CVE-2022-23968: Xerox vulnerability allows unauthenticated users to remotely brick network printers (UPDATED)

Posted on January 24, 2022 by **Mahmoud Al-Qudsi**

In the world of network security, it pays to always remember that many (if not most!) security bugs start off their lives as seemingly innocuous "regular" bugs, and it's only by diligently considering how aberrant behavior – say, incorrect results returned for particular inputs or a mere "stability issue" that turns out to actually be a use-after-free causing the observed crashes – could be abused by determined malicious actors that the underlying security implications become obvious. This has great benefits: for instance, it can be argued that it wasn't until Microsoft started taking BSoDs that could be triggered by unprivileged users seriously, recognizing them for the open backdoors most of them were, that Windows actually became usably stable.

Of course, then there are the bugs that have such blatantly obvious security implications that it would be hard to qualify them as wolves in sheep's clothing. Someone encountering such a bug, even if not particularly security-minded, would be forced to immediately recognize the risk they pose even if only because they have to deal with its consequences. This post is about such a

security bug that I encountered in the same vein as many others in the past: simply trying to do something completely unrelated and running into a vulnerability that made the task at hand that much harder.

In September of 2019, I ran into an issue while developing a one-click scan-to-print daemon that POSTed documents to the (by default, unsecured/unauthenticated) web interface on Xerox network printers. While the original project was working fully, the goal was to have scanned multipage documents print as multipage documents (rather than as several one-page documents) without introducing a PDF or PostScript dependency in the daemon, which was submitting the images to the printer's job queue as JPEGs at the time. Seeing that the Xerox web interface supported TIFF documents, my approach was to create and send a multi-page TIFF document and see if that was supported by the printer.[1] Given that the code was already handling the scanned images as JPEG and given that the TIFF document model actually lets you use TIFF as a container hosting non-TIFF (i.e. JPEG) data, I created a multi-page TIFF with each page set up as a TIFF Image File Directory hosting a page of the scanned document as a JPEG-compressed image.

I never got to test my idea as I was hasty in the coding stage and unwittingly attempted to test the daemon with a valid multi-page TIFF container, but one where the TIFF Image Directory wasn't correctly finalized (the pages within the TIFF were incomplete). Imagine my surprise – and anguish – when I discovered that the payload would seemingly permanently brick the network-attached printer! Here are the details of what happens:

The Xerox network brick vulnerability

1. The user/attacker submits a TIFF document with an incomplete Image Directory payload to the network printer. In addition to being done manually (by printing the document from a USB-attached PC or selecting the document to print via the web interface) this can also be done programmatically with a script that does the same, but over the network. What's more, **this is even doable over the web** via a JavaScript payload (easier if the IP address of the printer is known) because unauthenticated network users may POST payloads to the printer via bog-standard http(s) POST requests that aren't even secured by a nonce (which I'm generally grateful for in the name of scriptability and convenience) and for which current cross-origin mitigations are insufficient!
2. The printer, before it actually prints the document, opens the document to inspect its contents and to determine what resources are needed to complete the job (so it can request paper in the correct tray, count pages for accounting purposes, etc). In this case, **the TIFF handler in the Xerox firmware runs into unhandled/undefined behavior as it attempts and fails to parse the image directories within the TIFF container**, and the printer firmware panics, displaying a message to the user indicating that an unexpected error has occurred and that a hard reboot is required for the printer to resume working.
3. Upon reboot, the print queue manager attempts to resume the job at the front of the print queue, **which is still our buggy document**, runs into the same issue described in the previous point, and panics requesting a reboot once more.
4. This continues ad infinitum, as jobs are persisted to non-volatile memory and are not cleared when the unit is unplugged or restarted **effectively bricking the device**. The print queue

management (web or on-device) interface is inaccessible before the printer reaches the point where it tries to read the failed job and it is inaccessible after the panic, meaning there's no means via any of the available user interfaces for the print queue to be cleared to break out of this vicious loop.

Timeline of discovery and reporting

To the best of my knowledge, this vulnerability remains unpatched and continues to affect a number of Xerox printers across different product/model lines. This full, public disclosure is being made given the egregious amount of time that has elapsed since this issue was brought to Xerox's attention. The exact timeline of events (below) has been recreated from the emails I have regarding this issue:

1. I discovered this issue the week of September 23, 2019. After running into the initial issue and figuring out a tedious workaround that enabled me to recover my printer (see below), I attempted to narrow down the parameters of the exploit until I had a sub-kilobyte payload that could reliably reproduce the issue.
2. After confirming the security ramifications of the vulnerability, I first contacted Xerox via the automated security-related contact form on their United States website with a brief synopsis of the issue. To the credit of the security officer assigned to the case, I had a response within roughly an hour with the requested contact and disclosure information I had requested, while thanking me for my "responsible disclosure."
3. On September 26, 2019, I replied with the details of the exploit and provided a sample payload (attached to this post) that would trigger the network brick procedure. Half an hour later, the security specialist assigned to the case had replied:

> *I will turn this over to the proper development team and see what they can determine.* ***We'll let you [know]*** *as soon as we have our assessment done and some idea as to what we might plan to do to address the issue, and* ***we definitely will keep you apprised of our progress*** *on this.*

4. I did not hear anything back until I emailed requesting an update on January 14, 2020 (over three months later):

> *I have not heard back regarding this matter, but there have been new firmware releases since then.*
>
> *Was this issue addressed?*

5. I received a reply within the hour informing me that **the vulnerability was confirmed but still not fixed** for the Versalink line of printers and found to not affect the Altalink line of printers (which I'd included in the original report for sharing portions of the software stack):

> *We did finish our assessment and was able to confirm your results for VersaLink. However, we found that AltaLink is not vulnerable to this TIFF vulnerability.*
>
> *We have forwarded your results to the 3rd party vendor that developed the VersaLink software to develop a fix for this TIFF vulnerability but so far we have yet received it yet.*

Given that it has now been not 90 days but closer to two-and-a-half years since this issue was disclosed to Xerox Corp and I have not received any updates regarding the matter, I have decided to disclose this publicly (which I probably should have done much sooner but kept putting off for the same $reasons that make me put a lot of things off).

Vulnerability details

- CVE: CVE-2022-23968
- Class: Denial-of-service
- Severity: Critical (bricked, semi-permanent)
- Exploitable: Over the internet, over the local network, in person from a connected PC, in person at the device in question
- Privately reported: September 25, 2019 to Xerox product security
- Publicly disclosed: January 24, 2022
- Status: Confirmed by Xerox product security specialist and security researchers. Update: Xerox has released firmware upgrades version xx.61.23 (or higher) in response to this vulnerability.
- Affected devices: all Xerox Versalink business printers and copy machines, including the VersaLink B400, B405, B600, B605, B615, B70xx, C400, C405, C500, C505, C600, C605, C7000, C70xx, C8000, and C9000 series copy machines and printers. Additionally at least the Phaser 6510 and WorkCentre 6515 printer and copy machine have been reported to be affected as well; other Phaser and WorkCentre models are suspected to also be included.
- Affected firmware versions: tested against firmware version xx.42.01 and xx.50.61, confirmed by Xerox to affect all firmware versions up to xx.61.23 (which is the first version to include a patch for this vulnerability).
- Permissions required for in-person exploit: None (documents may be submitted over USB, LAN, or selected in-person from a USB stick)
- Permissions required for LAN exploit: None by default (as all LAN users are allowed to POST jobs to the handling endpoint over http(s) by default)
- Permissions required for exploit over the internet: None by default. The device's web interface exposes an http(s) POST interface that is not protected by any nonce and for which cross-site origin mitigations are useless as the response may be freely discarded. Only the device's name or IP address on the destination network is required, although even that is not required as it may be discovered via JavaScript given that the endpoint URL is fixed and IPv4 is enabled by default, limiting the possible search space.
- Discovered by: Mahmoud Al-Qudsi, NeoSmart Technologies

Proof-of-Concept

The contents of the following archive may be submitted to a vulnerable Xerox printer to trigger the remote brick: `xerox brick.rar` (345 bytes). The TIFF payload is also included in this base64-encoded attachment:

```
UmFyIRoHAQAzkrXlCgEFBgAFAQGAgAD5BbdHEwMC5QAE5QAA9kPUNIAAAANDTVRYZXJveCByZW1v
dGUgYnJpY2sgcGF5bG9hZCBieSBNYWhtb3VkIEFsLVF1ZHNpDQpTZWUgaHR0cHM6Ly9uZW9zbWFy
dC5uZXQvYmxvZy8/cD00ODY1IGZvciBtb3JlIGluZm8uAOsG2ysrAgMLjQEEvAMgCd+uuYADAA94
ZXJveCBicmljay50aWYKAwIA/Fsg4nPVAcISiiBENSb2YDSTz9+g+ofkEQVoaUFeJvK3kDY8WbGp
HgjY0bFPe8gzgjwjaJNmzSGzlGGm0ZRkySYEISicQttsKElCEti8EbSsdkcDz6/WmRz/N1o/EIEf
YPQUn+fPO4RLXjWeRbJT8isQTI5AnW6pF0WsD5DaxM4tgNHp3U7xR1fsHuvMYwMeDGyHIB13VlED
BQQA
```

I am withholding the sample code to exploit this over the network or over the internet at this time, although it is trivially implemented given the payload TIFF file above and some monitoring of outgoing requests in the browser inspector while normally submitting a job via the on-device web interface.

Workarounds and Mitigations

Immediate Mitigation:
As of January 28, 2022, Xerox has announced that devices upgraded to firmware versions xx.61.23 or greater will be protected against this vulnerability. Accordingly, please check xerox.com to see if a firmware update is available for your device and update immediately where possible.

For deployments of devices for which no firmware upgrade is available at this time or in environments where it is not possible to immediately deploy such a firmware upgrade, configure the devices to deny all print privileges to unauthenticated users, both over the network and in-person. Remove access to affected printers from any USB- or network-connected PCs or devices used by untrusted individuals.

Recovering Bricked Devices:
If there are any unapplied firmware updates, a network firmware update procedure may be initiated which will upgrade the device and in the process, clear the job queue. This breaks the device out of its reboot-panic loop. Otherwise, manually desoldering and wiping/reprogramming the storage module on the device mainboard is required to clear the job queue and unbrick the device. *It is likely – but not verified – that a Xerox field technician may be able to clear the NVRAM by initiating an undocumented startup sequence that bypasses the print queue or by jumping specific pins on the mainboard.*

Updates

Follow me on twitter @mqudsi and follow NeoSmart Technologies @neosmart for updates, or see below:

- **January 25, 2022:** This vulnerability has been assigned CVE ID 2022-23968
- **January 27, 2022:** Xerox appears to have posted then removed version 1.0 of special security bulletin XRX22-002 on their advisories page. It acknowledged the vulnerability but stated that firmware versions xx.50.61 (released October 2019) and later contained a fix for the issue. However in a January 2020 email from Xerox security division (see above), a Xerox security officer confirms that the latest firmware upgrade *did not* contain any such fix and the vulnerability remained unpatched.
- **January 28, 2022:** After reaching out to Xerox, a new version 1.1 of Xerox security bulletin XRX22-002 has been posted with the corrected information. As suspected, the claim that the fix was released in October of 2019 was retracted; per the updated document, firmware versions xx.61.23 and above are patched against this vulnerability. The sections on affected models, firmware versions, and suggested mitigations have been updated based on this information.

~~*This article will be updated with a CVE id when one is assigned.*~~ *Any other updates will be posted as they come to light.*

> *I'm reporting a zero-day that lets unauthenticated network users remotely brick Xerox printers. I reported this to Xerox in 2019 and decided it's been long enough. https://t.co/EZtnDz4AzC*
>
> *— Mahmoud Al-Qudsi (@mqudsi) January 24, 2022*

1. Baseline TIFF support requires that decoders are capable of handling multi-page TIFF documents, even if they are only capable of reading/displaying the first page. ↩

This entry was tagged with **network security**, **security**, **vulnerability**, **xerox**, **zero day** by **Mahmoud Al-Qudsi**.

Related Posts

- Implementing truly safe semaphores in rust
- SecureStore 0.100: KISS, git-versioned secrets management for rust
- C# file size formatting library PrettySize 3.1 released
- PrettySize 0.3 release and a weakness in rust's type system
- Easy Window Switcher 1.3.0 released