

Linux ve Realtek RTL9210 USB-NVMe Köprüsü

Özet:

- **Belirtiler:** Linux altında tekrarlanan USB sıfırlamaları, G/Ç hataları veya kaybolan diskler.
- **Etkilenenler:** Realtek RTL9210 (onaylandı) ve RTL9220 (muhtemelen).
- **Neden:** Sağlama toplamı hatasından sonra dahili ROM'a (**f0.01**) geri dönüş.
- **Etki:** Kalıcı kararsızlık, Linux için yeniden flaşlama aracı mevcut değil.
- **Çözüm:** Yalnızca OEM Windows yardımcı programları ürün yazılımını geri yükleyebilir - Realtek açık kaynak alternatifleri engelliyor.

Önsöz

2025 yılında, bir Raspberry Pi'yi USB üzerinden bağlanmış bir SSD'den başlatmak tamamen makul olmalıydı. Ancak, Realtek ürün yazılımının tuhaflıkları nedeniyle, bu makul hedef bir maceraya dönüştü. Aylar süren açıklanamayan kararsızlıktan sonra - rastgele sıfırlamalar, kaybolan diskler, bozulmuş dosya sistemleri - yazar tüm olağan düzeltmeleri tüketti: yeni kablolar, güçlendirilmiş hub'lar, çekirdek güncellemeleri, USB ince ayarları ve ürün yazılımı optimizasyonu. Atılım, ancak ChatGPT'nin gece geç saatlerde sorulan tuhaf bir soruya yanıt vermesiyle geldi: "USB-NVMe köprüsünün eski bir ürün yazılımına geri dönmesi mümkün mü?"

Giriş

Eğer Realtek tabanlı NVMe kasanız, haftalarca kusursuz çalıştıktan sonra aniden kararsız hale gelirse - tekrarlanan USB sıfırlamaları, G/Ç hataları veya kaybolan diskler - yalnız değilsiniz. Bu desen, isimsiz birimlerden Sabrent ve Orico gibi tanınmış OEM'lere kadar birçok markada ortaya çıktı. Ortak payda: **Realtek RTL9210 ve muhtemelen RTL9220** USB-NVMe köprü çipleri.

Başlangıçta her şey çalışır. Sonra, görünürde bir neden olmadan, cihaz yük altında veya uzun süreli kullanım sırasında, özellikle Linux veya Raspberry Pi sistemlerinde bağlantıyı kesmeye başlar. Gerçek neden ne SSD ne de güç kaynağıdır - ürün yazılımı kontrolörü, Realtek'in dahili olarak hala **f0.01** olarak gönderdiği **ROM'a gömülü yedek koda** sessizce geri dönüyor.

Gizli Mekanizma - Tasarım Gereği Ürün Yazılımı Geri Dönüşü

Realtek köprü çipleri, işletim ürün yazılımını ve yapılandırma verilerini harici bir SPI flaşında saklar. Güç açıldığında, kontrolör basit bir sağlama toplamı kontrolü yapar. Eğer bu sağlama toplamı eşleşmezse, harici ürün yazılımını yüklemeyi reddeder ve bunun yerine dahili ROM'undan başlatılır.

Bu yedek ürün yazılımı eski ve hatalıdır. Sonraki revizyonlarda bulunan birkaç USB kararlılık düzeltmesi ve bağlantı durumu yönetimi iyileştirmelerinden yoksundur, bu da her Linux kullanıcısının tanıdığı klasik diziye yol açar:

```
usb 3-2: xhci-hcd kullanarak yüksek hızlı USB cihaz numarası 2 sıfırlanıyor
usb 3-2: cihaz tanımlayıcı okuma/64, hata -71
EXT4-fs uyarısı (cihaz sda2): inode yazma sırasında G/Ç hatası ...
```

Sağlama toplamı, yapılandırma verileri yeniden yazıldığında - örneğin, köprü güç yönetimi veya UAS ayarlarını güncellediğinde - ve cihaz yazma sırasında güç kaybettiğinde geçersiz hale gelebilir. Bir sonraki önyükleme, bozulmuş bir sağlama toplamı tespit eder ve kalıcı olarak ROM ürün yazılımına geri döner.

Bu noktada, “yüksek performanslı NVMe kasanız” en ucuz isimsiz kabuk gibi davranır, çünkü dahili olarak silikona kazınmış aynı hatalı temel kodu çalıştırır.

Sorunun Doğrulanması

Bu durumu Linux altında kolayca doğrulayabilirsiniz:

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

Sağlıklı bir Realtek köprüsü, ürün yazılımı revizyonunu (**bcdDevice**) 1.00'ün üzerinde bildirir. Geri dönen bir köprü şunu gösterir:

```
bcdDevice f0.01
```

Bu **f0.01** imzası, kontrolörün ROM'dan başlatıldığı anlamına gelir - ve hiçbir miktarda fişten çekme, yeniden biçimlendirme veya çekirdek ayarlaması bunu düzeltmez.

Bu geri dönüş mekanizması **RTL9210 için onaylanmıştır**. **RTL9220**, aynı tasarım mimarisine ve ürün yazılımı düzenine sahip gibi görünüyor, bu nedenle aynı davranışı sergileyebilir, ancak bu **muhtemel olmaktan çok kanıtlanmış değildir**.

Neden Kendiniz Düzeltemezsiniz

Prensipte çözüm basittir: Doğru ürün yazılımını SPI'ya yeniden flaşlayın. Uygulamada, Realtek bunu imkansız hale getirir.

Şirket, OEM'lere ve entegratörlere kapalı kaynaklı bir Windows güncelleyici sağlar. Linux kullanıcılarına hiçbir şey sunulmaz. Topluluk geliştiricileri, Linux sistemlerinden tam ürün yazılımı kurtarmayı sağlayan uyumlu flaş araçlarını (**rtsupdater**, **rtl9210fw**, **rtsupdater-cli**)

tersine mühendislik yaptı - ta ki Realtek, bunları bastırmak için **DMCA kaldırma bildirimleri** yayınlayana kadar.

Böyle araçları engellemek için makul bir fikri mülkiyet gerekçesi yoktur: mikro kodu ifşa etmezler, sadece USB üzerinden güncelleme dizisini düzenlerler. Realtek'in kaldırmaları koruma ile ilgili değildi. İdeolojikti.

Bir İdeolojinin Maliyeti

Bu, açık kaynak idealizmiyle ilgili değil. Bir donanım satıcısının **açık sistemlere karşı ideolojik düşmanlığı**, *Linux uyumlu* olarak pazarlanan cihazları bozar.

Realtek'in dokümantasyon ve açık araçlara karşı direnci yirmi yıldır devam ediyor, Wi-Fi, Ethernet, ses ve şimdi depolama kontrolörlerini kapsıyor. Bu izolasyon, yalnızca Windows dünyasında fark edilmeyebilir, ancak aynı çipler **Sabrent EC-SNVE** gibi çok platformlu ürünlere entegre edildiğinde toksik hale gelir; bu ürün, ambalajında açıkça Linux logosunu sergiler.

Linux flaş araçlarını yasaklayarak ve topluluk bakımını engelleyerek, Realtek etkili bir şekilde **kendi kendine onarımı suç haline getirdi**. Sonuçlar dışarıya doğru yayılır:

- Linux kullanıcıları, “desteklenen” donanımın kararsızlığa düştüğünü görür.
- Sabrent ve Orico gibi OEM'ler gereksiz RMA ve garanti maliyetleriyle karşı karşıya kalır.
- Realtek'in Linux ile zayıf uyumluluk konusundaki uzun süredir devam eden itibarı bir kez daha pekişir.

Sonuçta, Realtek cihazlarını bozan açık kaynak değil – onları bozan **Realtek'in açık kaynağa düşmanlığıdır**.

İleriye Dönük Rasyonel Bir Yol

Çözüm, ideolojik bir değişim gerektirmez, sadece pragmatizm gerektirir. Realtek şunları yapabilir:

1. Linux için satıcı tarafından imzalanmış bir komut satırı güncelleyici yayınlayın (kaynak kodu ifşası gerekmez).
2. Entegratörlerin flaş görüntülerini güvenli bir şekilde doğrulaması için sağlama toplamı algoritmasını yayınlayın.
3. İşletim sisteminden bağımsız olarak USB kitle depolama yoluyla güncellemeleri kabul eden bir DFU tarzı modu benimseyin.

Bunların her biri garanti maliyetlerini önler, OEM ilişkilerini korur ve iş istasyonu üreticilerinden Raspberry Pi geliştiricilerine kadar profesyonel Linux kullanıcıları arasında Realtek köprü çiplerine olan güveni yeniden inşa eder.

Ne Yapabilirsiniz

Eğer kasanızın ROM ürün yazılımına geri döndüğünden şüpheleniyorsanız:

- **lsusb -v | grep bcdDevice** ile kontrol edin.
- Eğer **f0.01** gösteriyorsa, sorunu OEM'inize bildirin.
- **dmesg** çıktısından bir alıntı ekleyin ve bu belgelenmiş geri dönüş mekanizmasına işaret edin.
- Tedarikçinizden sorunu Realtek'e iletmesini isteyin ve Linux uyumlu bir güncelleyiciye olan ihtiyacı belirtin.

Realtek'in ürün yazılımı politikası sadece meraklıları rahatsız etmez; kendi ekosistemleri için somut mali kayıplar yaratır. Bu gerçek şirket içinde ne kadar erken fark edilirse, Linux kullanıcıları ve OEM ortakları o kadar erken önlenebilir RMA döngülerinde zaman harcamayı bırakabilir.

Üreticilerin Yanıtları

Hem Realtek hem de Sabrent, yukarıda tarif edilen ürün yazılımı geri dönüş sorunuyla ilgili açıklamalar sağlamaya davet edildi. Alınan yanıtlar – eğer alınırsa – buraya eklenecek.

Ek – Etkilenen Cihazların Tanımlanması

Kontrolör	Satıcı Kimliği	Ürün Kimliği	Notlar	Durum
RTL9210	0x0bda	0x9210	USB 3.1 Gen 2 10 Gb/s köprü	Onaylandı geri dönüş davranışı
RTL9220	0x0bda	0x9220	USB 3.2 Gen 2x2 20 Gb/s köprü	Muhtemel , benzer mimari

Ürün yazılımı geri dönüş imzası: **bcdDevice f0.01**

Bilinen kararlı revizyonlar: **1.23 – 1.31**