**Final project – communication network**

323834309\_329808554\_325445997\_322241654

**Part 1** –

**שאלה 1**

גורמים אפשריים להאטה:

* עומס ברשת - TCP משתמש במנגנון בקרת עומס (Congestion Control) כדי להקטין את קצב השידור כאשר הרשת עמוסה.
* בקרת זרימה - TCP משתמש במנגנון חלון הקבלה כדי לוודא שהשולח לא ישלח יותר נתונים ממה שהמקבל יכול לעבד, אם הReceive Window קטן, העברת הנתונים תהיה איטית.
* אובדן חבילות - אובדן חבילות יכול לגרום להאטה משמעותית בגלל מנגנון ההעברה מחדש של TCP (Retransmission)​.
* עיכוב חבילה - פרוטוקול TCP כולל מדידה של Round Trip Time (RTT), ואם הוא גדול, זה עלול להאט את ההעברה.

דרכי פתרון:

* בדיקת עומסים ברשת עם Wireshark כדי לזהות עומס.
* הגדלת חלון הקבלה של TCP כדי לשפר את קצב השידור.
* הקטנה של גודל החבילות הנשלחות כך שיתאימו ליכולת הרשת להעביר אותן בשלמותן, ובכך למנוע עיכובים ואובדן מידע.
* מעבר לפרוטוקול UDP במקרים שבהם אמינות לא חשובה (למשל סטרימינג).

**שאלה 2**

השפעת בקרת הזרימה של TCP:

TCP משתמש בחלון קבלה (Receive Window) שמציין לשולח כמה נתונים ניתן לשלוח מבלי להציף את המקבל​. אם השולח חזק מאוד (כגון שרת מהיר) והמקבל איטי יותר (למשל מחשב בעל משאבים מוגבלים), קצב העברת הנתונים יהיה מוגבל בהתאם לחלון הקבלה.

השפעה כאשר השולח חזק מהמקבל:

אם המקבל לא מצליח לעבד את הנתונים במהירות מספקת, הוא ישלח חבילות עם Receive Window קטן, מה שיגרום לשולח להאט את ההעברה.

**שאלה 3**

תפקיד הניתוב:

ניתוב ברשת מאפשר לחבילות מידע לעבור מנתיב המקור אל היעד על פי אלגוריתמים שונים שמטרתם למצוא את המסלול היעיל ביותר​.

השפעת בחירת הנתיב על ביצועים:

* עיכוב - מסלול קצר יותר מפחית עיכובים.
* עומס ברשת - מסלול עמוס גורם להאטת העברת הנתונים.
* אמינות - בחירת נתיב אמין יותר מונעת אובדן חבילות ועיכובים.

גורמים בהחלטות ניתוב:

* שימוש בפרוטוקול OSPF לחישוב מסלולים ברשתות מקומיות.
* שימוש בפרוטוקול BGP לניהול תעבורה בין ספקיות אינטרנט.

**שאלה 4**

MPTCP מאפשר שימוש במספר נתיבים במקביל במקום להשתמש רק בחיבור אחד, מה שמאפשר:

* שיפור ביצועים - הזרמת נתונים ביותר מנתיב אחד מגדילה את מהירות ההעברה.
* עמידות בפני תקלות - אם אחד הנתיבים נופל, החיבור ממשיך לעבוד דרך נתיב אחר.
* איזון עומסים - עומס הרשת מתחלק בין מספר חיבורים​.

**שאלה 5**

גורמים אפשריים לאובדן חבילות:

שכבת הרשת -

* עומס ברשת - כאשר נתבים מקבלים יותר חבילות ממה שהם יכולים לטפל, חלק מהחבילות נזרקות.
* בעיות חומרה – כשל בנתב או בעיות בתשתית הרשת עלולות לגרום לאובדן חבילות.

שכבת התעבורה -

* מנגנון בקרת עומס של TCP TCP עשוי להקטין את קצב השידור במקרה של עומס ולגרום לעיכובים.
* פרגמנטציה של חבילות – אם חבילות מחולקות למקטעים קטנים יותר, ייתכן שחלקן ילכו לאיבוד.

פתרונות אפשריים:

1. בדיקה עם Wireshark כדי לנתח חבילות ולזהות היכן הן אובדות.  
2. שיפור תשתית הרשת והגדרת Quality of Service (QoS) לניהול עדיפויות של חבילות.  
3. שימוש בפרוטוקול MPTCP כדי לאזן את העומס ולמנוע הצטברות נתונים על נתיב אחד.

**Part 2** –

**FlowPic\_Encrypted\_Internet\_Traffic\_Classification\_is\_as\_Easy\_as\_Image Recognition**

**התרומה העיקרית של המאמר**:

המאמר מציג **שיטה חדשה לזיהוי וסיווג תעבורת אינטרנט מוצפנת**. הרעיון הוא **לקחת רק את זמני ההגעה וגודל החבילות, להפוך אותם לתמונה (FlowPic) ולהשתמש ברשתות נוירונים קונבולוציוניות (CNN) כדי לנתח את הדפוסים**.  
השיטה מאפשרת **לזהות את קטגוריית תעבורה, כמו - גלישה, וידאו, או צ'אט. וגם את היישום הספציפי כמו למשל – Skype או YouTube, Facebook Video,** גם כשנעשה שימוש **בהצפנה ע"י VPN או Tor** , שיטות שבדרך כלל מקשות מאוד על סיווג תעבורה.  
בניגוד לשיטות קודמות, **אין צורך לבחור ידנית מאפיינים סטטיסטיים**, מה שמקל על היישום שלה ומשפר את הדיוק.  
בנוסף, FlowPic משתמש בחלק קטן מאוד מהנתונים הנשלחים ולא צריך לחכות לכל התקשורת בין שני הצדדים כדי לסווג את התעבורה. זה אומר שהשיטה יכולה לסווג תעבורה מהר יותר ובצורה יעילה יותר משיטות אחרות שדורשות את כל המידע מההתחלה עד הסוף.

יתרון נוסף הוא שהשיטה **לא משתמשת בתוכן החבילות ,** מה שאומר שהיא **שומרת על פרטיות המשתמשים**, וגם **דורשת פחות משאבי אחסון וחישוב** לעומת שיטות קודמות.

זאת הפעם הראשונה שנעשה שימוש בטכניקות זיהוי תמונה לסיווג תעבורת רשת מוצפנת.

ברוב המקרים, השיטה מצליחה להשיג תוצאות טובות יותר משיטות קודמות.

**מאפייני תעבורה שנעשה בהם שימוש (שגרתיים וחדשים)**:

המאמר משתמש רק בשני מאפיינים מתוך זרימות הרשת -גודל החבילות זמן ההגעה שלהן.

החידוש במאמר הזה הוא איך שהנתונים האלה מעובדים - **הנתונים הופכים לגרף שבו ציר ה-X מראה את זמני ההגעה של החבילות וציר ה-Y מראה את גודל החבילות.** **הגרף הזה הופך לתמונה שנקראת .FlowPic התמונה הזאת מוכנסת לרשת נוירונים (CNN) שמזהה את סוג התעבורה בצורה מאוד מדויקת.**

היתרון של השיטה הזאת הוא שהיא עובדת באופן אוטומטי לגמרי. בשיטות הישנות, החוקרים היו צריכים לבחור בעצמם ובאופן ידני איזה נתונים חשובים לזיהוי - למשל כמה חבילות נשלחו, מה הגודל הכולל שלהן, או מידע מהכותרות של הפרוטוקולים. לעומת זאת, השיטה החדשה פשוט לוקחת את התמונה שנוצרה – FlowPic , ומאפשרת לרשת הנוירונים לגלות בעצמה אילו דפוסים חשובים לזיהוי. זה עובד טוב יותר כי התמונה מראה דפוסים שקשה לתאר במספרים פשוטים, והשיטה עובדת על כל סוגי התעבורה בלי קשר לפרוטוקול שבשימוש.

**ממצאים עיקריים ותובנות:**

התוצאות העיקריות:  
1. דיוק בסיווג תעבורה -

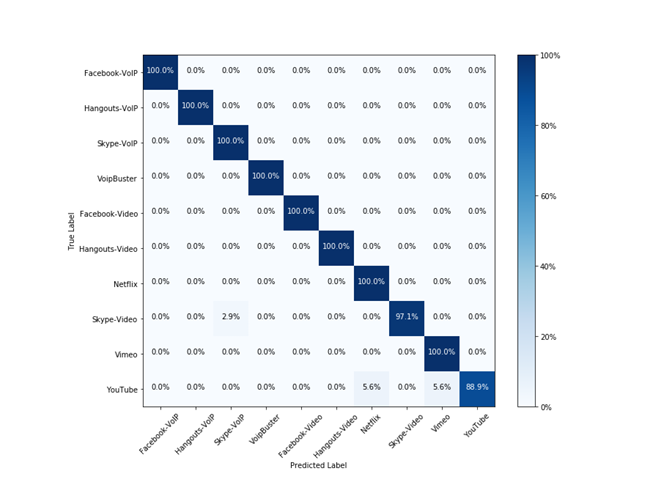
A table with numbers and a number of results

AI-generated content may be incorrect.

2. סיווג בינארי של קטגוריה מול שאר הקטגוריות -  
A table with text and numbers

AI-generated content may be incorrect.

3. זיהוי אפליקציות ספציפיות –

דיוק של 99.7% בזיהוי בין 10 אפליקציות VoIP ווידאו שונות  


4. זיהוי סוג ההצפנה –

דיוק כולל של 88.4% בזיהוי בין Non-VPN , VPN ,Tor.

זיהוי גבוה של תעבורת Tor – דיוק של 97.7%  
A table with text and numbers

AI-generated content may be incorrect.

התובנות העיקריות:  
1. למידת דפוסים כללים- השיטה לומדת את המאפיינים של קטגוריות תעבורה כמו וידאו או צ'אט, ולא רק של יישומים ספציפיים. זה מאפשר זיהוי מוצלח של יישומים חדשים שלא נראו במהלך אימון השיטה.

2. VPN משפיע על דפוסי התעבורה אבל עדיין ניתן לזהות בדיוק גבוה למרות ההצפנה שלו. ואילו Tor מקשה יותר על הזיהוי ומפחית את הדיוק של השיטה.

3. יעילות – השיטה לא משתמשת בתוכן החבילות עצמן ולכן פרטיות המשתמשים גדולה יותר וגם יש פחות צורך בשימוש בזיכרון. בנוסף השיטה משתמשת בחלק קטן מאוד מהתעבורה , ובכיוון אחד בלבד , וזה מאפשר לסווג את התעבורה בצורה מהירה ויעילה יותר.

**Early Traffic Classification With Encrypted ClientHello: A Multi-Country Study**

**רעות צריכה להוסיף**

**התרומה העיקרית של המאמר**:

**התרומה העיקרית של המאמר**:

**ממצאים עיקריים ותובנות:**

**Analysing HTTPS Encrypted Traffic to Identify User’s Operating System, Browser and Application**

**התרומה העיקרית של המאמר**:

המאמר מציג שיטה חדשנית ומדויקת (96.06% התאמה) לזיהוי מערכת ההפעלה, הדפדפן והאפליקציה של המשתמש עבור תקיפה \*פסיבית , בהתבסס על ניתוח תעבורת HTTPS מוצפנת, תוך התייחסות למאפיינים נוספים שלא נשקלו במחקרים קודמים כחלק מהניתוח.

השיטה כוללת איסוף נתונים, ניתוח המאפיינים (נעשה סיווג למאפיינים שנלקחו בעבר בחשבון ומאפיינים נוספים בכדי לבדוק את השפעת 2 הקבוצות ופרט הקבוצה המחדשת במחקר), ולבסוף סיווג הנתונים באמצעות אלגוריתם למידת מכונה ושימוש בדפוסים חוזרים למערכת ההפעלה, הדפדפן והאפליקציה המתאימים.

\*תקיפה פסיבית = תקיפה שאינה ישירה אל מול המותקף, אלא דרך תעבורת הרשת.

**מאפייני תעבורה שנעשה בהם שימוש (שגרתיים וחדשים)**:

\*פירוט המאפיינים מתואר בנספח א מטה.

מאפיינים בסיסיים שכבר נעשה בהם שימוש במחקרים קודמים:

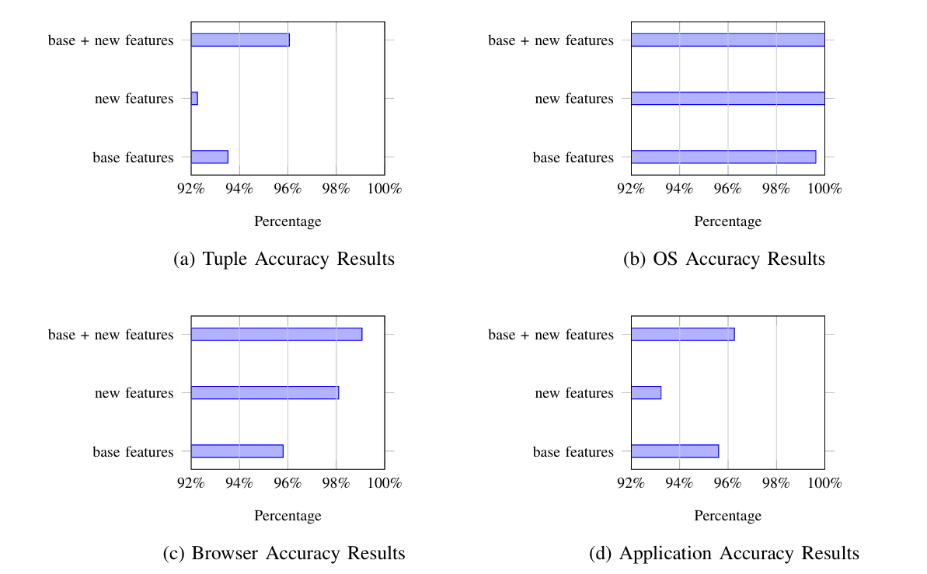
1. מספר חבילות שנשלחו ונקלטו.
2. גודל של חבילות.
3. זמני הגעה בין חבילות.

מאפייני תעבורה חדשים שנלקחים בחשבון:

1. מאפייני TCP.
2. מאפייני SSL.
3. קצב העברת הנתונים.
4. Bursts = התפרצויות של חבילות (לדוג' בדפדפנים שנדרשת טעינה של הרבה מרכיבים יחד).
5. זמנים בין הגעת חבילות עבור שיאי התעבורה.

**ממצאים עיקריים ותובנות:**

1. תרומת המאפיינים המחדשים:

**ניתן לראות שהמאפיינים שאין בהם חידוש לבדם, סייעו במחקר זה לאיתור מערכת ההפעלה, הדפדפן והאפליקציה יחד בדיוק של 93.52% . יחד עם הנתונים החדשים האלגוריתם הגיע לדיוק של 96.06%.** בנוסף לכך עבור סיווג מערכת ההפעלה האלגוריתם הגיע לדיוק של 100%, אפילו בהתבסס על המאפיינים החדשים בלבד. בסיווג הדפדפן והאפליקציה – לנתונים החדשים יש תרומה בסיווגם יחד עם מאפייני הבסיס וכן הקושי העיקרי הוא בזיהוי האפליקציה.

1. דיוק האלגוריתם:

*דיוק הסיווג עולה מתוצאות של מעל 20,000 ניסויים ומוצג במאמר במטריצות בלבול כך שהשורות מייצגות את הערכים האמיתיים והעמודות מייצגות את הערכים הצפויים. במטריצה זו ערכים גבוהים באלכסון מעידים על דיוק גבוה ואילו ערכים גבוהים מחוץ לאלכסון מעידים על זיהוי שגוי של קטגוריה מסוימת.*

A chart with black and white text

AI-generated content may be incorrect.בהתאם לשיטה שהמטריצות מוצגות, ניתן לראות בהסתכל על כל המרכיבים יחדיו: מערכת ההפעלה, הדפדפן והאפליקציה שהסיווג מדויק באופן יחסי אך לא תמיד.

בנוסף לכך, מערכת ההפעלה מזוהה בדיוק מוחלט, הדפדפן והאפליקציה פחות. באפליקציה רוב הבלבול נובע בשל זיהוי שגוי של אפליקציית Facebook.

חשוב לציין שחוסר התאמה בסיווג לא בהכרח נובע מזיהוי שגוי אלא יכול לנבוע בשל חוסר וודאות לגבי הסיווג (2 פלטפורמות קרובות, היעדר פלטפורמה ממאגר הנתונים ועוד).

**ובשורה התחתונה – הרגלי השימוש של המשתמש בפלטפורמות השונות אכן משפיעים על מאפייני התעבורה המוצפנת כך שניתן לזהות אותם.**

**Part 3** –

**ממצאים מההקלטות:**

1. IP header fields

נועה צריכה להוסיף –

ttl

ipv4 ipv6

1. TCP header fields

*בחרנו לבחון את שכיחות סוגי הדגלים השונים עבור כל אפליקציה.*

A graph of different colored squares

AI-generated content may be incorrect.

1. ACK – ניתן לראות שבכל האפליקציות יש שכיחות של דגל זה בעיקר, ובמיוחד בדפדפנים. השכיחות בדפדפנים מעידה על כך שהם מתבססים על אישור קבלת החבילות כחלק מהמנגנון המהימן שמבטיח את הגעת החבילות בסדר הנדרש. לעומת זאת ביתר האפליקציות השכיחות נמוכה יותר כיוון שבסטרימינג יש פחות תלות באישור מידי.

בנוסף לכך, השכיחות הקיימת עבור zoom משמשת עבור אישור מידי בשידור חי והימנעות מאובדן מידע.

1. PSH+ACK – גם פה, רוב מופעי הדגל הם בדפדפנים וזה מעיד על שליחה ישירה של המידע ללא המתנה. לעומת זאת, עבור YouTube ו – Spotify יש פחות שימוש בדגל זה כיוון שסטרימינג לא דורש תגובתיות מיידית ונעשה שימוש ב- buffering.

באופן דומה לדגל ACK, ניתן לראות שעבור zoom התדירות של דגל זה בינונית ביחס ליתר האפליקציות מפני שיש צורך לשמור על איזון בין מהירות ליציבות באפליקציה זו.

1. דגלי ACK ו-PSH+ACK עבור אפליקציית ה- zoom נמצאים בשכיחות בינונית בשל עיקרון השירות של האפליקציה ורצון לשמור על איזון בין מהירות ליציבות. האישור הפחות תלותי באישורים מעיד על כך שניתן לבצע שידור חי תקין ולצד כך גם להימנע מאובדן חבילות.
2. SYN&SYN+ACK -

מופיע בשכיחות הכי גבוהה ב- chrome ויותר מ- firebox, מה שיכול להצביע על פתיחת מספר חיבורים במקביל לטעינת דפים עם הרבה תוכן. מנגד ניתן לראות שכמעט ואין שכיחות יתר האפליקציות וזה יכול לסמן על חיבור אחד ארוך ומתמשך כך שהשידור רציף.

1. FIN+ACK –

מופיע בשכיחות הכי גבוהה ב- chrome ובהתאמה לפתיחת חיבורים מרובה (4), ב- firefox השכיחות נמוכה יותר וזה יכול להעיד על כך שאפליקציה זו נוטה להשאיר חיבורים לזמן רב יותר. מנגד ניתן לראות שכמעט ואין שכיחות ביתר האפליקציות בהתאמה לפתיחת החיבורים (4).

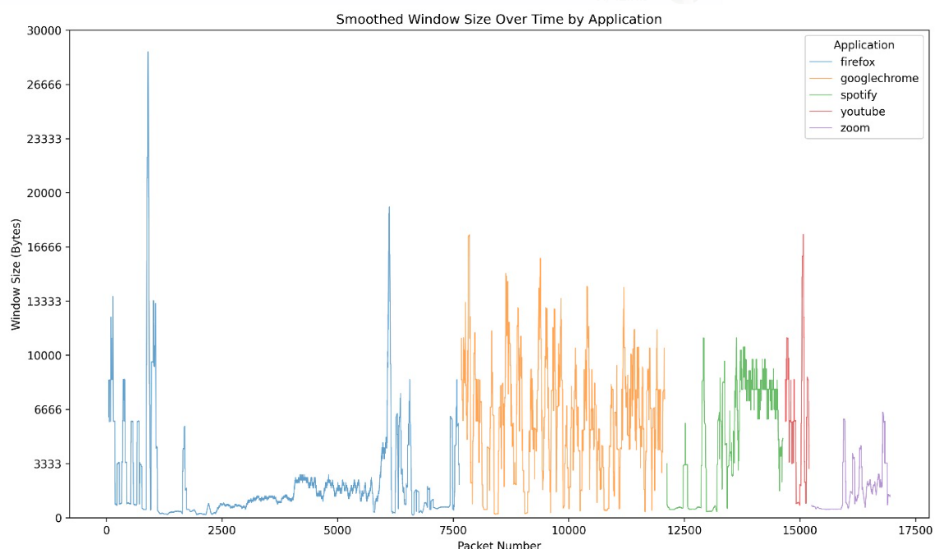
1. RST+ACK -

ניתן לאבחנה בין כל האפליקציות רק עבור Spotify וזה מעיד שבעת כשל באפליקציה זו הסגירה מתבצעת באופן שאינו פתאומי, הדגל מאפשר לאותת לשרת המקבל על הניתוק העתידי ואישור על קבלת ההודעה על סיום החיבור.

1. 0X0019 –

ניתן לראות דגל מותאם אישית רק עבור Spotify וזה מעיד שהוא משתמש בפרוטוקול מותאם אישית לעומת שאר האפליקציות שהיעדר הדגל עשוי להעיד על שימוש בפרוטוקולים שגרתיים.

*בנוסף לכך, בחרנו לבחון את גודל החלון כחלק ממנגנון מהימנות TCP תוך הצגת מגמת השינוי של גודל החלון לאורך זמן (ההקלטה).*

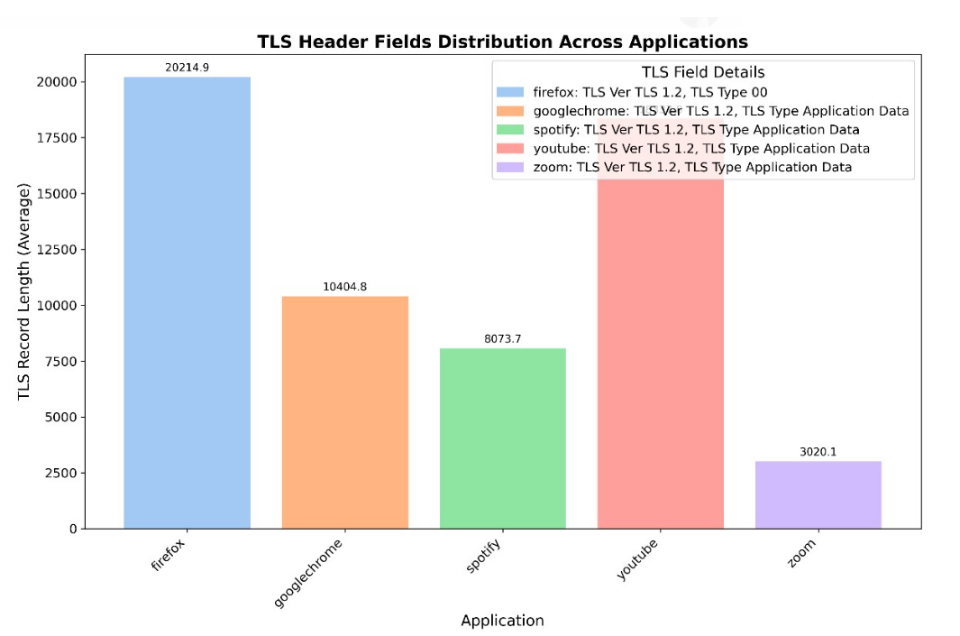


1. ניתן לראות שהמגמה עבור firefox משתנה בתנודתיות גבוהה יותר לעומת chrome שהמגמה יותר מתונה. הבדל זה יכול להעיד שהתאמת גודל החלון עבור firebox נעשתה באמצעות שינויים קיצוניים של הגדלת החלון מה שהוביל ל"כישלון" ואילו ההתאמה עבור chrome נעשית בצורה יותר מבוקרת.
2. ניתן לראות שהמגמה עבור Spotify ו- YouTube יציבה יחסית לאורך ההקלטה ובנוסף לכך גודל החלון נשאר בערכים הנמוכים. לעומת זאת עבור zoom המגמה משתנה בצורה יותר קיצונית וזה בא בהלימה עם כך שאפליקציה זו תלויה ברוחב פס מספק בכדי לאפשר וידאו בשידור חי.

היציבות ב- Spotify תואמת להעברת הנתונים באופן רציף וכן חלון הזזה קטן עבור YouTube תואם לכך שלאפליקציה יש buffering המאפשר את השידור ברציפות.

1. TLS header fields

*בחנו את שדות הכותרת בפרוטוקול TLS המוצפן שניתחנו בעזרת קבצי המפתחות.*



קודם להבין על ה-00 הזה

1. כל האפליקציות עשו שימוש בפרוטוקול TLS מגרסה 1.2 והיא הנפוצה ביותר.
2. ממוצע גודל החבילה ב- firefox הוא מקסימלי וכן גדול פי 2 מהממוצע של chrome. נובע מהעובדה שב- firefox מידע מועבר גם כחלק מתהליך ההצפנה
3. Packet sizes

*כדי לבחון את גדלי החבילות באפליקציות השונות ראינו לנכון להציג את ההתפלגות גודלי החבילות לאורך זמן (ההקלטה) כך שיהיה ניתן לנתח גם את המגמה, שיאים ועוד.*

A graph showing a number of different colored lines

AI-generated content may be incorrect.

1. ניתן לראות שבהקלטה של zoom בשונה משאר האפליקציות יש מגמת שינוי בגודל החבילות לאורך כל ההקלטה, כלומר שונות גבוהה יותר בגודל החבילות. הסיבה לכך נובעת מהעובדה שאפליקציה זו כוללת מספר סוגי תעבורה (וידאו, אודיו ושיתוף מסך).

לעומת זאת עבור YouTube ניתן לראות שגודל החבילות עקבי וממוקד סביב טווח גדלים מסוים ובהתאמה לכך שאפליקציה זו מבוססת על העברת נתונים בקצב אחיד. בנוסף לכך עבור הדפדפנים המגמה מעידה על שליחת חבילות קטנות.

כמו כן, עבור Spotify ניתן לראות פחות עקביות, ככל הנראה כיוון שהאודיו מועבר בדינאמיות ובהתאם לאיכות ורוחב הפס הזמין.

1. Packet inter-arrivals

*בהתאם לזמני הגעת החבילות וחישוב הזמנים בין כל 2 חבילות עבור כל אפליקציה, בחרנו להציג את ממוצע זמן ההמתנה בכל ערוץ*

A graph showing different colored bars

AI-generated content may be incorrect.

1. ניתן לראות ש- Spotify בעלת ממוצע הגבוה ביותר של זמני המתנה בין הגעת החבילות באופן משמעותי, כלומר התעבורה פחות רציפה מיתר האפליקציות והסיבה לכך שבאפליקציה זו יש שמירה בbuffering ע"מ לאפשר השמעה רציפה של האודיו. לעומת זאת, האפליקציות YouTube ו- zoom מספקות זרימה רציפה של חבילות (Streaming) כדי לשמור על איכות השירות. זו עשויה להיות הסיבה לכך שאפליקציות אלה מספקות זמני המתנה קצרים יותר ובאמצעות כך יכולים לספק העברה מהירה של הנתונים.
2. שירותי הדפדפנים – chrome ו – firefox משתמשים בפרוטוקול TCP אשר מעביר את החבילות ברציפות תוך שמירה על מנגנון הסדר ואישור ההגעה שלמדנו. לאור מנגנון מהימן זה העיכובים בהגעת החבילות מצטמצמים עבור הדפדפנים ביחס ליתר האפליקציות ובנוסף לכך הדפדפנים בשונה מיתר האפליקציות לא נדרשים בהעברת אודיו/וידאו.
3. Flow size
4. Flow volume

**ניתוח הקלטה נוספת:**

שירת צריכה להוסיף

נספח א:

מאפיינים בסיסיים שכבר נעשה בהם שימוש במחקרים קודמים:

*\*Forward = מהלקוח לשרת, Backward = מהשרת ללקוח. בניתוח השיטה שהוצעה במאמר בחנו את העברת המידע לשני הצדדים.*

1. **מספר חבילות שנשלחו ונקלטו:**

* Forward/Backward packets – מספר החבילות שנשלחו בבדיקה אחת.
* Mean Forward/Backward packets – מספר החבילות הממוצע שנשלח עבור מספר בדיקות.
* STD forward/Backward packets – סטיית התקן של מספר החבילות שנשלחו עבור מספר בדיקות.
* Total packets – סה"כ מספר החבילות שנשלחו בשני הכיוונים (Forward and Backward).

1. **גודל של חבילות:**

* Forward/Backward total Bytes – מספר הבתים הכולל שנשלח (עבור כל החבילות).
* Minimum Forward/Backward packet – הגודל הקטן ביותר של חבילה שנשלחה.
* Maximum Forward/Backward packet – הגודל הגדול ביותר של חבילה שנשלחה.
* Minimum packet size – הגודל הקטן ביותר של חבילה שנשלחה מתוך החבילות שנשלחו בשני הכיוונים.
* Maximum packet size – הגודל הגדול ביותר של חבילה שנשלחה מתוך החבילות שנשלחו בשני הכיוונים.
* Mean packet size – הגודל הממוצע לחבילה שנשלחה מתוך החבילות שנשלחו בשני הכיוונים.
* Packet size variance – שונות של גודל החבילה מתוך החבילות שנשלחו בשני הכיוונים.

1. **זמני הגעה בין חבילות:**

* Min Forward/Backward inter arrival time difference – הזמן הקצר ביותר שעבר בין 2 חבילות רצופות.
* Max Forward/Backward inter arrival time difference – הזמן הארוך ביותר שעבר בין 2 חבילות רצופות.
* Mean Forward/Backward inter arrival time difference – הממוצע של הזמנים בין 2 חבילות רצופות.
* Mean forward TTL value – ערך ה- TTL הממוצע של חבילות שנשלחו Forward.

מאפייני תעבורה חדשים שנלקחים בחשבון:

1. **מאפייני TCP:**

* TCP initial window size – גודל חלון חיבור TCP בהתחלה.
* TCP window scaling factor – מקדם שינוי גודל החלון מעבר.
* Keep alive packets – מספר החבילות שנשלחו לצורך שמירה על חיבור פעיל.
* TCP Maximum Segment Size – גודל החבילה המרבי שניתן לשליחה בחיבור.

1. **מאפייני SSL:**

* SSL compression methods – מספר שיטות הדחיסה הנתמכות בפרוטוקול SSL.
* SSL extension count – מספר ההרחבות המשמשות בפרוטוקול TLS/SSL (יכולות לכלול מאפיינים נוספים בתהליך ההצפנה).
* SSL cipher method – מספר שיטות ההצפנה הנתמכות בפרוטוקול SSL/TLS.
* SSL session ID len – אורך המזהה אשר משמש לניהול חיבורי TLS.
* Forward SSL Version – גרסת SSL/TLS שנעשה בה שימוש בתעבורה Forward.

1. **קצב העברת הנתונים:**

* Forward/Backward peak MAX throughput – קצב העברת הנתונים המקסימלי לכל כיוון.
* Mean throughput of forward/backward peaks – הממוצע של שיאי קצב נתונים בתעבורה לכל כיוון.
* Forward/Backward min peak throughput – קצב העברת הנתונים המינימלי בתעבורה לכל כיוון.
* Forward/Backward STD peak throughput – סטיית התקן של שיאי קצב הנתונים בתעבורה לכל כיוון.

1. **Bursts = התפרצויות של חבילות (לדוג' בדפדפנים שנדרשת טעינה של הרבה מרכיבים יחד):**

* Forward/Backward number of bursts – מספר התפרצויות של חבילות לכל כיוון.
* Forward min peak throughput – קצב העברת הנתונים המינימלי שנמדד בהתפרצויות של חבילות בתעבורה Forward.

1. **זמנים בין הגעת חבילות עבור שיאי התעבורה:**

* Minimum backward/forward peak inter arrival time diff – הזמן המינימלי בתעבורה לכל כיוון.
* Maximum backward/forward peak inter arrival time diff – הזמן המקסימלי בתעבורה לכל כיוון.
* Mean backward/forward peak inter arrival time diff – ממוצע בין הגעת שיאי התעבורה לכל כיוון.
* STD backward/forward peak inter arrival time diff – סטיית התקן של הזמנים בתעבורה לכל כיוון.