Technologic Papers

התחמקות מזיהוי על ידי "Shadow keys"

מאת אריק קובלנוב

מבוא

מעבדות CyberArk גילו לאחרונה פגיעות במנגנון ייצור המפתחות של גוגל, מאמר זה מציג את הפגיעות שזכתה לכינוי "Shadow keys". הפגיעות דווחה לגוגל בינואר 2018 ותוקנה אחר תהליך "responsible disclosure". בהמשך מפורט ציר הזמן של הגילוי והדיווח עד תיקונו על ידי גוגל.

לפני התיקון של הפגיעות, גוגל יצרה עבור כל מפתח חוקי של משתמש לשירותי ה-API שלה, 3 מפתחות נוספים המיוחסות לאותו פרוייקט של המשתמש, שאמור להשתמש ביישומים שלו עם המפתח החוקי שקיבל מגוגל. "Shadow keys" הם למעשה העתקים לא חוקיים עם שינוי קטן כפי שיבואר בהמשך, אשר המשתמש לא מודע כלל להיווצרותם ולקיומם.

זאת פגיעות משמעותית מכיוון שאם תוקף היה מצליח בדרכים שונות להשיג את ידיו על מפתח חוקי של גוגל, הוא היה יכול ליצור "Shadow keys" מהמפתח החוקי ובעזרתן לקבל גישה ליישומים כאילו הם היו המפתחות החוקיים. בנוסף, התוקף היה יכול לעקוף את מנגנון הזיהוי והתשלום של מספר שירותים של גוגל.

טכניקת גילוי "Shadow keys" שמתוארת בהמשך המאמר מאפשרת לתוקף או משתמש זדוני ליצור רשימה של מפתחות לא חוקיים על ידי שירות "script loader" של גוגל.

הסבר קצר על שירותי ה-API של גוגל

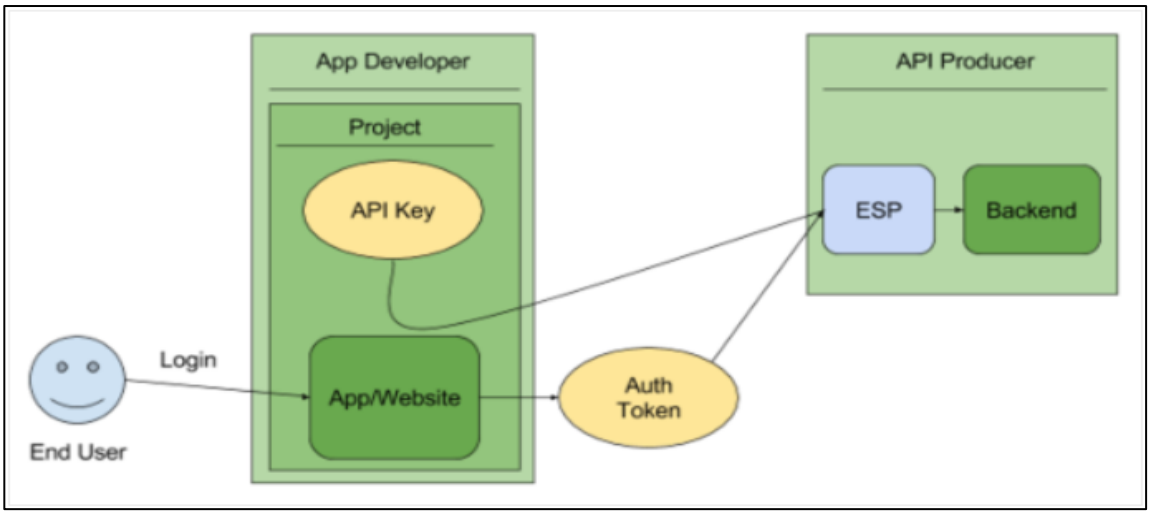
מפתח ה-API מאפשר לגוגל ולהמשתמש לעקוב אחר השימוש של היישומים שהוא מפתח בשירותי המידע והנתונים של גוגל. עבור לקוח תוכנית סטנדרטי, מפתח ה-API נותן גישה למכסה יומית מוגבלת בחינם, כמו גם את האפשרות לשלם לגוגל כדי להגדיל את המכסה היומית המוגבלת. עבור לקוח תוכנית פרימיום, המשתמש חייב להשתמש במפתח ה-API כדי לגשת לכל התכונות המתואמות אישית של תוכנית הפרימיה.

הרשמה וקבלה של מפתח API מבטיחה שגוגל תוכל ליצור קשר עם המשתמש לגבי היישום, במידת הצורך. ב-API console של גוגל, המשתמש יכול גם לחפש מפתח קיים או להציג רשימה של שירותי API מופעלים, באמצעות מפתח ה-API לאימות ישומים, משתמשים יכולים לנהל את כל ה-API שלהם ב-API console של גוגל, לגשת לנתוני שימוש בזמן אמת ול-30 יום של נתוני שימוש היסטוריים ולהציג דוחות שימוש עם יותר מ-30 ימי נתונים בפורטל התמיכה של הענן של גוגל.

למפתח ה-API יש שני תפקידים עיקריים:

1) זיהוי פרויקט - מזהה את היישום או את הפרוייקט שמבצע קריאה ל-API ולשירות ספציפי של גוגל.
2) הרשאת פרוייקט - בודק אם היישום קיבל אישור גישה ל-API והשירות של גוגל.

על ידי זיהוי הפרוייקט הקריטי, מפתחות ה-API מאפשרים לייחס פרוייקט של המשתמש ולאפשר או לדחותו על ידי פרוייקטים שלא קיבלו גישה או הופעלו על ידי ה-API. (ראה איור 1).



[איור 1 - זיהוי ואישור של מפתח על ידי שירות ה-API של גוגל.]

אופן הגילוי של Shadow keys

השתמשנו בשירות של גוגל "script loader" אשר מאפשר למשתמש לייבא בקלות מפתחות API ולציין הגדרות נוספות (כגון שפה, מיקום, גרסת ממשק וכן הלאה). מפתחים מתמצאים יכולים גם להשתמש בטעינה דינמית או טעינה אוטומטית כדי לשפר את הביצועים של היישום שלהם על ידי שירות זה.

לפני תיקון הפגיעות, גוגל הודיע למשתמשים שלה ש-"שירות זה כבר לא דורש מפתחות API ואין צורך לערוך שינויים בקוד של היישום שלהם גם אם היישום משתמש במפתח API של גוגל". יש לציין ששירות זה היה משתמש במפתחות ישנים (שונים במבנם מהמפתחות החדשים של גוגל שבהם השתמשנו) אשר היו מיוצרים על ידי גוגל וכיום כבר לא בשימוש.

שירות "script loader" הוא השירות שעל ידו גילנו את "Shadow keys". שירות זה בהינתן מפתח API חוקי מחזיר תגובה אשר מצביעה על האם המפתח הוא חוקי או לא חוקי והאם הוא קיים ב-API Console של גוגל.

כפי שניתן לראות עבור מפתח **חוקי** השירות החזיר:

google.loader.KeyVerified = true:

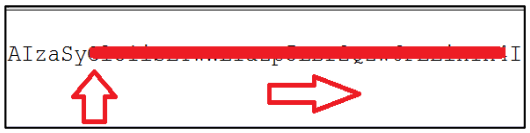
ועבור מפתח **לא חוקי** השירות החזיר:

google.loader.KeyVerified = false:

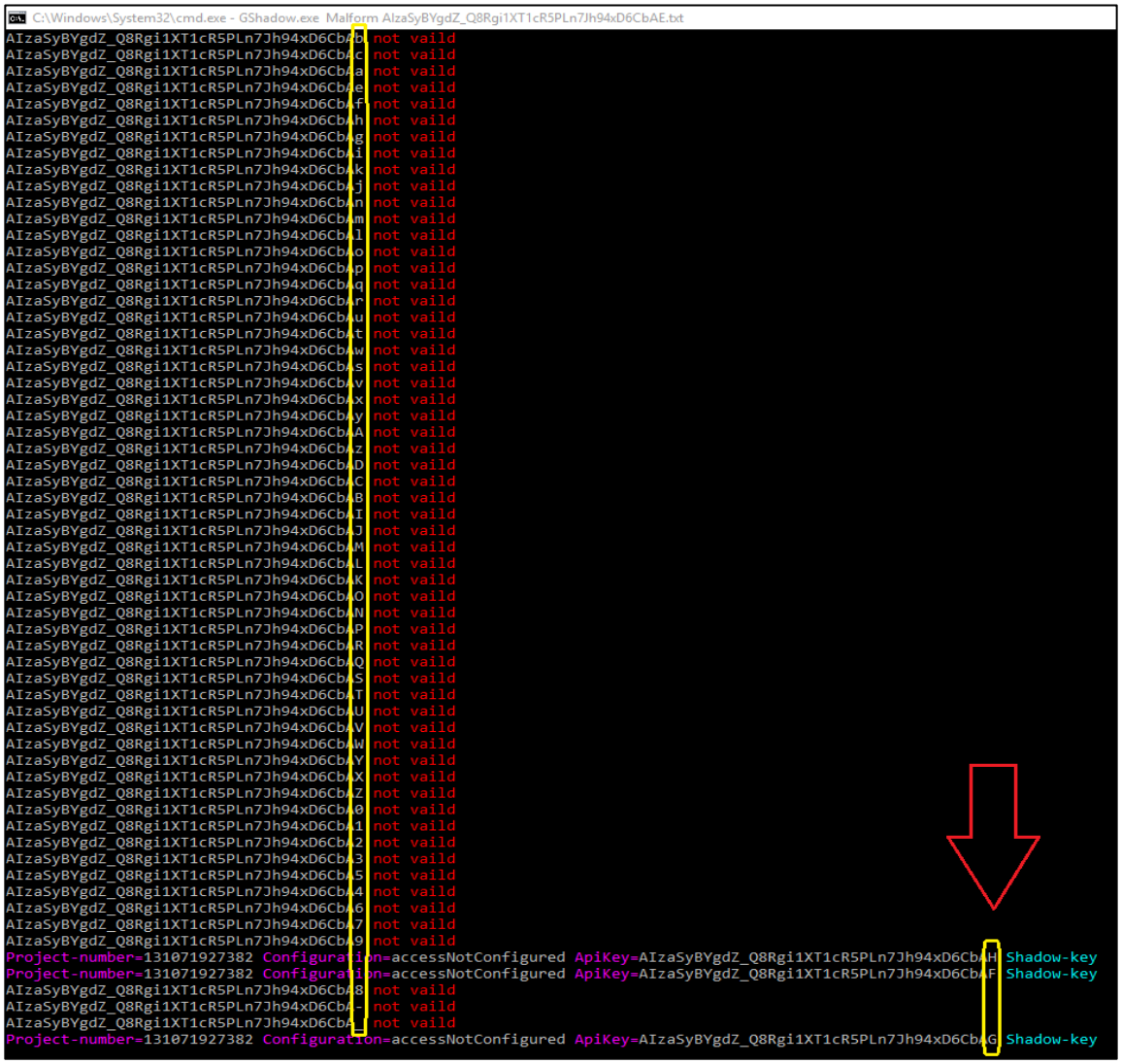
לפי גוגל: "היכולת לגלות מפתחות API קיימים וחוקיים היא בלתי ניתנת למימוש על פי התכנון המבנה והיצירה של המפתחות על ידי גוגל. מפתח בן 36 אותיות כשכל אות יכולה להיות מבין 64 אותיות שונות מצריך 36 בחזקת 64 קומבינציות אפשריות של מפתחות. מתקפה שתנסה למצוא את כל המפתחות החוקיים שקיימים על ידי מעבר על כל האפשריות, לא אפשרית מבחינת זמן סביר למצוא אפילו מפתח חוקי אחד".

אך עם זאת, גילינו שזה אפשרי למצוא מפתחות נוספים מבלי לרוץ על כל האפשריות והקומבינציות, אלא ריצה על מפתח מקורי חוקי אחד מספיקה כאשר בכל פעם אנו מחליפים משמאל לימין של המפתח מהאינדקס במיקום השישי עד האינדקס במיקום ה-39 בכל פעם להיות כל אחת מהאותיות שלהלן:

```
abcdefghijklmnopqrstuvwyxzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789-_
```

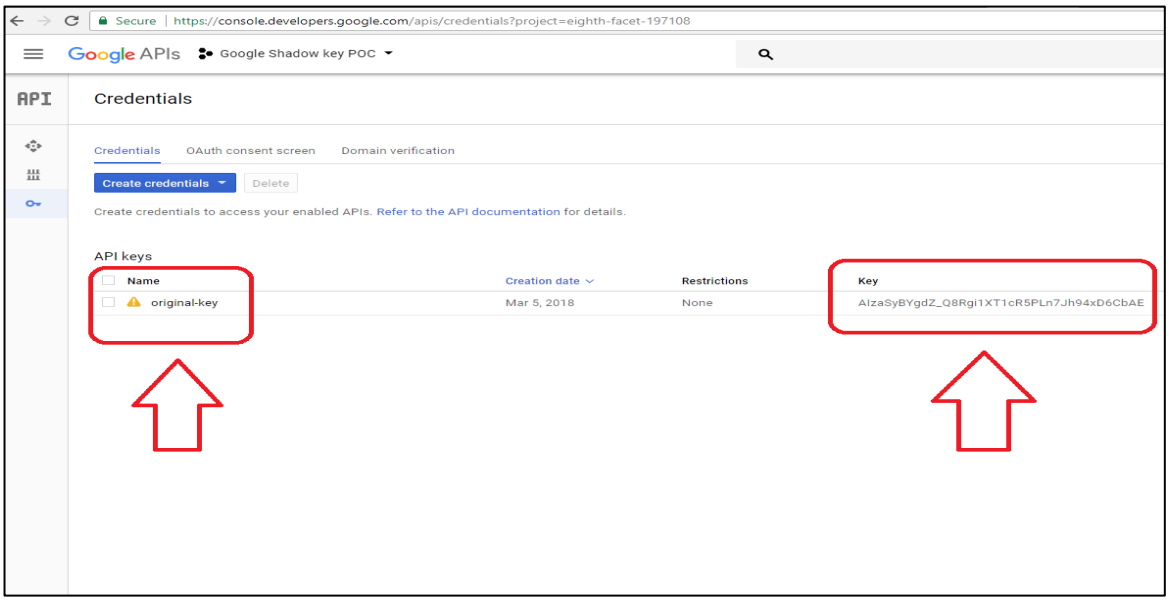


כפי שניתן לראות באופן ברור יותר על ידי התצלום מסך הבא:



התוצאה שקיבלנו הם 3 מפתחות API נוספים - "Shadow keys". ניתן להבחין שמפתחות הללו שונים מן המפתח המקורי רק במיקום 39. גילינו שאנו יכולים להשתמש בכל 3 המפתחות הללו בשביל לקבל גישה למידע משרתי גוגל על ידי היישום שלנו.

לשם הוכחת היכולת בשימוש של "Shadow keys" החלטנו לקבל מפתח מקורי אחד שניתן לנו על ידי שירותי ה-API של גוגל ב-API Console. כפי שניתן לראות בתצלום המסך הבא יש לנו רק מפתח חוקי אחד שנוצר עברנו ב-API Console.



לקחנו את המפתח המקורי והפעלנו עליו את הכלי שמממש את הטכניקה שתיארנו למעלה. כפי שניתן לראות קיבלנו העתקים של אותו המפתח השונים זה מזה ומן המפתח המקורי רק באות אחת, האחרונה מימין במיקום 39:

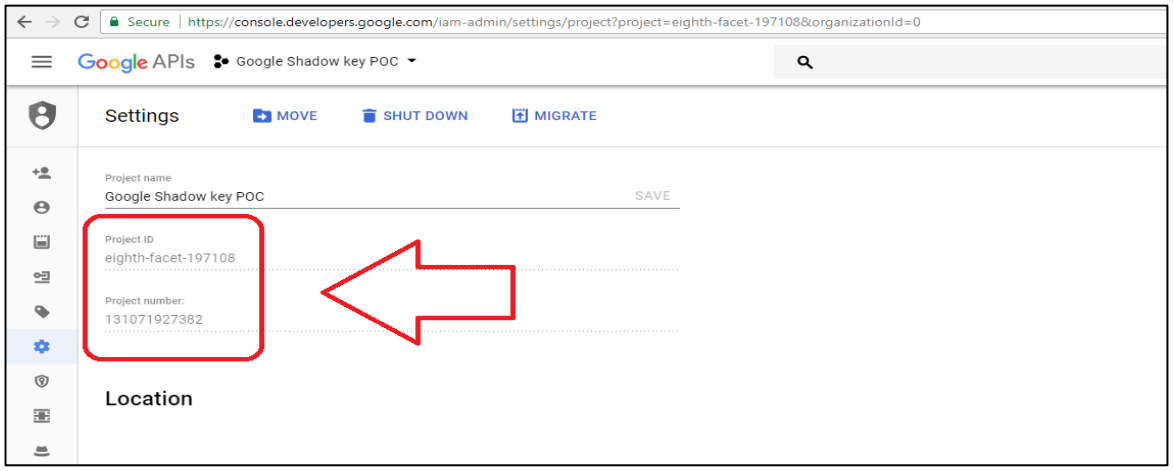
כפי שניתן לראות גם קיבלנו מידע שמדווח לנו לאיזה מספר פרוייקט בגוגל שלושת המפתחות הללו מיוחסים.

את המידע הזה השגנו בעזרת השירותים הבאים ופשוט הוספנו שירותים אלו לכלי שלנו.

https://www.googleapis.com/youtube/v3/videos?part=id&key=[API_KEY]

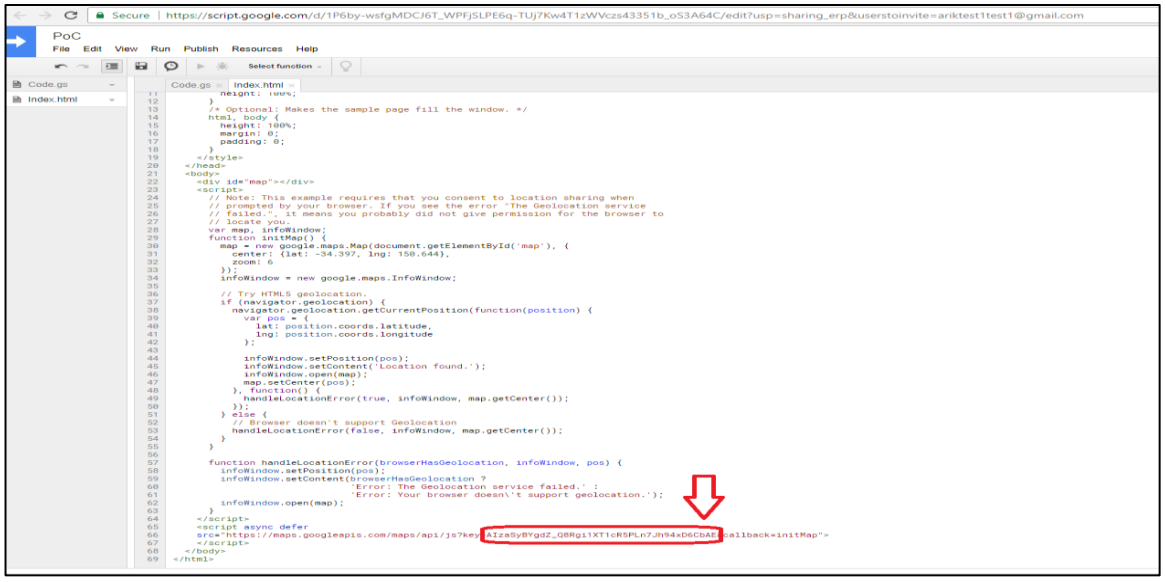
https://translation.googleapis.com/language/translate/v2/detect?key=[API_KEY]

כפי שניתן לראות, כל שלושת המפתחות שמצאנו מיוחסים למספר הפרוייקט שפתחנו ב-API Console בגוגל. בנוסף, ניתן לראות כי שלושת המפתחות הללו כלל לא קיימים אצלנו בפרוייקט:

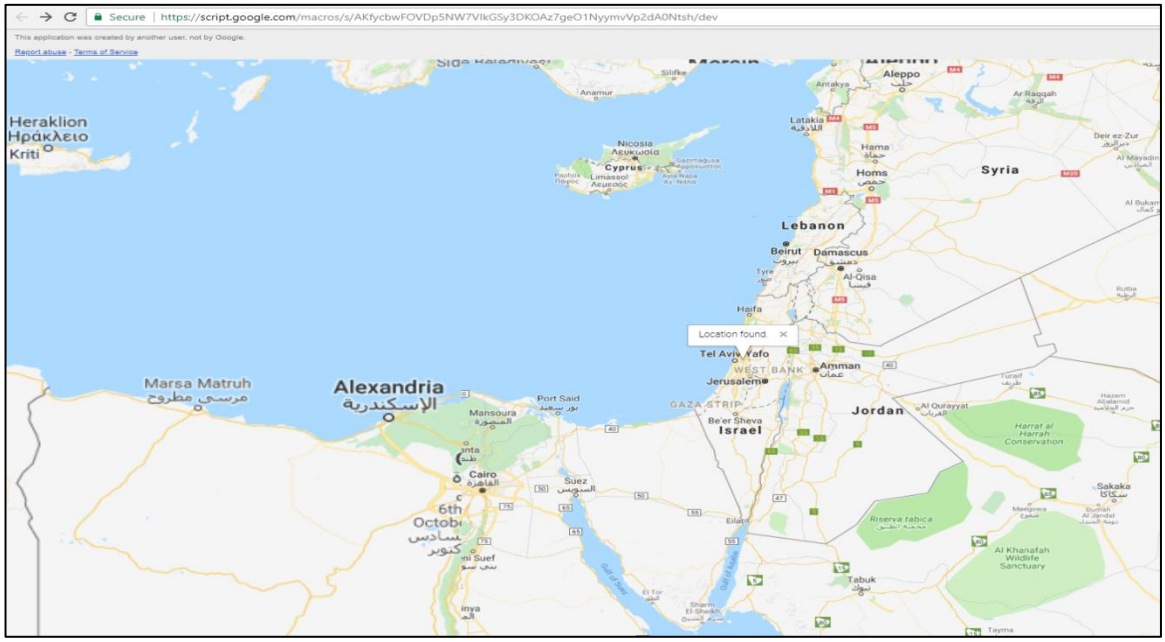


השימוש ב- Shadow keys

יצרנו יישום בגוגל המאחסן את הקוד הפשוט שלנו שמשתמש בשירותי גוגל מפות. היישום מציג למשתמש את המקום בו הוא ממוקם ושממנו הוא גולש ליישום. הקוד משתמש במפתח המקורי שהשירות של גוגל יצר עבורנו כפי שניתן לראות בתצלום הבא:

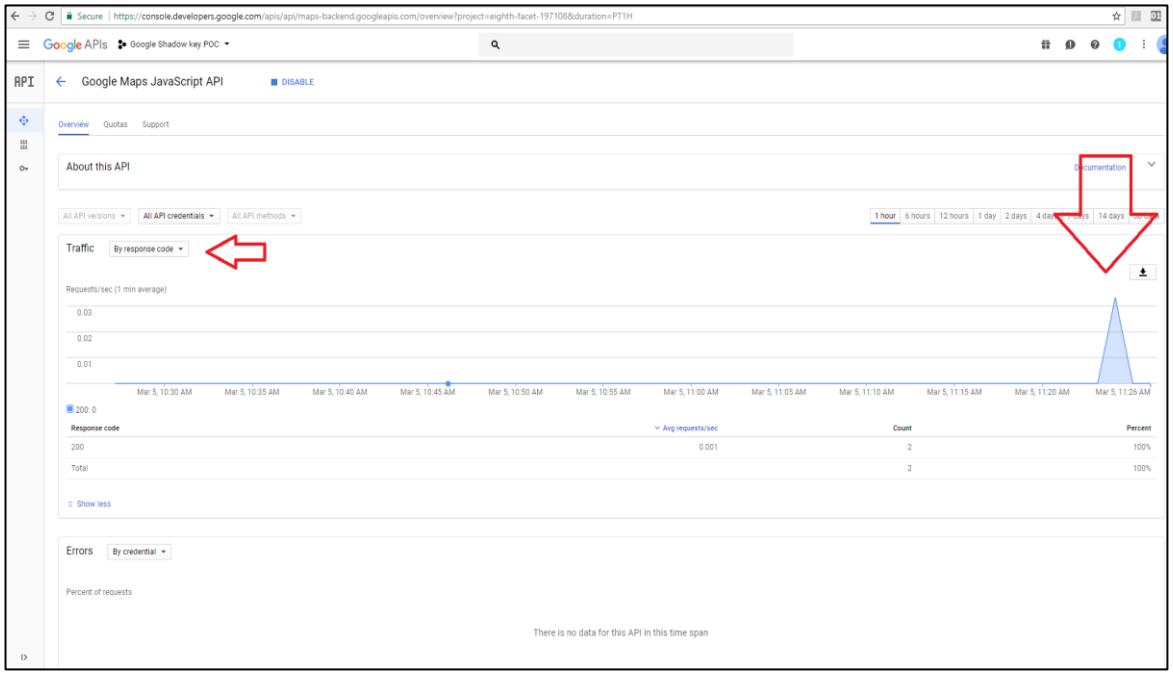


גלישה על ידי המשתמש ליישום שלנו אשר מאוחסן בשרתי גוגל יציג את מיקומו בדפדפן:

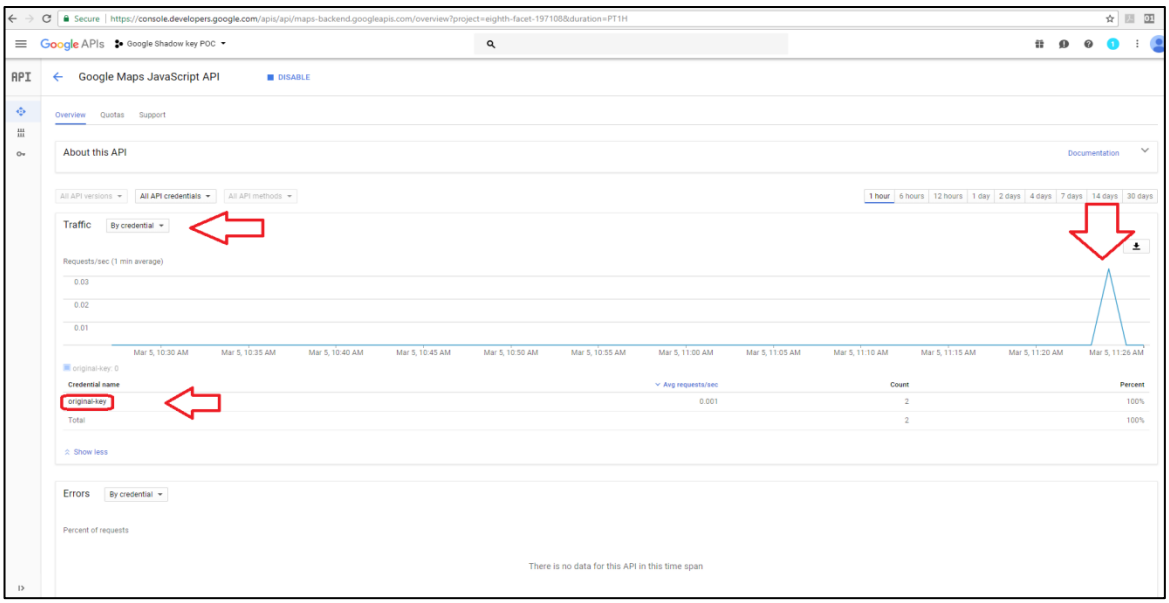


כפי שניתן לראות בתצוגה הגרפית של המשתמש, השרת של גוגל החזיר מידע למשתמש בהתייחסות למפתח המקורי בו הוא השתמש ואשר זוהה במערכת API Console של גוגל.

שימוש בפילטר "By response code" מראה את הגרף של התגובה:

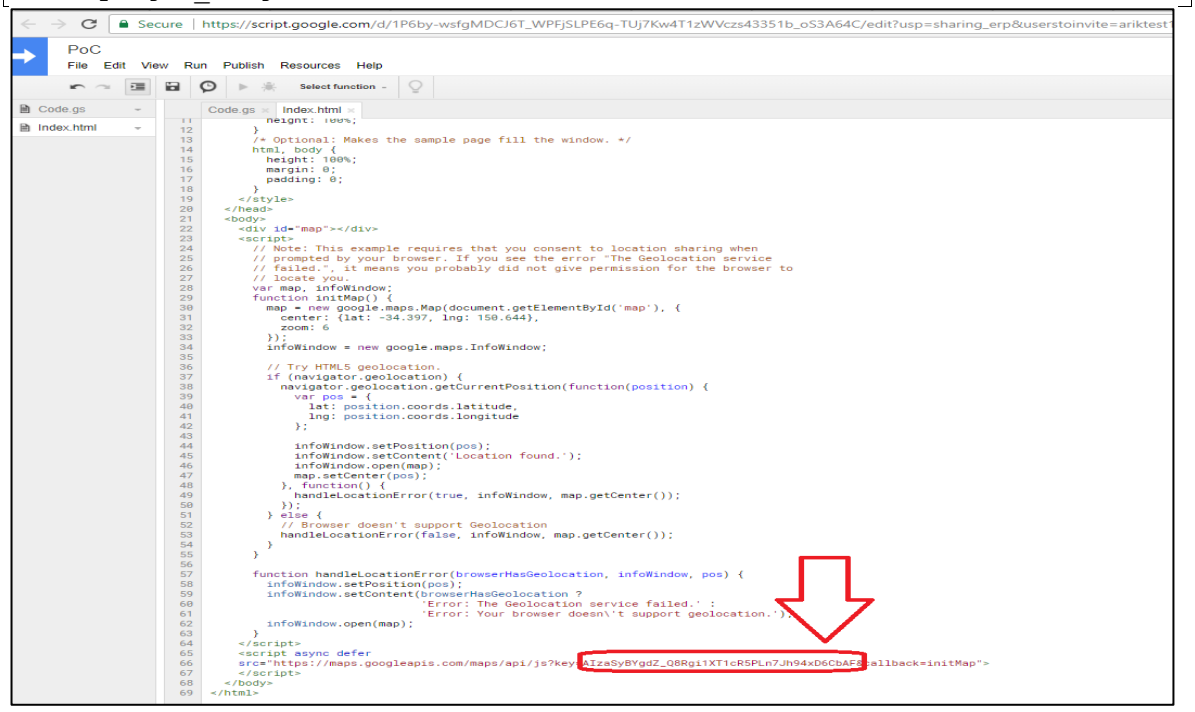


את אותה התגובה על גרף ניתן לראות גם על ידי הפילטר של "By credentials" אשר מעיד על כך שהמפתח המקורי אכן זוהה ונעשה בו שימוש לקבלת גישה למידע של שירותי גוגל מפות:

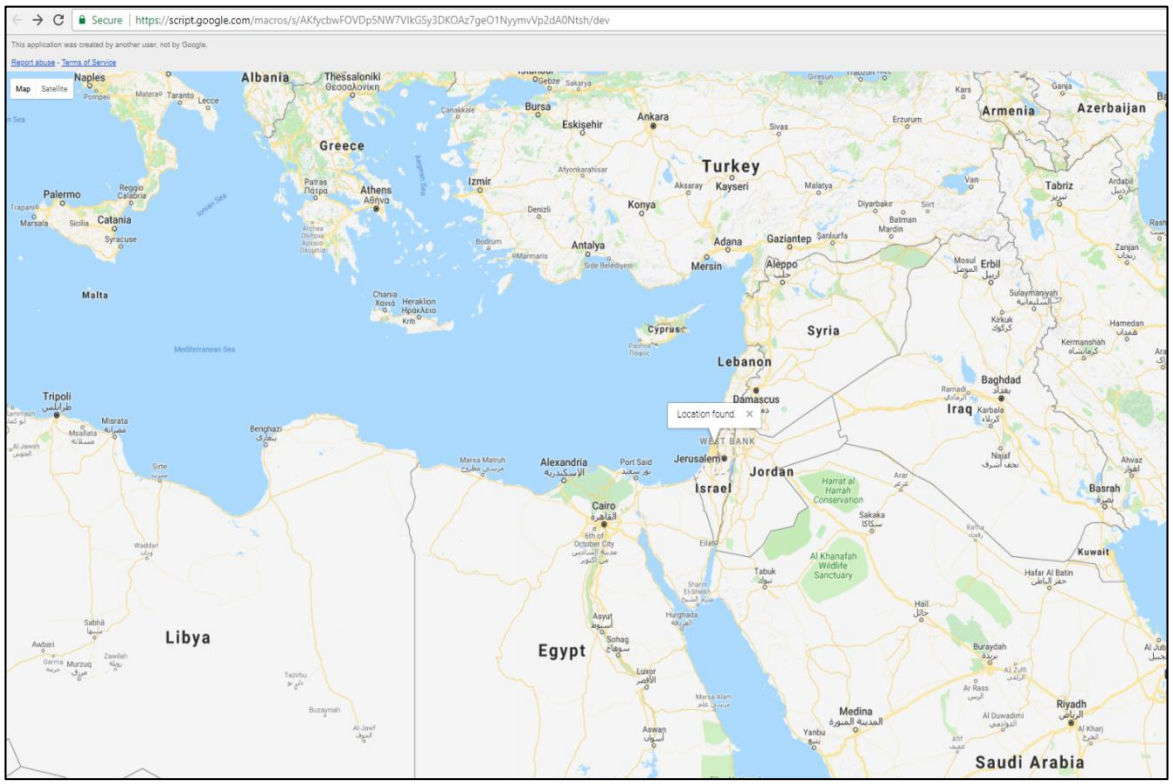


כעת אנו נשתמש ב-"Shadow keys" שמצאנו מוקדם יותר ביישום שלנו. "Shadow key" :

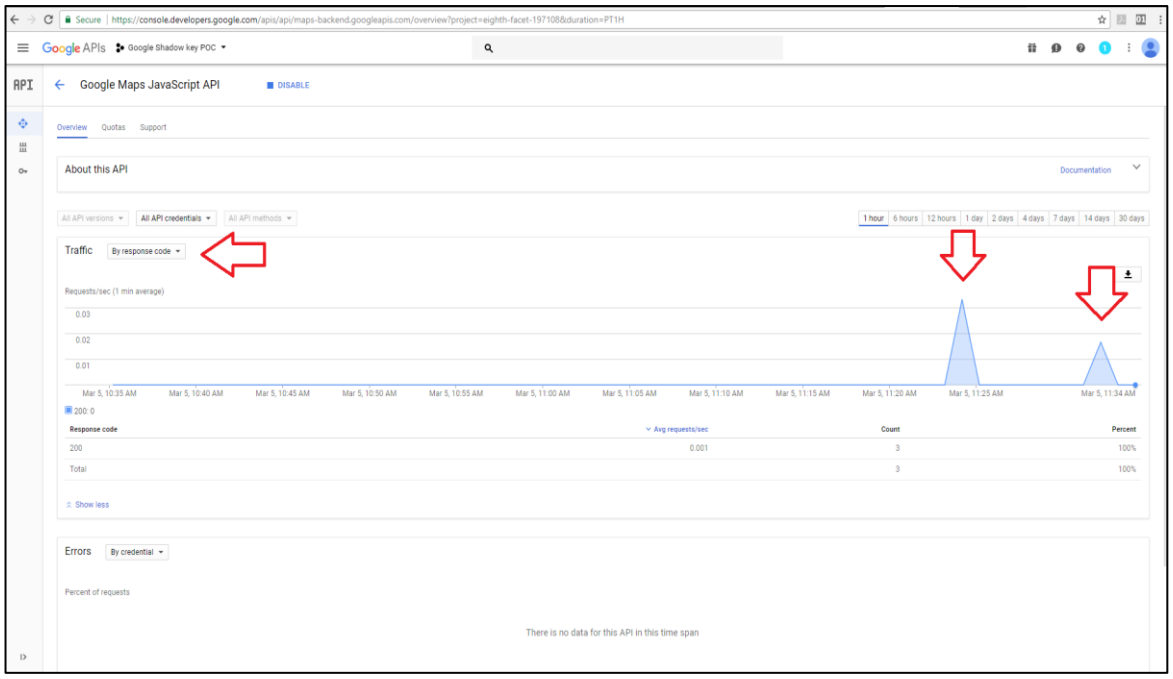
```
AIzaSyBYgdZ_Q8Rgi1XT1cR5PLn7Jh94xD6CbAF
```



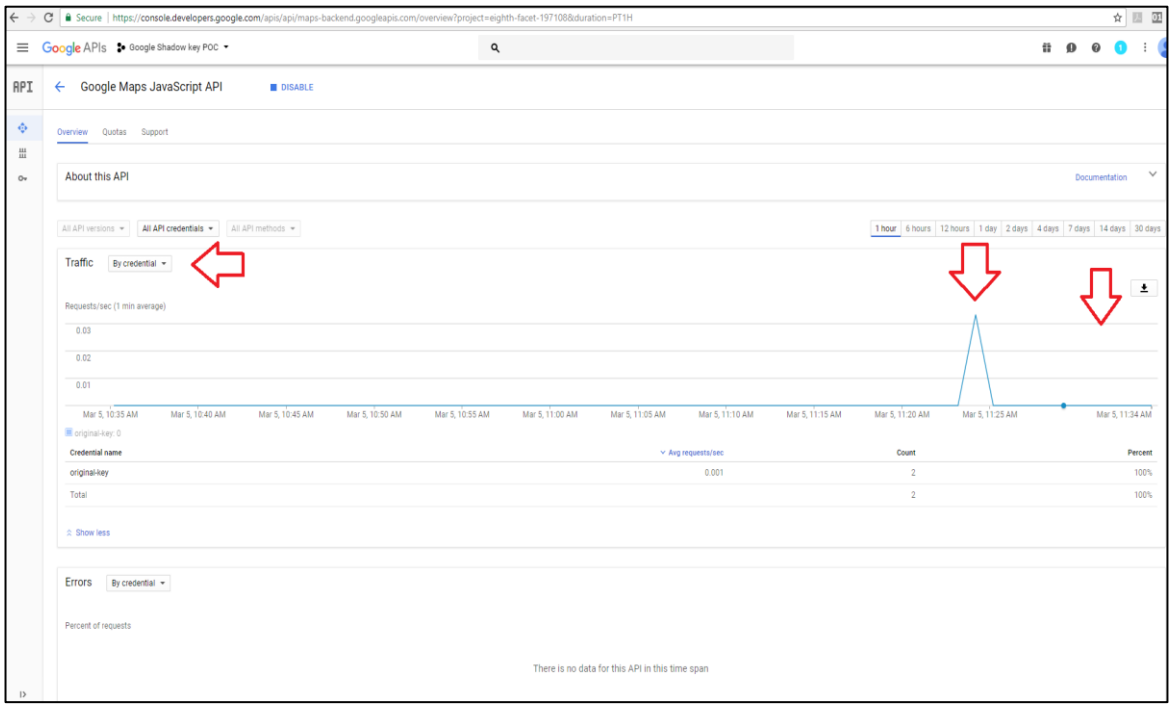
כפי שניתן לראות גלשנו לדף של היישום שלנו בדפדפן והוא אכן מציג את מיקומנו , כלומר "Shadow key" אכן ניתן לשימוש וחוקי מבחינת השירותים וה-API Console של גוגל:



כפי שניתן לראות על ידי פילטר "By response code" ה-API Console זיהה שמשתמש גלש לדף וביקש מגוגל שירותים לגבי מיקומו:

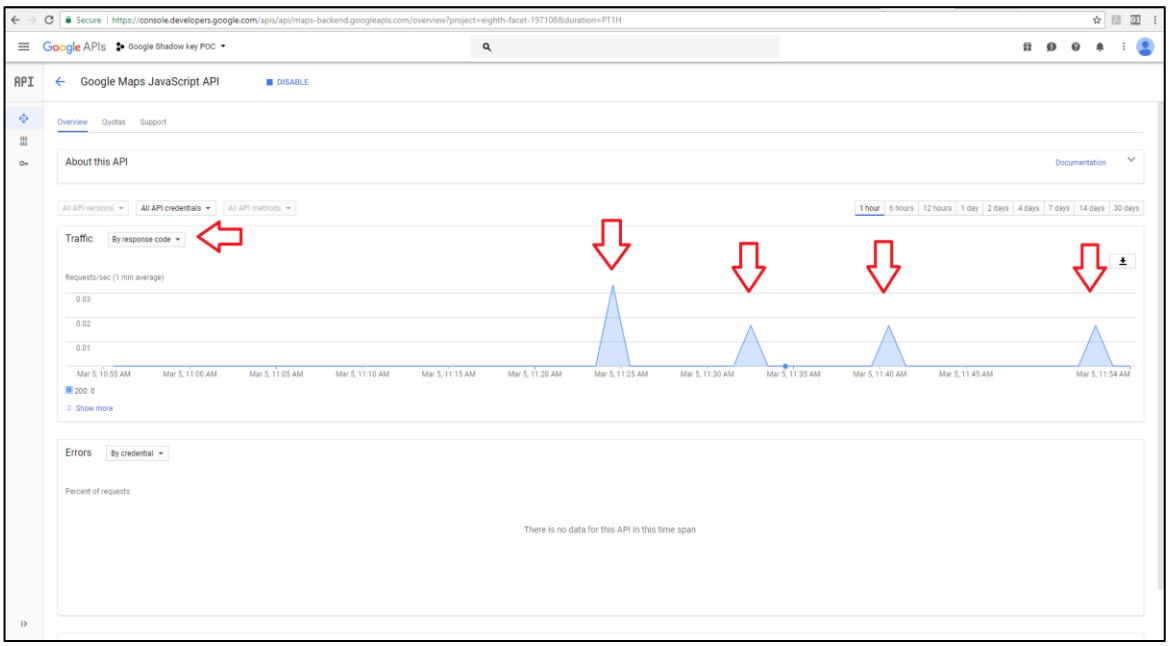


אך כאשר אנו משתמשים בפילטר של "By credentials" המערכת של ה-API Console לא מזהה את המפתח שנעשה בו שימוש לבקשת הנתונים משרתי גוגל מפות:

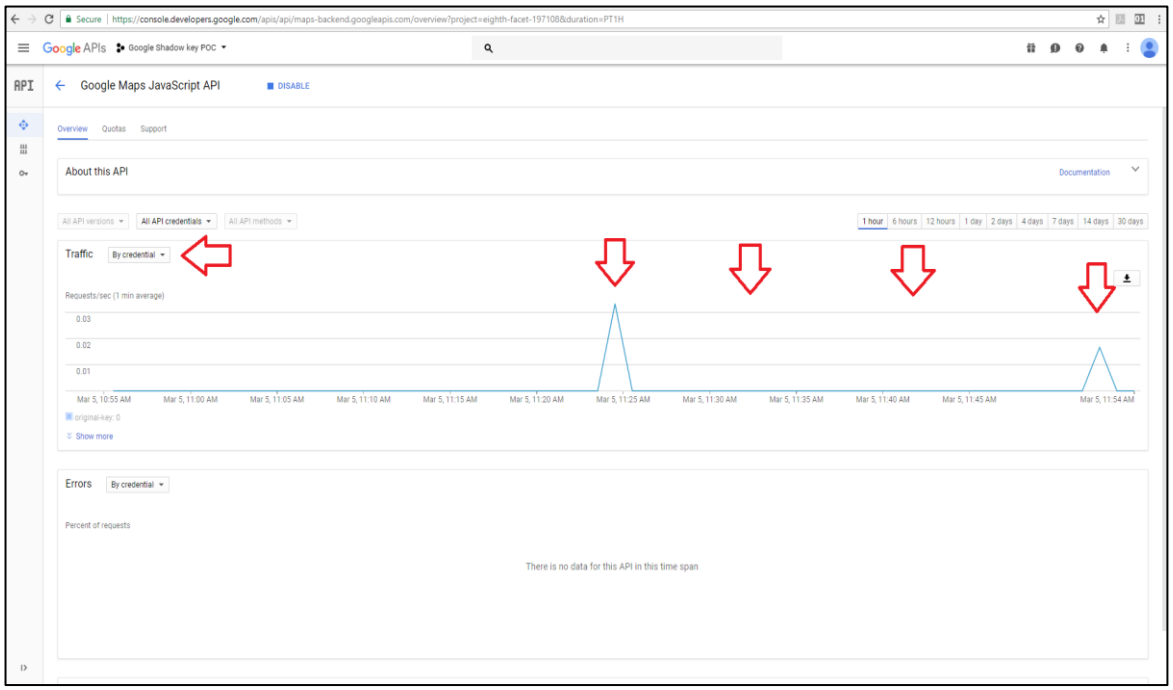


עשינו עוד מספר בקשות בשימוש של "Shadow key" ואז שוב החזרנו את המפתח המקורי שקיבלנו לקוד האפליקצייה.

כפי שניתן לראות על פי "By response code" נעשה שימוש ובקשה משירותים של גוגל מפות על ידי האפליקצייה שלנו:

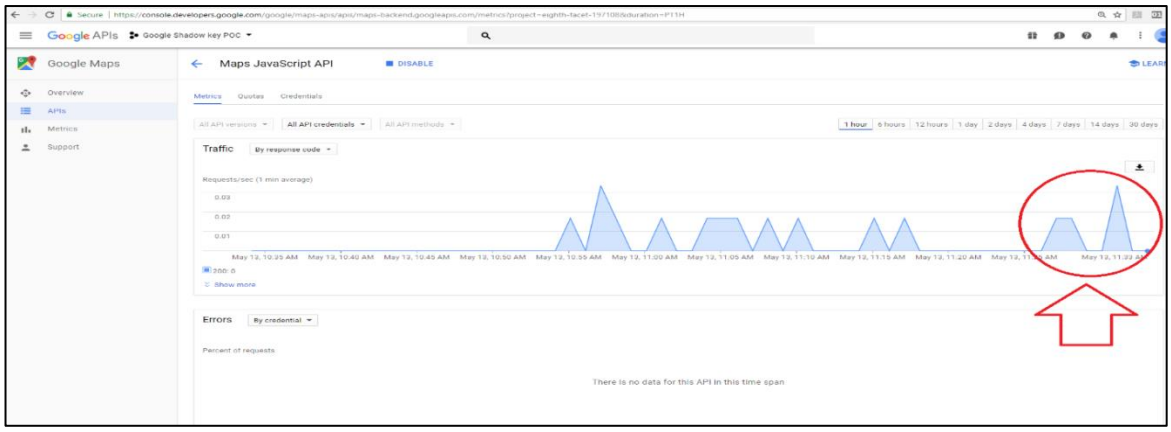


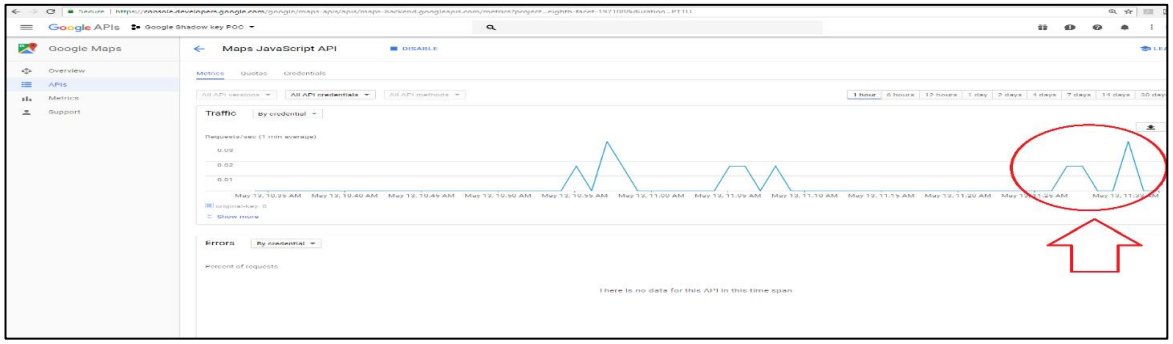
אבל כפי שניתן לראות יכולנו לגשת לשירותים של גוגל מפות מבלי להיות מגולים בשימוש של "Shadow key". המקומות הריקים המסומנים על פי הפילטר של "By credentials" מראים כי המערכת לא זיהתה את המפתחות ולכן הגרפים שונים וחסרים:



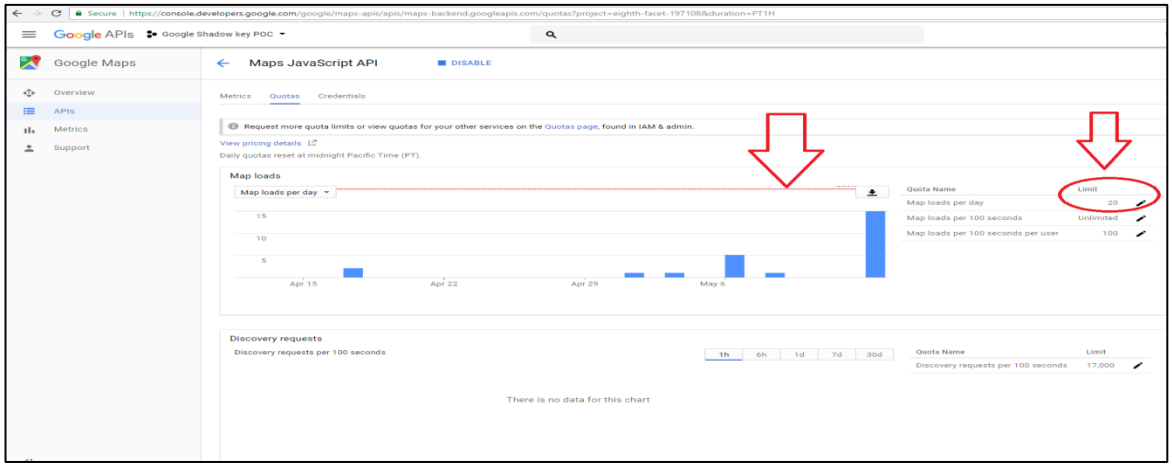
בתצלומי המסך הבאים ניתן לראות בלשונית החיובים שאנו מוגבלים במספר הבקשות שלנו לטעינת המפה ובקשת מידע משרתי גוגל מפות. השימוש של המפתח שלנו מוגבל ל-20 טעינות ובקשות ליום.

כאשר אנו משתמשים במפתח המקורי שקיבלנו מגוגל בתוך קוד האפליקציה שלנו המערכת מזהה שהמשתמש ביצע ועבר את מכסת הבקשות המותרת לו ליום:

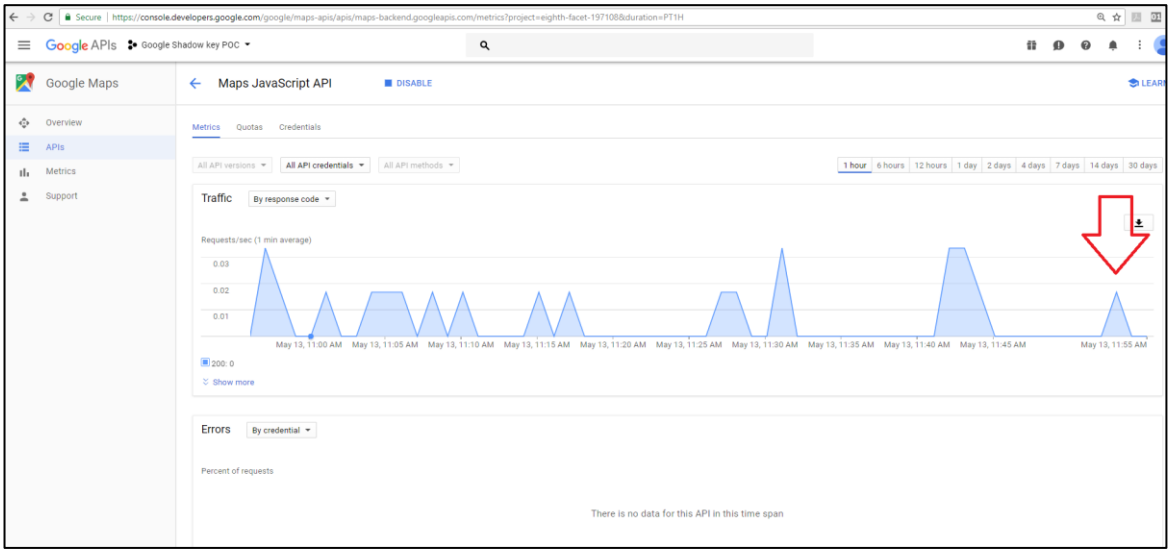


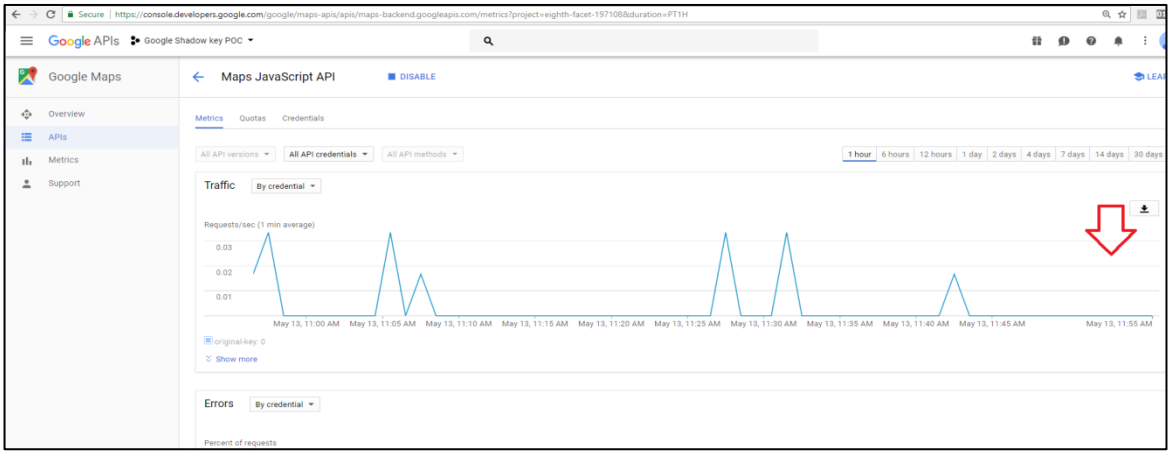


הקו האדום אשר מוצג בתצוגת המשתמש מעיד ומצביע על כך שעברנו את מכסת הבקשות המותרת לנו ליום:

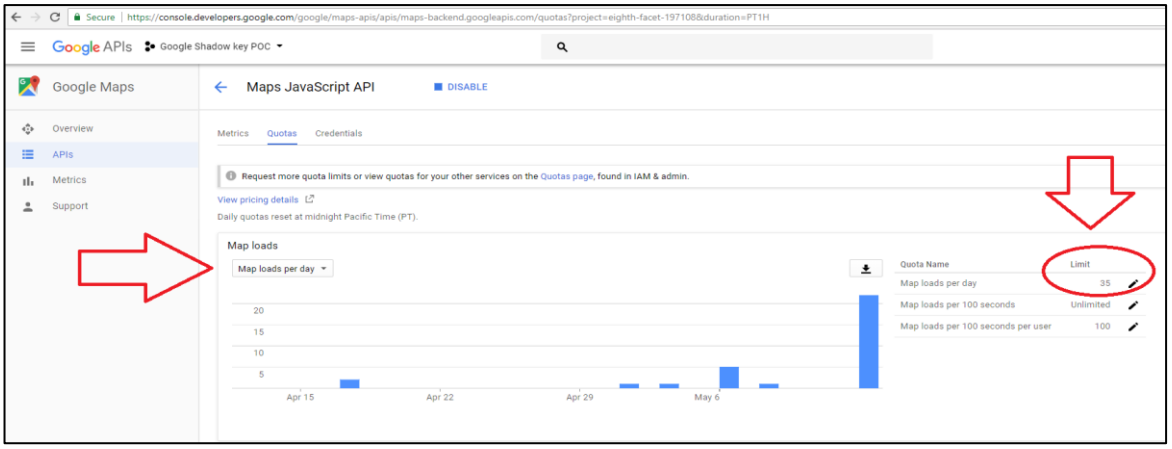


אך כאשר אנו משתמשים בקוד שלנו ב-"Shadow key" ניתן לראות שמערכת ה-API Console לא מזהה שעברנו את מכסת הבקשות המותרת לנו ליום שהגדרנו כעת להיות 35 בקשות ליום:





כפי שניתן לראות לא הוצג בתצוגת המשתמש קו אדום שמעיד על כך שעברנו את המכסה היומית של הבקשות:



הצלחנו לגשת ליישום, לעקוף את האיתור ולהימנע מתשלום - גם לאחר שחרגנו ממגבלת הבקשות היומית שלנו - מכיוון שה-"Shadow key" לא שייך למשתמש על פי הלשונית "Credentials" ב-Dashboard של בעל הפרוייקט אשר ב-API Console של גוגל.

סיכום

באמצעות מפתח API מקורי יחיד של גוגל הצלחנו למצוא "Shadow keys" שנוצרו באופן אוטומטי בגלל פגיעות במנגנון יצירת המפתחות של גוגל. מפתחות אלו אפשרו לנו להתחמק ממנגנון הזיהוי והתשלום של ה-Console API ולהימנע מביצוע תשלום לגוגל עבור מכסת בקשות שחרגה מהמותר.

על ידי מפתח מקורי יחיד תוקף יכול למצוא וליצור "Shadow keys", למכור אותם למשתמשים זדוניים אחרים, להביא להם את היכולת לקבל גישה לאותם יישומים ומידע כאילו השתמשו במפתח המקורי אך מבלי להתגלות על ידי מערכת הזהוי והתשלום ב-API Console של גוגל.

מעבדות CyberArk עמדו בכללי "responsible disclosure" והודיעו לגוגל על הפגיעות - אשר תוקנה לאחר מכן. כעת לא ניתן עוד למצוא וליצור "Shadow keys" על ידי שימוש בשירות "script loader" של גוגל.

בנוסף CyberArk בדקה ואישרה שה-"Shadow keys" שנוצרו לפני התיקון לא ניתנים לשימוש יותר וש-"Shadow keys" חדשים לא מיוצרים יותר על ידי מנגנון יצירת המפתחות של גוגל.

ציר הזמן של הדיווח

- 28/01/2018: קיום של "Shadow keys" דווח לגוגל על ידי חוקר ממעבדות CyberArk
- 09/03/2018: גוגל הגיבה שהפגיעות שדווחה על ידנו בבדיקה של צוות ה-API של גוגל
- 20/03/2018: גוגל אישרה ש-"Shadow keys" היא אכן פגיעות ונתנה פרס כספי לחוקר ממעבדות CyberArk במסגרת תוכנית התגמול למציאת באגים ופגיעויות של גוגל
- 03/04/2018: גוגל מציינת כי בעיה זו תטופל בעוד מספר שבועות
- 23/05/2018: החוקר ממעבדות CyberArk הבחין כי הפגיעות תוקנה
- 24/05/2018: גוגל הודיעה כי הם הוציאו תיקון עבור "Shadow keys"

מקורות

- https://www.googleapis.com/youtube/v3/videos?part=id&key=
- https://translation.googleapis.com/language/translate/v2/detect?key=
- https://www.google.com/jsapi?key=
- https://developers.google.com/maps/documentation/javascript/usage?hl=en_US
- https://support.google.com/googleapi/answer/7036164
- https://support.google.com/googleapi/answer/7035610
- https://developers.google.com/maps/pricing-and-plans/
- https://developers.google.com/maps/documentation/directions/get-api-key
- https://console.developers.google.com