנתב.

ניתוב -routing: קובע מסלול מראש שנלקח על ידי מנות ממקור ליעד, בהתבסס על הרשת. משתמש באלגוריתמי ניתוב. הֵם: הם איטיים ורצים ברקע, לדעת מראש את הטופולוגיה של הרשת, עדכון את טבלאות ההעברה.

העברה- forwarding: העברת חבילות מקלט הנתב לפלט הנתב המתאים. מהיר, פועל על כל מנה ומשתמש בטבלאות ההעברה.

Control Plane - Destination-Based Routing-

נתיבים לאותו יעד לעולם לא חוצים, ברגע שהנתיבים ליעד נפגשים, הם מעולם לא התפצלו, חייב לכסות כל צומת בדיוק פעם אחת, לכל הצמתים יש נתיב ליעד.

Control Plane - Source/Destination-Based Routing-

נתיבים משני מקורות שונים (לאותו יעד) יכולים להיות שונים מאוד

Control Plane - The Crucial Routing Question-

SDN- הרעיון ב-SDN הוא להוציא את ה-control plane שמטפל ב-routing מהנתבים, ולהעביר את הטיפול בניתוב הגלובלי לענן. הנתבים יקבלו את טבלאות הניתוב מהענן.

סיבות – 1. :Vertical integration: האינטגרציה של הפונקציונליות של L3 היא אנכית, כך שכל חברה ממשת את ה-data plane ואת ה-control plane בצורה שונה.   
2. Propriety implementation: הנתבים מיוצרים ע"י חברות רבות, ועל מנת שהציוד יעבוד בשטח, דרוש מימוש מאוד קפדני של הסטנדרטים בתוכנה  
3. Costly routing hardware: התוכנה מאוד מסובכת ודורשת מעבדים יקרים בתוך הנתבים, וזה מיקר ומסבך את הציוד.

טבלאות ניתוב- Router forwarding table-look up- cash של הנתב.

Ipv4-2^32 מס כתובות הip שנמצאות תחת ipv4

גודל של כתובת הוא 32-bit , header - version(4 bits)  
header length (4 bits), total length (16 bits),  
 two ip address (32 and 32 bits)- source and destination, protocol (8 bits),  
type of service (8 bit), ttl( 8 bits) , Checksum (16 bits) , Fragmentation

Checksum – מחושב בכל נתב מחדש

Longest Prefix Match (LPM)-

הנחה בסיסית: טבלאות העברה מתעדכנות על ידי control plane  
חבילה מגיעה -> שכבת L3 בנתב מחלצת את כתובת ה- IP של 32 סיביות מכותרת החבילה -> שכבת L3 בנתב מחפשת בטבלת ההעברה L3 אחר התאמה לקידומת הארוכה ביותר (LPM) -> קוראת מהטבלה את פלט היציאה -> הנתב מעביר את החבילה ליציאה המתאימה והחבילה עוברת  
**מציאת כתובת ip-** מורכב משני חלקים prefix (k) ומספרים חופשיים host part))  
2^host -2 נותן את מספר הכתובות האפשרי (לבדוק אם כולל בתוכו עוד כתובות אחרות בשאלה).   
**prefix- subnet – network part**  
**אם לכתובת אחת יש prefix קטן יותר מכתובת אחרת אבל prefix זהה אז הכתובת עם הprefix הקטן מכילה בתוכה את הכתובת השניה.**  
**כשמחפשים ערך טבלת העברה לכתובת יעד נתונה, השתמש בקידומת הכתובת הארוכה ביותר שתואמת את כתובת היעד.**

**עבודה עם כתובת ip- מעבירים לבינארי, מחלקים ל8 ביטים עם רווח ביניהם. 121.12.13.121/23 – 23 מציין prefix 32-23, ציין את מספר הביטים החופשיים**  
**מציאת מסכת רשת network mask- שמים 1 במקום ביטים prefix**

Subnet-

נקרא גם רשת מקומית LAN , מכשירים תחת אותה רשת משנה של כתובות ip יכולים להגיע אחת לשני מבלי לעבור דרך נתב.

מציאת subnet לשים 0 מימין לביטים prefix

כתובת broadcast- הכל 1 , משמש רק ככתובת יעד של חבילות . פירושו: כל המארחים ברשת המשנה שצוינו בקידומת הכתובת  
בשביל למצוא את הכתובת הזו לשים מצד ימין לprefix 1 ואז זו תהיה כתובת broadcast של אותה רשת.  
Switch-  
מעביר מנות בין מארחים (hosts) ברשת מקומית (LAN) העברה בשכבת L2 (באמצעות כתובת L2). כל המארחים ברשת LAN שייכים לאותה רשת משנה IP.   
Routers-   
מעביר מנות בין מארחים ברשת LAN שונה (חבילות עוברות מרשת משנה אחת לאחרת) העברה בשכבת L3  
הוא מקבל חבילה , קורא את הכתובת מip את הport המתאים ומקדם את החבילה לתור של port.

Scheduling-  
*FIFO*- first in first out -  
 fixed priority -שולחים את החבילה בעלת העדיפות הגבוהה ביותר כל פעם. תלוי ביעד, מקור, או port  
*Round Robin (RR) scheduling*- שולחים כל פעם חבילה שלמה של תור. זא אם יש כמה חבילות מכמה תורים והסיבוב נמצא בתור מסוים אז שולחים את כל החבילות של אותו התור.  
CIDR - חלק רשת המשנה של הכתובת באורך שרירותי . פורמט כתובת: a.b.c.d / k, כאשר k הוא # סיביות בחלק המשנה של הכתובת

DHTP-Dynamic Host Configuration Protocol   
מטרה: לאפשר למארח להשיג באופן דינמי את כתובת ה- IP שלו משרת הרשת כשהוא מצטרף לרשת. יכול לחדש את שימוש באותה כתובת ip . יתרון הוא שהכל מתנהל באופן עצמאי ללא התערבות של מנהל רשת. רק למארחים פעילים יש כל הזמן כתובות ip פעילות.  
DHTP מחזיר ip, צומת ראשונה של נתב, ip of DNS server, מסכת רשת  
מאחר שולח לשרת DHSP בקשה לip “DHCP discover” . שרתי DHTP שולחים “DHCP offer”. יותר משרת אחד עלול להגיב. המארח מקבל כתובת ip ו DHTP  
שולח ack  
שרת DHCP: מכשיר ברשת המריץ את שירות DCHP המחזיק כתובות IP ומידע תצורה קשור. לקוח DHCP: נקודת הקצה המקבלת פרטי תצורה משרת DHCP. מאגר כתובות IP: טווח הכתובות הזמין ללקוחות DHCP. כתובות בדרך כלל מחולקות ברצף מהנמוך לגבוה ביותר.  
חכירה: משך הזמן שבו לקוח DHCP מחזיק את פרטי כתובת ה- IP. כאשר חוזה השכירות פג, על הלקוח לחדש אותו.  
ממסר DHCP: נתב או מארח שמאזין להודעות לקוח המשודרות ברשת זו ואז מעביר אותם לשרת מוגדר. לאחר מכן השרת שולח תגובות חזרה לסוכן הממסר שמעביר אותן ללקוח. ניתן להשתמש בזה לריכוז שרתי DHCP במקום שיהיה שרת בכל רשת משנה.  
 **NAT-network address translation –**  
מספק כתובות ip בתוך רשמת מקומית.  
לא נתמך ע"י האינטרנט. הרשת המקומית משתמשת בכתובת IP אחת בלבד מבחינת העולם החיצוני. הרשת הפנימית נקראת WAN,   
יתרונות – כתובת ip אחת לכל המכשירים הפנימיים. יכול לשנות כתובות של התקנים ברשת המקומית מבלי ליידע את העולם החיצון. מכשירים שנמצאים ברשת המקומית לא בהכרח ניתנים לאיתור והם מאובטחים.

Ipv6- 128 bit

שוני לעומת ipv4- אין checksum, לכן IPv6 דורש בדיקת בדיקות בכותרות UDP. כאשר TCP פועל מעל IPv6, השיטה המשמשת לחישוב בדיקת הבדיקה משתנה בהתאם ל- RFC 2460. עומס בדיקת בדיקה. כדי להקל על מערכת ההפעלה משימוש במחזורי מעבד יקרים לחישוב סכום הבדיקה: סיוע בחומרה לחישוב אוטומטי של בדיקת הבדיקה במתאם הרשת לפני העברה לרשת או בקבלה מהרשת לצורך אימות. פונקציות ניהול קבוצות מרובות שידורים

Header בipv6 יותר גדול. בipv4 צריך להגדיר באופן ידני או להשתמש ב DHTP על מנת להגדיר רשתות או תצורת רשת. וipv6 נעשה באופן אוטומטי. אין שימוש במסכת רשת בipv6. אין שימוש ב NAT בגלל שימוש בהרבה מאוד כתובות ip. מאפשר שליחת packets יותר גדולים. בipv4 האבטחה תלויה ביישומים – applications ועוד בipv6 קיים פרוטוקול ייחודי לזה IPSec