

רשתות מחשבים

פרק 4ג' - שכבת האפליקציה, פרוטוקול DNS

ברק גונן

מבוסס על ספר הלימוד "רשתות מחשבים" מאת

עומר רוזנבוים

מטרות הפרק



- ▶ בסיום הפרק נדע:
 - מה תפקיד פרוטוקול DNS?
 - לשם מה צריך DNS?
 - מהי ההיררכיה של כתובות דפי האינטרנט?
 - מהו TLD, ואילו סוגים של TLD ישנם?
 - מה ההבדל בין iterative query ו-recursive query?
 - מה זה reverse mapping?
- ▶ נתרגל הסנפת וניתוח פקטות DNS

Domain Names

► הצורך ב- Domain Names: יותר קל לזכור שמות
מאשר כתובות IP



מה תפקיד פרוטוקול DNS?

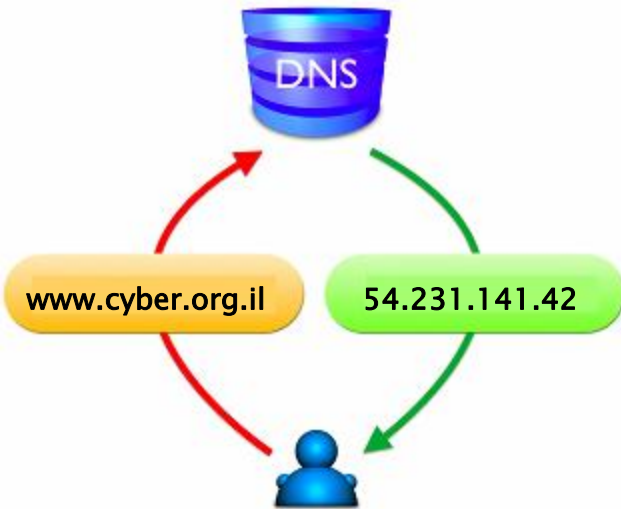
Domain Name System -DNS ▶
בלי DNS היה בלתי אפשרי לקיים את
רשת האינטרנט

למה? ▶

◦ DNS מבצע תרגום (מיפוי) של מ-URL
לכתובת IP

◦ DNS יכול לבצע גם תרגום של כתובת IP
ל-URL

▶ ... נו, אז מה?



מה היה קורה בלי DNS?



- ▶ בלי DNS, לקוח שהיה רוצה לגשת למשאב כלשהו באינטרנט, היה צריך לדעת את כתובת ה-IP שלו
- ▶ עם מאות מיליוני דפי אינטרנט- זו משימה בלתי אפשרית
 - איך מעדכנים ומפיצים לכל העולם את כתובת כל הדפים?
- ▶ נוסף על כך, כתובת IP עשויה להשתנות. היינו צריכים לעקוב גם אחרי השינויים...

Domain Name Server

- ▶ נפתור את הבעיה על ידי שרת, שיחזיק את כל כתובות ה-IP שיש בעולם: Domain Name Server
- ▶ לקוח יפנה לשרת, לדוגמה: מה כתובת ה-IP של `www.example.com`? והשרת יחזיר לו כתובת IP מושלם?
- חישבו על בעיות במודל זה 😊



Domain Name Server - בעיות ופתרונות

- ▶ בעיה: אם השרת מתנתק, האינטרנט בכל העולם נופל
- ▶ פתרון: שרת גיבוי



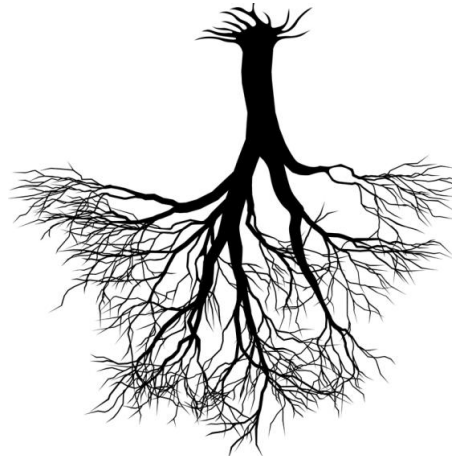
Domain Name Server - בעיות ופתרונות

- ▶ בעיה: אם כל הלקוחות בעולם יפנו לשרת אחד, זה ייצור עליו עומס עצום
- ▶ פתרון: חלוקת העומס בין מספר שרתים



Domain Name Server - בעיות ופתרונות

- ▶ בעיה: אם יש מיליוני רשומות, חיפוש ייקח זמן רב
- ▶ פתרון: היררכיה של שמות דומיין-
 - נחלק את ה- Domain Names לרמות שונות ("Zones")
 - כל Domain Name Server יכיר רק את כתובות ה-IP של השרתים ב-Zone מתחתיו
 - הקשרים בין השרתים נראים כמו עץ (הפוך 😊)
 - מיד נבין איך זה עובד



היררכיה של Domain Names

▶ בראש העץ יש Root- שרת שהכתובת שלו היא '.' (נקודה)
◦ שייך לגוף בינלאומי

◦ ICANN – Internet Corporation of Assigned Numbers & Names

▶ מתחת ל-root יש Top Level Domains -TLDs – משני סוגים:

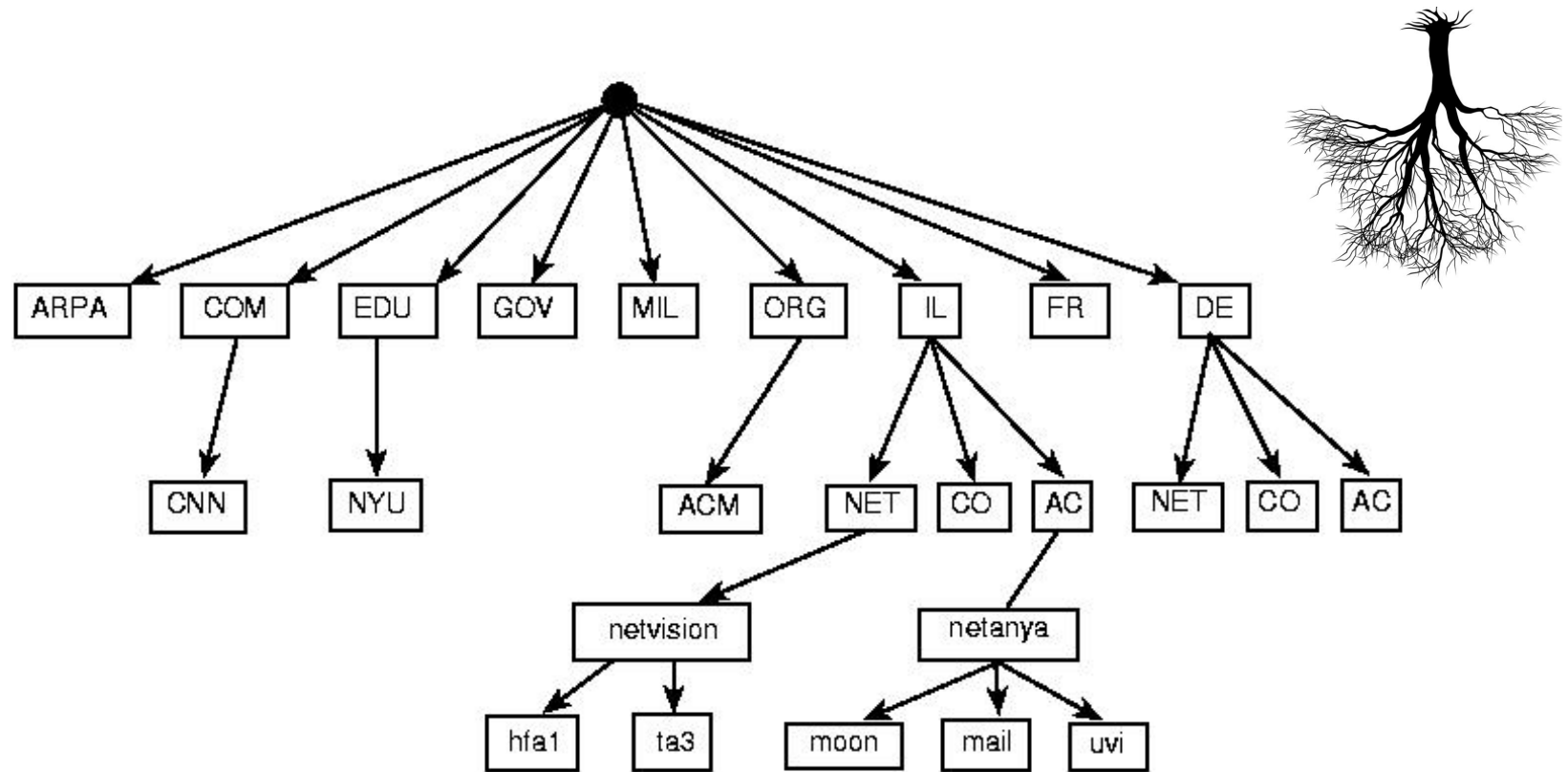
◦ Country Code TLD- .il, .us, .uk, .ru...

◦ Generic TLD – .com, .edu, .gov, .org, .net ...

▶ ה-TLDs מחולקים ל-zones

▶ ה-zones מחולקים לרמות שונות, עד שמגיעים לשרת שאחראי על משאב רשת יחיד

היררכיה של Domain Names - המחשה



מקור: <http://mars.netanya.ac.il/~unesco/cdrom/booklet/HTML/NETWORKING/node100.html>

שאלה למחשבה

▶ האם יש הבדל בין:

▶ www.cyber.org.il

▶ www.cyber.il.org

▶ האם שתי הכתובות ממופות לאותו IP?

▶ תשובה: בכתובות אינטרנט, סדר הכתיבה קובע.

◦ בכתובת הראשונה, ה-TLD הוא il ותחתיו נמצא zone בשם org

◦ בכתובת השניה, ה-TLD הוא org ותחתיו נמצא zone בשם il

היררכיה של Domain Names - המשך

- ▶ כשקוראים URL, ההיררכיה היא מימין לשמאל
 - את ה-root, שסימנו נקודה, לא כותבים
 - נקודה מפרידה בין שרתים ברמות שונות
 - משמאל נכתב סוג השירות (ברירת המחדל www- דפי אינטרנט)
- ▶ לדוגמה:

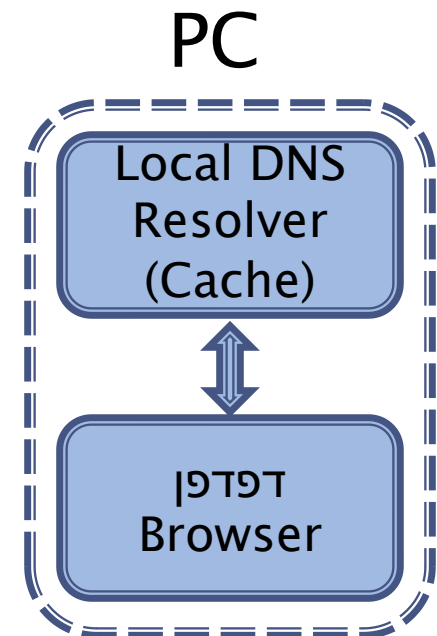
www.google.co.il →
root
il
co
google

היררכיה של Domain Names - סיכום

- ▶ השרת root מכיר רק את כתובות ה-IP של שרתי ה-TLD
- ▶ כל שרת TLD מכיר רק את ה-IP של השרתים שברמה אחת מתחתיו
 - לדוגמה: www.cyber.org.il השרת של il מכיר את כתובת השרת org.il אך לא את cyber.org.il
- ▶ יתרון: כל שרת צריך להחזיק רק רשימה מצומצמת של שרתים, החיפוש ברשומות הוא מהיר יותר
- ▶ נדרש מנגנון (פרוטוקול) למציאת כתובת ה-IP

אז איך עובד פרוטוקול DNS?

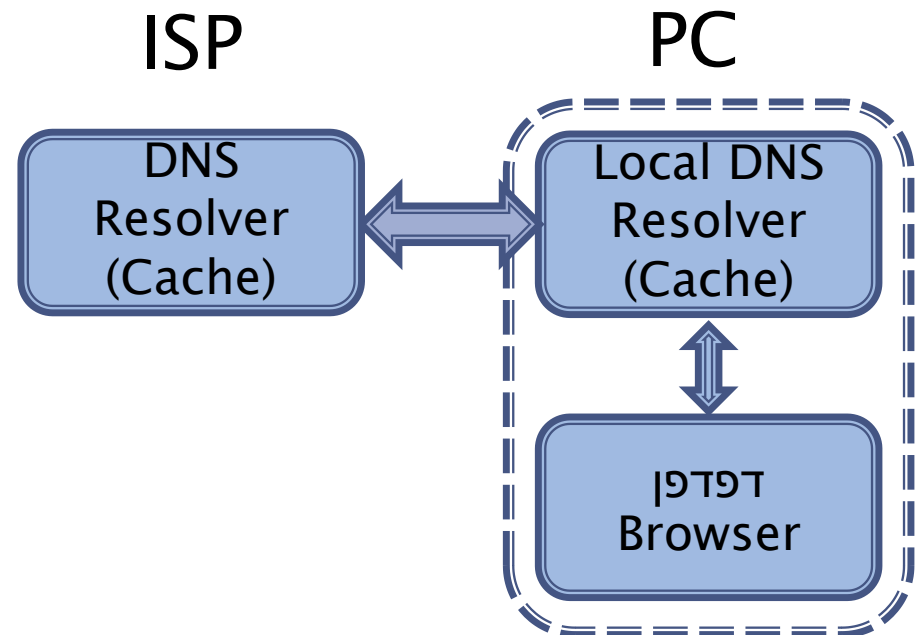
- ▶ שלב 1: הדפדפן שלנו מבקש URL כלשהו
 - המחשב בודק אם ה-URL נמצא ב-cache (גלשנו לשם לאחרונה)
 - אם כן- כתובת ה-IP נמסרת לדפדפן והתהליך מסתיים



אז איך עובד פרוטוקול DNS?

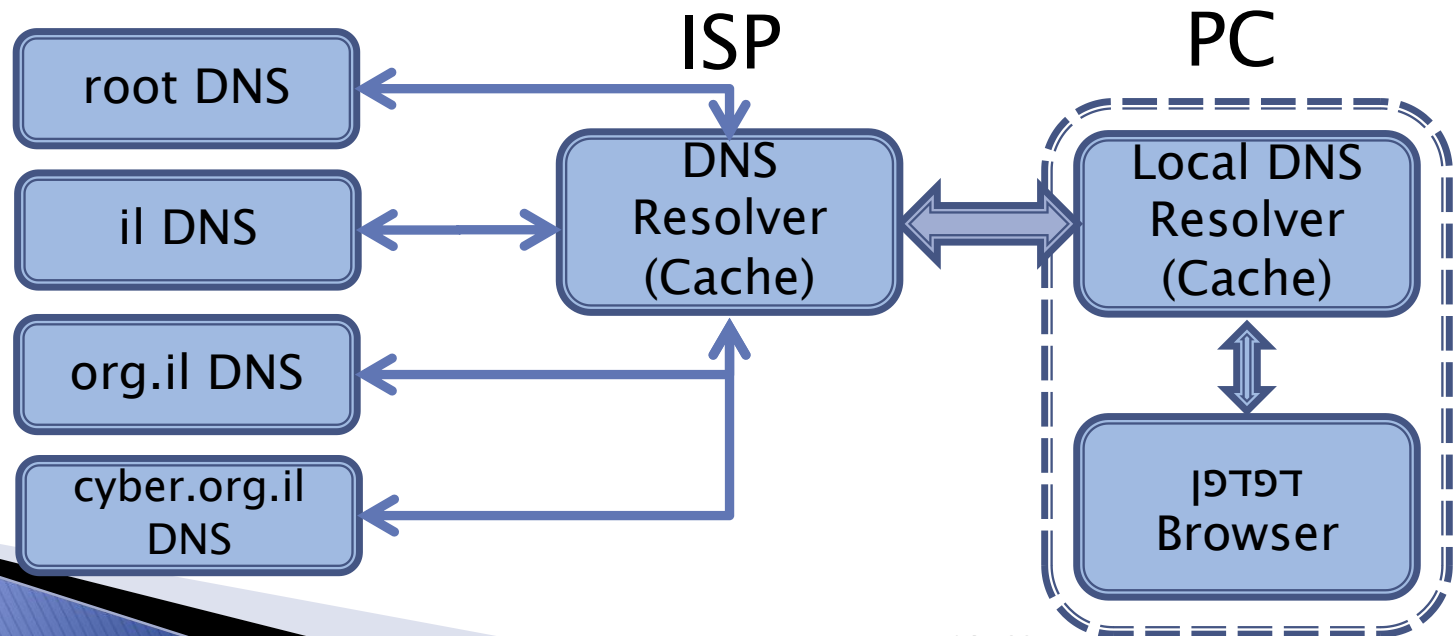
שלב 2: המחשב פונה לשרת DNS שמוגדר לו*

- בדרך כלל שייך לספק האינטרנט (ISP)
- נקרא DNS Resolver
- בודק אם כתובת ה-IP נמצאת ב-cache, אם כן התהליך מסתיים
- * למרבית הנתבים הביתיים יש cache ולעיתים החיפוש ייעצר שם



אז איך עובד פרוטוקול DNS?

- ▶ שלב 3: ה-DNS Resolver מפרק את ה-URL לרמות ומבצע בקשות לדוגמה, עבור `www.cyber.org.il`:
- פונה ל-root ומבקש את ה-IP של שרת ה-DNS שאחראי על il
- פונה לשרת שאחראי על il ומבקש את ה-IP של השרת שאחראי על `org.il`
- פונה ל-`org.il` ומבקש את ה-IP של השרת שאחראי על `cyber.org.il`
- פונה ל-`cyber.org.il` ומבקש את ה-IP של המשאב `www` (דף האינטרנט)
- הודות לשימוש ב-cache סביר שחלק מהבקשות ייחסכו



סוגים של DNS Queries

▶ האם שמתם להבדל בין התוצאה של בקשת ה-DNS של המחשב מה-ISP, לבין התוצאות של בקשות של ה-ISP משרתי ה-DNS?

- ה-ISP ביקש משרתי ה-DNS רק את ה-IP של השרת הבא
- **Iterative query** - יכול להחזיר הפניה לשרת הבא או כתובת מלאה (אם הגענו לשרת האחרון)
- המחשב ביקש משרת ה-DNS של ה-ISP כתובת IP מלאה
- **Recursive query** - בקשה אחת מחזירה תשובה מלאה

Iterative / Recursive Query

- ▶ כדי לקבוע אם ה-query הוא recursive או iterative, הלקוח שולח בקשה עם דגל RD (Recursion Desired)
- ▶ האם אפשר לבקש recursive query מכל שרת DNS?
 - אפשר לבקש, אבל סביר שהבקשה תיפסל
 - שרתי DNS צריכים להגן על עצמם מהטרדות 😊
- ▶ תרגיל: בצעו את תרגיל 4.14 בספר – התבוננות בשאילתת DNS
 - חפשו את ה-RD בין הדגלים
 - פירוט הדגלים: <https://www.wireshark.org/lists/wireshark-users/201207/msg00015.html>

Reverse Mapping

▶ כמו שאפשר לבקש המרה מ-URL ל-IP, אפשר לבקש המרה מ-IP ל-URL

▶ משתנה ה-type של שאילתת ה-DNS

- Type 'A': URL → IP
- Type 'PTR': IP → URL

▶ בצעו את תרגיל 4.15

◦ מיצאו את ה-Domain Name של הכתובת 8.8.8.8



[הצג את דף הבית](#)
[הצג את דף הבית](#)
[הצג את דף הבית](#)

מ'ל גילוף יכצו | גוגל הצג

[דף הבית של גוגל](#) - [גוגל](#) - [גוגל](#) - [גוגל](#) - [גוגל](#)

© 2002 גוגל

TTL- Time To Live

- ▶ איך ה-DNS Resolver במחשב שלנו / ב-ISP יודע אם כתובת ה-IP שיש ב-cache שלו היא עדכנית?
 - תשובת DNS מגיעה עם TTL, שמציין למשך כמה זמן היא עדכנית
- ▶ בצעו את תרגיל 4.16, חקירת תשובת DNS
 - מיצאו את שדה ה-TTL



- בשיעור זה למדנו אודות פרוטוקול DNS. הסבירו:
- למה בכלל צריך DNS?
 - איך נקראים השרתים בראש ההיררכיה של DNS?
 - מה ההבדל בין שאילתא רקורסיבית לאיטרטיבית?
 - איזה סוג רשומה ממפה בין כתובת IP לשם דומיין?
 - מה החשיבות שבשדה TTL?

