Attacking Xerox multi function printers

Raphaël Rigo

INFILTRATE 2020

AIRBUS

1. Intro
2. Xerox WorkCentre and AltaLink
3. Xerox VersaLink
4. Post exploitation
5. Tips for the Blue Team

Intro

Context

Who am I?

- Reverser since more than 20 years
- Security evaluation expert in Airbus

Airbus security lab missions include:

- Evaluating products that Airbus uses or sells to increase the company's overall security
- Red Teaming (Blue Team sparring partner)
- Check `https://airbus-seclab.github.io`, `@AirbusSecLab`

This presentation:

- Deep dive into Xerox VersaLink, WorkCentre and AltaLink models
- How I found previously unknown vulnerabilities in all models
- Tips for both the Red and Blue teams

Why target multi function printers (MFP)?

Easy target:

- Big companies have thousands of them
- Connected to the LAN, often not firewalled from the clients
- Most probably managed by a contractor (security policies?)
- Probably overlooked by the IT security team

Big MFP: AltaLink C8070

Why target multi function printers (MFP)?

Interesting for an attacker:

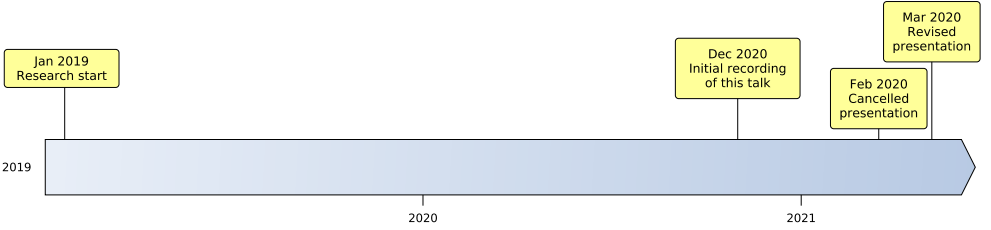
- Handle confidential documents (prints, scans)
- Active Directory (LDAP) credentials (nobody would use a *domain admin*, *right?*)
- Sometimes SMB access to some server (templates, scan to folder)
- Good place for persistence:
  - Often Linux based (tools work)
  - Rarely (if ever?) monitored by the SOC

Previous research

- O'Connor: Vulnerabilities in Not-So Embedded Systems (BHUSA 06)
- Heiland: From printer to pwnd (DEF CON 19)
- Weidenbach, Ernst: PWN Xerox Printers (. . . again)
- Müller: Printer Exploitation Toolkit (PRET)
- Romero, Rivas: Why you should fear your "mundane" office equipment (DEF CON 27)
- Probably others I'm missing, sorry

My story

- Read about the “Printer Exploitation Toolkit” on the week-end
- Tested it on the office’s printer on Monday morning. . .
- . . . Saw it doesn’t work
- Downloaded firmware from Xerox website for a “quick” look
- Got trapped in a reversing loop!
- . . .
- Then I discovered we had other models
- Trapped again!



Our attacker model

Prerequisites:

- Network access to admin interface
- No credentials

Objectives (descending):

- Steal all documents (printed, scanned)
- Establish persistence and relays
- Recover infrastructure creds (AD, email, etc.)
- Recover printer credentials

So the research will look for ways to move from anonymous to root.

Xerox WorkCentre and AltaLink

AIRBUS

Overview

Big machines (high-end models):

- WorkCentre 7835 (~2011): CPU: E500v2 PPC, 1.5GB RAM, 160GB HDD
- AltaLink 8030 (~2016): CPU: Intel Atom, 8GB RAM, 250GB HDD. WorkCentre evolution.

EAL2+ certified: doc gives a good overview of functionalities and attack surface

WorkCentre 7835 , 151 kg, >15k USD

AltaLink 8030, 151 kg, >15k USD

Attacker's view

Network attack surface:

- Xerox's *Information Assurance Disclosure Document* lists up to 24 open ports (!)
- In practice:
  - SNMP
  - HTTP: Web UI, Web Services and endpoints for centralized management
  - Printing protocols (LPR, IPP, 9100)

Functional attack surface:

- File format parsers (PDF, PS, PCL, XPS)
- Image parsers (embedded in PDF, etc)

Local physical attack surface:

- USB ports
- HDD access
- Maintenance access (serial, USB)

First step: extracting firmware updates

DLM File format:

- `tar.gz` with text header:
  - RSA signature
  - RSA-encrypted AES key in AltaLink case

Finding the AltaLink firmware RSA private key:

- Remove hard drive from printer
- Mount filesystem
- Find all private keys on disk

Decrypting the AltaLink firmware:

- Use RSA private key to recover AES-256 key
- Decrypt using AES-256-CBC and null IV

Note: research was done on firmwares released from 2017 to 2020, for both AltaLink and WorkCentre.

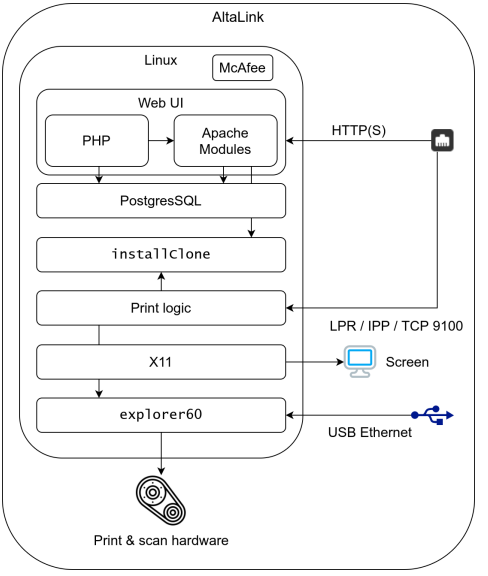
System architecture

Major components, split in firmware:

- OS: main Linux OS
- NC: network controller
- XUI: Xerox User Interface
- CCS_ACD: anti counterfeit detector
- CCS: copy control system

`exp55/explorer60`, 47MB ELF binary:

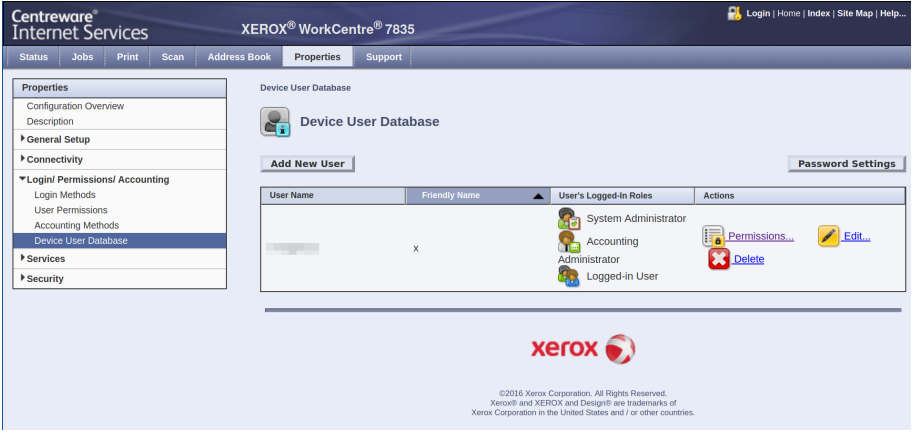
- Old code in charge of all the HW printing processing (scanning, paper path, printer, etc.)
- Old OS ported inside a Linux process?
- Supports TFTP connections: using Ethernet-over-USB port at the back



AltaLink components

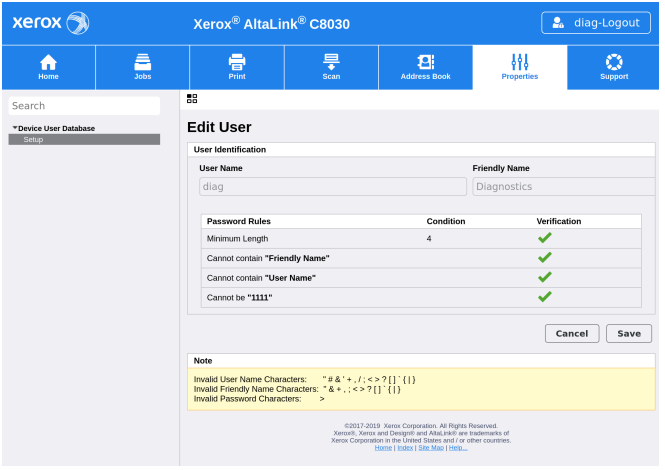
AIRBUS

Web UI: WorkCentre



WorkCentre

Web UI: AltaLink



AltaLink

Web UI: authentication

Basics:

- Auth is mandatory on AltaLink to access the Web UI
- Users authentication options:
  - Local: on device DB
  - Network: Kerberos, SMB, LDAP

Roles:

- Normal user
- Accounting administrator: accounting features only
- System administrator: full admin

System security model:

- Everything that runs is signed by Xerox
- No shell access
- Having `admin` should *not* imply code execution on the device

But having `admin` *definitely* increases the attack surface :)

Local accounts: defaults, encryption

User list is stored in PostgreSQL:

- Running as `postgres` user, not accessible from `nobody` user
- `ess` database, `xsa.accounts` table
- Default content (simplified):

```
admin           Admin          0x0D9D2F4DB1E2510D510851487331104EAA    SA
!$ecivreS       CSE Account    0xF399C25961A4A27F8FB0EE2FAA430DBCAA    CSE
diag            Diagnostics    0x9FA43E24372828D55A033E3CDF0D3957AA    CSE
forceonboxlogin Force[...]     0x8FAE69757CEBCEC23B8FE1E75CA948EAAA    CopyOnly
guest           Guest          0xCE617F371AE3817983F214EB7F41C6B9AA    GUEST
```

`esscrypto_encryptString`

Used a *lot* in various places to:

- Hide encryption keys
- Encrypt users passwords

Default and hardcoded accounts (2019 picture)

Decrypting passwords:

- Recover passphrase from call to `esscrypto_encryptString`
- Decrypt using AES-256-CBC and null IV

Five default accounts:

- `admin` / `1111`: default admin password
- `diag` / `3424`: hidden “service” account, documented in service manuals (Remote UI access)
- `!$ecivreS` / `2732`: hidden “service” account, documented in service manuals (local only)
- `forceonboxlogin` / `password`: hidden “technical” account, introduced in 2018
- `guest` / `2222`: hidden “technical” account, introduced in 2018

“Service” accounts?

- Explicitly hidden by UI code
- Password is impossible to change

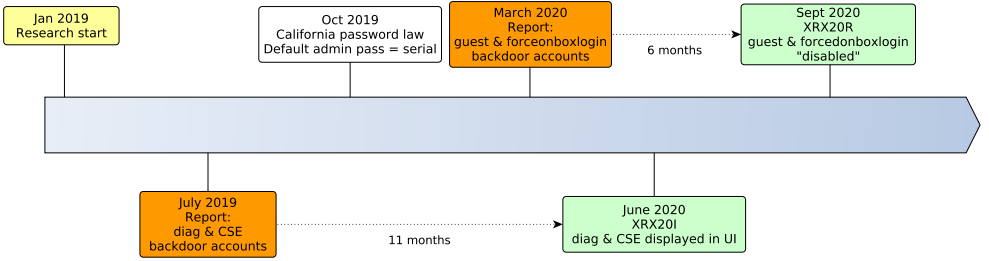
Security improvements as of XRX20-R (Sept 2020)

Account privileges and passwords:

- Some accounts are now only valid for physical access (local UI):
  - `!$ecivreS`, `forceonboxlogin` and `guest`
- Some accounts are now *really* unprivileged:
  - `diag`: can access remote UI (with diagnostics access) if enabled
  - `forceonboxlogin`
- Newer firmware comply with 2020 California Password law (SB-327) and use the serial number by default for `admin` and `diag`.
  - `snmpget -v 1 -c public HOST 1.3.6.1.4.1.253.8.53.3.2.1.3.1`
  - `curl https://HOST/diagnostics/SESStatus.txt | grep Serial`
- Future releases should:
  - randomize the passwords of `guest` and `forceonboxlogin` which are “internal” accounts
  - allow the password for `diag` to be changed

Passwords remain stored without hashing because `WS-Security` needs the password itself.

Accounts and backdoors progress



Web interface “internals”

High level overview:

- Runs as `nobody`
- UI in PHP (completely reworked in AltaLink):
  - (mostly) out of scope for this presentation/research
- A lot of the logic is actually implemented in C/C++:
  - `POST` to `/userpost/xerox.set` and `/dummypost/xerox.set`
  - → Apache modules `mod_loapost.so`, `mod_loaget.so`, `mod_upload.so`, etc.
- see O’Connor’s talk at BHUSA 06 for more info

`mod_loapost.so` extract:

```
Add_HTML_Set_fn("HTTP_Set_Http_Settings_fn", &HTTP_Set_Http_Settings_fn, 1);
Add_HTML_Set_fn("HTTP_Set_Diag_Log_Levels_fn", &HTTP_Set_Diag_Log_Levels_fn, 1);
Add_HTML_Set_fn("HTTP_Retrieve_Diag_Data_fn", &HTTP_Retrieve_Diag_Data_fn, 1);
Add_HTML_Set_fn("HTTP_WS_Reset_IP_Lockout", &HTTP_WS_Reset_IP_Lockout, 1);
```

Past vulnerabilities in PHP code:

```
https://HOST/diagnostics/diagnosticsAjaxHandler.php?command=viewLog&logName=;CMD_TO_EXEC;#
https://HOST/properties/accounting/download_csv.php?generated=../../TARGETFILE
https://HOST/ajax/fileDistributionRequestHandler.php?[...]&urlHeader=http://localhost;CMD_TO_EXEC;"
```

Finding vulnerabilities in the Apache modules

Classic/trivial methodology:

- Look for `.so` importing `system`, `popen`, `execve`, etc.
- Open in IDA
- Find vulnerable code

Remote command injection vulnerability:

```
curl -c cookie.jar -k
     -H 'Content-Type: application/x-www-form-urlencoded'
     --data "_fun_function=HTTP_Set_Diag_Log_Levels_fn"
     --data "&NextPage=%2F&CSRFToken=$csrf"
     --data-urlencode "http;$cmd;.lp" "https://$host/dummypost/xerox.set"
```

Assigned CVE-2019-10880:

- Xerox bulletins XRX19C, XRX19E, XRX19G, XRX19I, XRX19J, XRX19K, XRX19L and XRX19M.
- Fix: removed `HTTP_Set_Diag_Log_Levels_fn`

More Web vulnerabilities: privesc

Privilege escalation in AJAX handlers:

- `/ajax/cfgSetAjaxHandler.php?command=configServerSet` did not check the user's rights before setting conf entries
- Exploit:
  - Use `diag` account
  - Set `deviceAdmin.cloneDlmDownload.DLMPolicy=3` to allow unencrypted clone files submission
  - Submit clone file with new administrator password

Found while diffing XRX19-AQ (Unauthenticated RCE).

More Web vulnerabilities: SQLi

Story time:

- Xerox sent me a pre-release firmware to check some fixes
- Would not give me the admin password of the test printer
- . . . I decided I'd find a way to recover it

SQL injection in account management page:

- Found by looking for non-parametrized queries
- Exploit using `forceonboxlogin` and leak the admin password

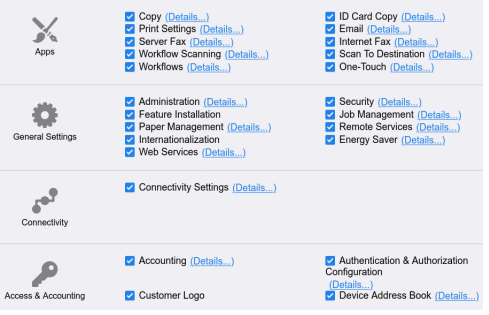
Fixes:

- Both were fixed in XRX20-R, Sept 2020

Clone files

Config backup you can apply to another printer (directly on TCP/9100!). Contains:

- Users configuration (including passwords)
- Network configuration (including passwords)
- Filtering configuration (`iptables`)
- ...



Clone options

Clone files: format

On WorkCentre:

- The clone file is an unencrypted `.tar.gz`
- The files are not encrypted

On AltaLink: `tgz` is encrypted using RSA+AES:

- Random AES key
- Encrypted with a public key. The private key is shared by all AltaLink.
- Sensitive data such as user accounts are also encrypted *inside* the clone file:
  - `.encrypted` extension
  - AES-256-CBC
  - Hardcoded hex key (`ess_crypto_get_hex_string_from_string`)

Network clients credentials

Various passwords are stored in configuration / clone files

- LDAP
- POP3
- SMTP
- SMB
- etc.

Example in `LDAPConfigAttributes.1.1.0`: `ldap.password=0xD7...AA`

Decrypting passwords:

- `esscrypto_encryptString` is used too, but with a different key
- hopefully interesting Active Directory accounts can be found here

Clone files RCE

Clone files restoration workflow (`installClone`):

- `tgz` is decrypted then extracted
- “Regular” configuration entries are applied
- Specific helpers are used in some cases (as root)

`iptables` command injection:

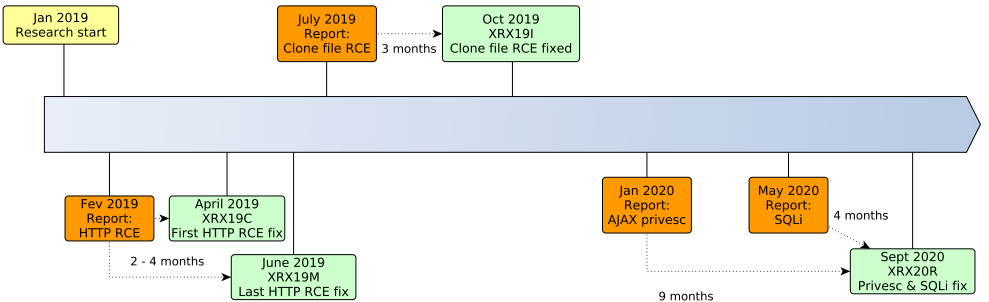
- Add an `iptable` rule in the right file, which passes the basic format check, such as:

`bash path_to_script -p all -j DROP -A INPUT -i eth0 -s 64.64.64.64/32`.

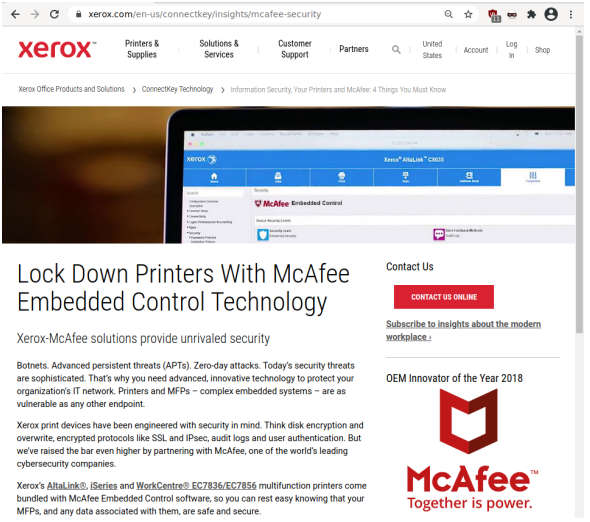
- A script can be embedded in the `AAP/fdi_card_access.png` file in the `tar.gz`. So the following will be executed as `root` when the clone file is loaded:

```
bash /path/to/AAP/fdi\_card\_access.png
 -p all -j DROP -A INPUT \
 -i eth0 -s 64.64.64.64/32
```

Xerox coordinated disclosure timeline



“Defense in depth”: McAfee Embedded Control



“Defense in depth”: McAfee Embedded Control

Host Intrusion Prevention System:

- Logs access to some files
- Prevents read access to some files, including the printer’s config
- Prevents write access to most important paths

Getting around it:

- Disable it through the WebUI (removed in XRX20L)
- Don’t care about it, it’s useless, as seen in `solidcore.conf`

```
CapabilityRules = {
    "0x16\"UPDATER: AUTO_105\"\"/etc/rc.d/init.d/cc_system\""
[...]
```

`cc_system` launches everything, including the web server. So all exposed binaries are authorized.

“Defense in depth”: Hardening

Applications were not compiled with basic exploit mitigations:

RELRO	STACK CANARY	NX	PIE	RPATH	RUNPATH	Symbols	FORTIFY
No RELRO	No canary found	NX enabled	No PIE	No RPATH	No RUNPATH	No Symbols	No

Linux kernel versions:

- WorkCentre: 2.6.34
- AltaLink: 3.10.62

Xerox said they backport security fixes.

AltaLink: Wrap-up

I *currently* do not have a way to remotely compromise an up-to-date AltaLink configured with a strong admin password.

All bulletins with Airbus acknowledgment for AltaLink and WorkCentre:

- XRX21F.
- XRX20G, XRX20I, XRX20R, XRX20X.
- XRX20L, XRX20M, XRX20V.
- XRX19AI, XRX19AP.
- XRX19C, XRX19E, XRX19G, XRX19I, XRX19J, XRX19K, XRX19L, XRX19M, XRX19Q.

Xerox VersaLink

Overview

Info:

- HW: 1 GHz dual-core ARM, 2GB RAM, optional HDD
- C405: 33 kg, ~700USD
- Linux 3.10.62

Xerox, really? Actually *Fuji* Xerox:

- Japanese comments
- Software architecture is completely different from other Xerox printers

System architecture

Software components:

- VxWorks:
  - Could not dig into it, handles the “Marking Engine” (Xerox’s *IAD* states v6.8.2)
- Linux:
  - User-facing OS: handles the network, printing, Web, etc.
  - “Special boot mode” OS (maintenance mode)

Exposed attack surface:

- Network: up to 16 listening ports in TCP/UDP (cf. *IAD*)
  - In practice: SNMP, HTTP, printing protocols
- Local: USB, maintenance access

Some positive security points:

- HW Root of trust
- Disk encryption

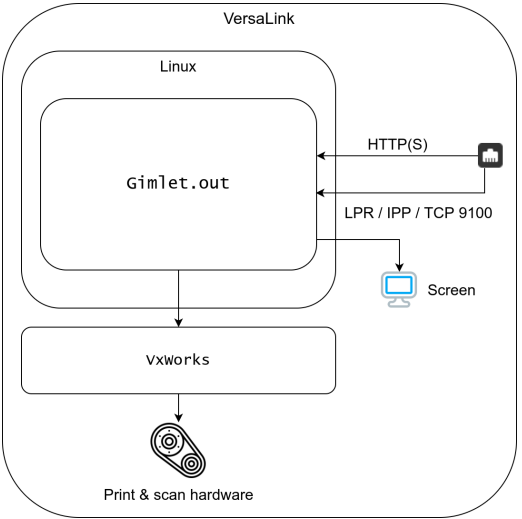
Main binary: `GimLet.out`

Some statistics:

- ~120MB (depending on model)
- 250k functions
- Biggest function is 150KB

Functionalities:

- Handles *everything*:
  - Scanning, printing
  - SMB client
  - Web Server
  - etc.
- Runs as `root`



VersaLink components

Reverse engineering

IDA struggles a bit:

- Partly written in C++
- “no” Xrefs because IDA does not handle this pattern:

```
.text:041384CC          MOV             R0, #0xF310
.text:041384D0          MOV             R1, #0xF740
.text:041384D4          MOVT            R0, #0x7A3
.text:041384D8          MOVT            R1, #0x7A3
```

- Ask Hex-Rays support -> will not be implemented (it was in 7.5)
- Must use the decompiler to have meaningful info

Methodology:

- Look at strings
- Produce full C file
- `grep`
- Look at decompiled code

Backdoor URLs

`https://HOST/backdoor.cmd`:

- Recover information on printing jobs
- Get job memory dump: lot of info leaking
- Get job file:
  - directory traversal allows anyone to recover small files (up to ~200KB) on the filesystem

`https://HOST/nananana/nananana.cmd`:

- Generate and recover diagnostic logs
- LOTS of information:
  - Internal OS info
  - Crashes, including registers and stack traces
  - etc.

Both require authentication, with an hardcoded account

Buffer overflows in HTTP server

Vulnerabilities:

- Several classical stack based buffer overflows in the backdoor argument parsing:

```
if (get_arg_value(v2, "TYPE", &arg) != -1 &&
    atoi(&arg.value, &type_val) == -1) {
    http_res_code = 400;
    strncpy(s, arg.value, arg.len); /* <-- overflow here */
...
```

- Small problem for the attacker: the overflowing data *must* be valid UTF-8

Exploitation:

- No stack cookies
- No ASLR
- Huge binary
- It's possible to find UTF-8 valid addresses to do a `ret2libc` attack:
  - `bash` connect-back

Clone files

Characteristics:

- Must be submitted through the web UI, after authentication
- AES encrypted Zip, with a hardcoded key and a random IV
- Configuration is stored in AES-256-CBC encrypted JSON files (hardcoded key and IV)

Once decrypted:

- passwords are in plain text

Command injection vulnerability:

- Some file names present in JSON files are handled without sanitization
- Shell command injection as root
- A big payload can be included in the Zip file

Xerox coordinated disclosure

Disclosure status:

- Reported all vulns (backdoor, BOF, Clone files RCE) in July 2019
- Pre-release sent in April 2020:
  - *Disables* the backdoor by default
  - Fixes the buffer overflows
  - Invalid fix for the clone files RCE
  - Adds firmware encryption
- Final release XRX20-K in June 2020 (a year later)

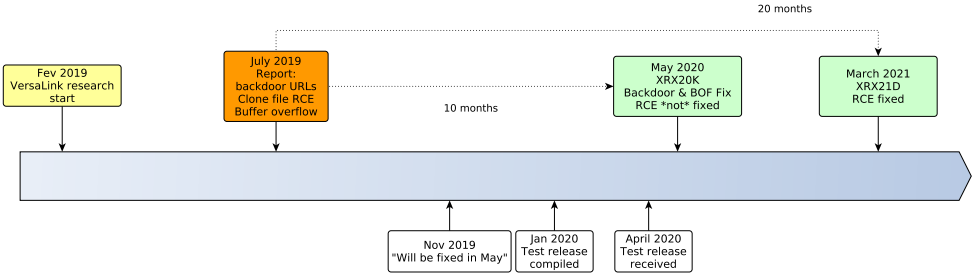
Clone file RCE:

- Finally fixed on March 5 2021 (XRX21D).
- Insecure `system` call to `unzip` replaced by a full *embedded* copy of `unzip`

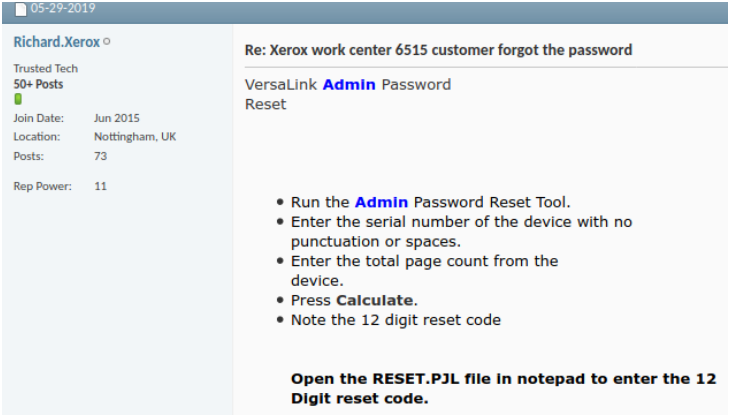
Fixing woes:

- All the development is done by Fuji
- Xerox is just a proxy

VersaLink timeline



Oh, one last thing: one more backdoor!



Forum post on VersaLink reset

RESET.PJL? admin password reset.zip?

`Admin Password Reset Tool.exe`:

- Checks if some Xerox software is present
- Coded in Visual Basic 6

`RESET.PJL`:
`@PJL SET JOBATTR="@RSAP=XXXXXXXXXXXX"`:

- Reverse `RSAP` handler in `GimLet.out`
- Trivial arithmetic to compute unlock code

→ Python script which uses SNMP to get printer counter and serial and resets the admin password to `1111`

Post exploitation

Documents exfiltration

Increased challenge: *secure erase* function

- Two modes: scheduled or right after document print/scan
- Actual file data is overwritten several times in as soon as the job is complete

Stealing docs on the fly:

- Need to win the race against secure deletion:
  - File is overwritten, not just deleted
  - Actual data copy is needed
- How?
  - `inotify` to watch for creation
  - Multi-threaded file copy
  - Use SMTP to send documents
- On-disk file formats:
  - Mostly PDF/PS
  - Proprietary compressed format for VersaLink scans (not RE'd yet)

Tips for the Blue Team

Securing them

Basic stuff:

- Restrict access to administration ports (Web, SNMP) from the LAN
- Disable “installing clone files through printing”
- Disable “Remote control panel” access or restrict it to admins only
- Change the default password, use network auth if possible (Kerberos)
- Use secure protocols: SNMPv3, TLS, etc.
- Keep them up to date!
- Collect logs (SFTP cron, or through fleet management) and check for:
  - backdoor accounts logins
  - password resets (VersaLink)
  - clone events
- Read and apply the security guide

Wait for Xerox to improve security:

- Security improvement planned in future products:
  - Password hashing
  - Hardware root of trust
- Tell them you care about (actual) security!

Last words

Time spent:

- Initial research: 40 person-days
- Diffing of updates and exchanges with Xerox: ~15 days
  - Check out diffware for a really useful tool by Jean-Romain Garnier

But there is still a *lot* to look at.

Big thanks to my colleagues for their contributions to the research:

- Benoît Camredon
- Xavier Mehrenberger
- Julien Lenoir
- Jean-Romain Garnier

Contact:

- Mail: raphael.rigo_at_airbus.com
- Twitter: `_trou_` (personal account)

References

- Xerox security bulletins: `https://security.business.xerox.com/en-us/documents/bulletins/`
- Security information for AltaLink C8xxx: `https://security.business.xerox.com/en-us/products/altalink-c8000-series/`
- AltaLink B8090 service manual: `https://yadi.sk/i/ckgNGb78d94Cxg`
- Security information for VersaLink C405: `https://security.business.xerox.com/en-us/products/versalink-c405/`
- VersaLink C405 service manual: `http://files.xdigital.com.br/Manuais/Xerox/C400/VersaLink%20C400_C405%20Service%20Manual.pdf`