

לינוקס וגשר USB ל-NVMe של Realtek RTL9210

סיכום:

- **תסמינים:** איפוס USB חוזרים ונשנים, שגיאות קלט/פלט, או דיסקים שנעלמים תחת לינוקס.
- **מושפעים:** Realtek RTL9210 (מאושר) ו-RTL9220 (אפשרי).
- **סיבה:** חזרה ל-ROM פנימי (f0.01) לאחר כשל בבדיקת סכום בדיקה.
- **השפעה:** חוסר יציבות קבוע, אין כלי עדכון מחדש זמינים עבור לינוקס.
- **תיקון:** רק כלי Windows של OEM יכולים לשחזר את הקושחה - Realtek חוסמת חלופות בקוד פתוח.

הקדמה

בשנת 2025, זה אמור להיות דבר סביר לחלוטין לאתחל Raspberry Pi מדיסק SSD המחובר דרך USB. עם זאת, הודות לגחמות הקושחה של Realtek, מטרה סבירה זו הפכה להרפתקה. לאחר חודשים של חוסר יציבות בלתי מוסבר - איפוסים אקראיים, דיסקים שנעלמים, מערכות קבצים פגומות - המחבר מיצה כל תיקון רגיל: כבלים חדשים, רכזות מופעלות, עדכוני קרנל, התאמות USB וכוונן קושחה. הפריצה הגיעה רק כאשר ChatGPT ענה על שאלה מוזרה בשעת לילה מאוחרת: "האם ייתכן שגשר ה-USB ל-NVMe חזר לקושחה ישנה?"

מבוא

אם מארז ה-NVMe מבוסס Realtek שלך הופך פתאום לבלתי יציב לאחר שבועות של פעולה חלקה - איפוס USB חוזרים, שגיאות קלט/פלט, או דיסקים שנעלמים - אתה לא לבד. התבנית הזו הופיעה במספר מותגים, מיחידות ללא שם ועד ל-OEM ידועים כמו Sabrent ו-Orico. המכנה המשותף: **שבבי גשר USB ל-NVMe של Realtek RTL9210 ואולי RTL9220.**

בהתחלה, הכל עובד. לאחר מכן, לכאורה ללא סיבה, המכשיר מתחיל להתנתק תחת עומס או במהלך שימוש ממושך, במיוחד במערכות לינוקס או Raspberry Pi. הסיבה האמיתית אינה ה-SSD או ספק הכוח - זהו בקר הקושחה עצמו שחוזר בשקט ל**קוד גיבוי המוטבע ב-ROM**, גרסה ש-Realtek עדיין מספקת באופן פנימי כ-f0.01.

המנגנון הנסתר - חזרה לקושחה לפי תכנון

שבבי הגשר של Realtek מאחסנים את הקושחה התפעולית שלהם ואת נתוני התצורה בזיכרון SPI חיצוני. בעת ההפעלה, הבקר בודק סכום בדיקה פשוט. אם סכום הבדיקה הזה אינו תואם, הוא מסרב לטעון את הקושחה החיצונית ומתחיל במקום זאת מה-ROM הפנימי שלו.

קושחת הגיבוי הזו ישנה ופגומה. חסרות בה מספר תיקוני יציבות USB ושיפורי ניהול מצב קישור שנמצאים בגרסאות מאוחרות יותר, מה שמוביל לרצף הקלאסי שכל משתמש לינוקס מכיר:

usb 3-2: איפוס מכשיר USB במהירות גבוהה מספר 2 באמצעות xhci-hcd
usb 3-2: קריאת תיאור המכשיר/64, שגיאה -71
אזהרת EXT4-fs (מכשיר sda2): שגיאת קלט/פלט בכתיבה לאינוד ...

סכום הבדיקה יכול להפוך לבלתי תקף כאשר נתוני התצורה נכתבים מחדש - למשל, כאשר הגשר מעדכן את הגדרות ניהול ההספק או UAS - והמכשיר מאבד חשמל באמצע הכתיבה. האתחול הבא מזהה סכום בדיקה פגום וחוזר באופן קבוע לקושחת ה-ROM.

בשלב זה, מארז ה-NVMe "בביצועים גבוהים" שלך מתנהג בדיוק כמו המארז הזול ביותר ללא שם, כי בפנים הוא כעת מפעיל את אותו קוד בסיס פגום שנצרר בסיליקון.

אימות הבעיה

אתה יכול לאשר מצב זה בקלות בלינוקס:

```
lsusb -v | grep -A2 Realtek
```

גשר Realtek תקין מדווח על גרסת קושחה (**bcdDevice**) מעל 1.00. גשר שחזר לאחר מראה:

```
bcdDevice f0.01
```

חתימה זו של **f0.01** פירושה שהבקר מתחיל מה-ROM - ושום כמות של ניתוק, פרמוט מחדש או כוונון קרנל לא יתקן זאת.

מנגנון החזרה הזה **אושר עבור RTL9210**. נראה כי **RTL9220** חולק את אותה ארכיטקטורת עיצוב ופריסת קושחה, כך שהוא עשוי להפגין התנהגות זהה, אך זה נשאר **סביר ולא מוכח**.

מדוע אינך יכול לתקן זאת בעצמך

באופן עקרוני, התיקון פשוט: לטעון מחדש את הקושחה הנכונה ל-SPI. בפועל, Realtek הופכת זאת לבלתי אפשרי.

החברה מספקת כלי עדכון סגור עבור Windows ל-OEM ולמשלבי מערכות. למשתמשי לינוקס לא מוצע דבר. מפתחי הקהילה ביצעו הנדסה הפוכה לכלי פלאש תואמים (**rtsupdater**, **rtl9210fw**), שאפשרו שחזור מלא של הקושחה ממערכות לינוקס - עד ש-Realtek הוציאה **הודעות הסרה של DMCA** כדי לדכא אותם.

אין הצדקה סבירה לזכויות קניין רוחני לחסימת כלים כאלה: הם לא חושפים מיקרו-קוד, אלא רק מתזמרים את רצף העדכון דרך USB. ההסרות של Realtek לא היו קשורות להגנה. הן היו אידיאולוגיות.

המחיר של אידיאולוגיה

זה לא עניין של אידיאליזם של קוד פתוח. זה עניין של **עוינות אידיאולוגית של ספק חומרה כלפי מערכות פתוחות** ששוברת מכשירים שמשווקים כ-**תואמי לינוקס**.

התנגדותה של Realtek לתיעוד וכלים פתוחים נמשכת כבר שני עשורים, ומשתרעת על Wi-Fi, Ethernet, שמע, וכעת בקרי אחסון. בידוד זה עשוי לעבור ללא שם לב בעולם של Windows בלבד, אך הוא הופך

לרעיל כאשר אותם שבבים משולבים במוצרים רב-פלטפורמיים כמו **Sabrent EC-SNVE**, שמציג בגלוי את לוגו לינוקס על האריזה שלו.

על ידי איסור כלי פלאש ללינוקס וחסימת תחזוקת הקהילה, Realtek למעשה **הפיללה תיקון עצמי**. ההשלכות מתפשטות החוצה:

- משתמשי לינוקס רואים חומרה "נתמכת" מתדרדרת לחוסר יציבות.
- OEM כמו Sabrent ו-Orico נתקלים בעלויות מיותרות של RMA ואחריות.
- המוניטין הוותיק של Realtek לתאימות לינוקס גרועה מתחזק שוב.

בסופו של דבר, לא הקוד הפתוח הוא ששובר את המכשירים של Realtek - **זו העוינות של Realtek כלפי קוד פתוח** ששוברת אותם.

דרך רציונלית קדימה

הפתרון אינו דורש שינוי אידיאולוגי, אלא רק פרגמטיזם. Realtek יכולה:

1. לשחרר כלי עדכון בשורת הפקודה החתום על ידי הספק עבור לינוקס (אין צורך בחשיפת קוד המקור).
2. לפרסם את האלגוריתם של סכום הבדיקה כדי שמשלבי מערכות יוכלו לאמת תמונות פלאש בבטחה.
3. לאמץ מצב בסגנון DFU שמקבל עדכונים דרך אחסון המוני USB, ללא תלות במערכת ההפעלה.

כל אחד מאלה ימנע עלויות אחריות, יגן על קשרי OEM וישיב את האמון בשבבי הגשר של Realtek בקרב משתמשי לינוקס מקצועיים - ממבני תחנות עבודה ועד מפתחי Raspberry Pi.

מה אתה יכול לעשות

אם אתה חושד שהמארז שלך חזר לקושחת ROM:

- בדוק עם `lsusb -v | grep bcdDevice`.
- אם מופיע **f0.01**, דווח על הבעיה ל-OEM שלך.
- כלול קטע מ-**dmesg** והפנה למנגנון החזרה המתועד הזה.
- בקש מהספק שלך להעלות את הבעיה עם Realtek, תוך ציון הצורך בכלי עדכון תואם לינוקס.

מדיניות הקושחה של Realtek לא רק מפריעה לחובבים; היא יוצרת הפסדים כספיים מוחשיים עבור המערכת האקולוגית שלה. ככל שהמציאות הזו תוכר מוקדם יותר בתוך החברה, כך יוכלו משתמשי לינוקס ושותפי OEM להפסיק לבזבז זמן על מחזורי RMA שניתן להימנע מהם.

תגובות היצרנים

גם Realtek וגם Sabrent הוזמנו לספק הצהרות בנוגע לבעיית החזרה לקושחה שתוארה לעיל. תגובותיהם - אם יתקבלו - יצורפו כאן.

נספח - זיהוי מכשירים מושפעים

בקר	מזהה ספק	מזהה מוצר	הערות	סטטוס
RTL9210	0x0bda	0x9210	גשר USB 3.1 Gen 2 10 Gb/s	מאושר התנהגות חזרה

בקר	מזהה ספק	מזהה מוצר	הערות	סטטוס
RTL9220	0x0bda	0x9220	גשר USB 3.2 Gen 2×2 20 Gb/s	אפשרי, ארכיטקטורה דומה

חתימת חזרה לקושחה: **bcdDevice f0.01**
גרסאות יציבות ידועות: **1.23 – 1.31**