Tear Down this Zywall!

Breaking Open Zyxel Encrypted Firmware
AGENDA

○ Introduction
○ How it Started
○ The Firmware
○ The Hardware
○ The Software
○ How it’s Going
○ Future Work
○ Questions
INTRODUCTION

Who’s this moron?
whoami

- LAGORIO\Jay (jay@lagor.io)
- B.S. Computer Science, UMBC (2008)
- CISSP, various SANS certifications over the years
- Windows Developer, Security Consultant, Licensed Private Investigator
- Talks at Hackfest.ca 10, DC27 Biohacking Village, Kernelcon 2
- Unrepentant nerd
HOW IT STARTED

There’s backdoors in them thar hills
IT WAS DECEMBER 2020

Security News <security_news@zyxel.com.tw>
Zyxel New Release to Elevate your Network!

If there are problems with how this message is displayed, click here to view it in a web browser.

Thank you for choosing ZyWALL Series. Zyxel Networks maintenance release will be available for major models as part of a security improvement process. This ZLD4.60 patch 1 release enhances all-round functions of ZyWALL products including:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>CVE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enhancement</td>
<td>Undocumented user account</td>
<td>CVE-2020-29583</td>
</tr>
<tr>
<td></td>
<td>vulnerability fix.</td>
<td></td>
</tr>
<tr>
<td>Enhancement</td>
<td>Enhanced HA Pro reliability</td>
<td>-</td>
</tr>
</tbody>
</table>
IT WAS DECEMBER 2020
IT WAS DECEMBER 2020

Undocumented user account in Zyxel products (CVE-2020-29583)

Niels Teusink / Cybersecurity Specialist

TL;DR: If you have a Zyxel USG, ATP, VPN, ZyWALL or USG FLEX you should update to the latest firmware version today. You can find the full list of affected devices here and the Zyxel advisory here.

Zyxel is a popular brand for firewalls that are marketed towards small and medium businesses. Their Unified Security Gateway (USG) product line is often used as a firewall or VPN gateway. As a lot of us are working from home, VPN-capable devices have been quite selling well lately.

When doing some research (rooting) on my Zyxel USG40, I was surprised to find a user account 'zyfwp' with a password hash in the latest firmware version (4.60 patch 0). The plaintext password was visible in one of the binaries on the system. I was even more surprised that this account seemed to work on both the SSH and web interface.
IT WAS DECEMBER 2020

100,000 – 300,000 Potential Devices
. THE Firmware

We’ll just binwalk it and call it a day
THE Firmware
THE FIRMWARE
We know it’s a Zip file
The file is password protected
We can see what files are inside
Some files are available online
The update process doesn’t require a password
7zip says there’s a trailer
Default passwords didn’t work

Zip Plain-Text Attack
- Attack from 1994
- Given a ZipCrypto archive and one file from inside the archive
- Derive the keystream for the rest of the archive
- Now you have the decrypted archive
THE FIRMWARE

INSERT DIRLIST OF FTP.ZYXEL.COM
Security Announcement - Change FTP Service to HTTPS Download Service

Tobias Hermanns
6 days ago · Updated

Due to security concerns and the modern browsers not supporting FTP (for example Chrome or Firefox) anymore, we decided to change our FTP server to a download library based on HTTPS.

This change will take effect by March 2021.

We will stop providing FTP services from March 2021 due to security concern. Users will be unable to download materials (including firmware, datasheet... etc.) from ftp://ftp and ftp://ftp2.

We will replace ftp://ftp.zyxel.com/ and ftp://ftp2.zyxel.com/ with our Zyxel Download Library, so users will still be able to download materials via HTTPS.
## The Firmware

Files for ftp.zyxel.com

<table>
<thead>
<tr>
<th>Name</th>
<th>Last modified</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>↑ Go to parent directory</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2014.06.ftp.zyxel.com.tar (View Contents)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2014.06.ftp.zyxel.com.tar.txt</td>
<td>09-Jun-2014 10:42</td>
<td>58.2G</td>
</tr>
<tr>
<td>ftp.zyxel.com_archive.torrent</td>
<td>05-Oct-2016 04:11</td>
<td>293.2K</td>
</tr>
<tr>
<td>ftp.zyxel.com_files.xml</td>
<td>17-Mar-2022 14:19</td>
<td>1.7K</td>
</tr>
<tr>
<td>ftp.zyxel.com_meta.sqlite</td>
<td>09-Jun-2014 11:07</td>
<td>9.0K</td>
</tr>
<tr>
<td>ftp.zyxel.com_meta.xml</td>
<td>17-Mar-2022 14:19</td>
<td>796.8B</td>
</tr>
</tbody>
</table>
THE Firmware

- Used github:kimci86/bkcrack to run the attack
- This didn’t work
- Maybe I don’t have the right files?
- One more strategy to try
Password Cracking
- Both LostPassword and JohnTheRipper
- Touchy about that trailer, had to strip it
- Gave each a couple days while brainstorming
- Total failure. Cool.
Are ya winning son?
THE Firmware

[@jaylagorio // jay@lagor.io]
THE HARDWARE

Crackin’ the case
THE HARDWARE
THE HARDWARE

Samsung K9F1G08U0 NAND Flash Chip
THE HARDWARE

- 10 pin connector
- PHISON PS2251-50-F
- ???

@jaylagorio // jay@lagor.io
THE HARDWARE
The drive mounts – we have a disk!
Use Win32DiskImager to image the entire drive
Throw the image in BinWalk
And out pops the firmware in plain-text!
...but only the firmware on THIS device.
But decrypting arbitrary firmware files?
THE SOFTWARE

Decompilers are free now!
THE SOFTWARE
THE SOFTWARE

- Device runs an Apache web server
- Submits to `firmware_upload-cgi` in `cgi-bin`
- Contains string `/util/zld_fsextract`
  - File System Extract?
- Let’s Google it
Seems to be

quote:

3.00(AQU.0) / 2012-02-17 16:50:38

You can always compare the .bin files once it shows on US ftp servers.

I would not worry that this is some kind of fake. USG firmwares are zipped with very strong password. The `zld_fsextract` update command then verifies the zip and unzips is before applying the update. For somebody to fake the firmware they must know how to generate the proper password (each update has different password) ... just watch console output during update if you’re curious.

2012-Mar-2 9:10 pm
○ The binary’s from a previous model though...
  □ Same processor architecture
  □ Firmware formats appear the same
  □ If it ain’t broke, why fix it?

Spoiler: the newer models have the same binary so we made a smart bet
THE SOFTWARE

○ Run file
  □ ELF 32-bit MSB executable, MIPS, N32 MIPS64 rel2 version 1 (SYSV), statically linked, stripped

○ Run strings
  ○ usage: `%s <input zip file> {-t | <unzip path> {-s <SET_EXTRACTION_COMMAND> | [-f file1 [-f file2 ...]] [-d dir1 [-d dir2 ...]] [-i]} [-D destDir]}
  ○ 7za
THE SOFTWARE

GHIDRA
THE SOFTWARE
○ Qemu Usermode Emulation for MIPS to run it
  □ sudo cp ld-2.3.3.so /lib/
  □ sudo ln /lib/ld-2.3.3.so /lib/ld.so.1

○ Run it with `strace` to find calls out to 7za
  □ qemu-mipsn32 -strace zld_fsextract 460AAKY1C0.bin -i &2>1 > out.txt
23402 open("460AAKY1C0.bin", O_RDONLY) = 3
23402 ioctl(3, 21517, 1082130192, 0, 1082130828, 0) = -1 errno=25 (Inappropriate ioctl for device)
23402 brk(0x10136000) = 0x10136000
23402 lseek(3, -12, 2, 269700176, 1082130828, 1) = 136687321
23402 Linux(3, 1082130448, 4, 269700112, 1082130828, 1) = 4
23402 Linux(3, 1082130452, 4, 269700112, 1082130828, 1) = 4
23402 brk(0x10155000) = 0x10155000
23402 lseek(3, -126860, 2, 269704280, 1082130828, 1) = 136560473
23402 Linux(3, 269704288, 126848, 269700112, 1082130828, 1) = 126848

Properties

| Size       | 140 036 412 |
| Packed Size| 136 477 581 |
| Folders    | 16          |
| Files      | 470         |
| CRC        | E6224753    |
| Path       | \wsl\Ubuntu-20.04\home\ubuntu\462AAKY0C0.bin |
| Type       | zip         |
| Warnings   | There are some data after the end of the payload data |
| Physical Size | 136 575 209 |
| Tail Size  | 126 860     |

@jaylagorio // jay@lagor.io
THE SOFTWARE

○ qemu-mipsn32 -strace zld_fsextract 460AAKY1C0.bin -i &2>1 > out.txt

○ 23408 execve("-i","-i","-o","-q","-P", "iRqoEoQpAROlW9ksF4XgFhOiyehh6YVMMxCYaO0ohnKsXC8emL8", "460AAKY1C0.bin","db/etc/",NULL)) = -1 errno = 2 (No such file or directory)
// printarg4.c

#include <stdio.h>

int main(int argc, char* argv[]) {
    printf("%s\n", argv[4]);
    return 0;
}

THE SOFTWARE
HOW IT’S GOING

We fucked around, now let’s find out!
THE BUG

○ It turns out my backup of the firmware got corrupted.
  □ Are you testing your backup strategy?
  □ No? Then do you actually have one?
○ This version of the firmware is completely gone.
○ What to do?
○ I extracted it like this, and since we already know the IOCs we can grep for them.
THE BUG

```
zyfwf: $5$cU1FaqIM$BQAz68B0$LnPwuAEpbc1MsWzbG2v/r37AcSJWI7pA2ZIkDiWnZ6dTBSMGK/5m7hOs5gPsQs80/1aRcsc4yYEvqoKsJ3EPyYKuxd+puMJY1/qQzvKARhE1kPTfzox0pryfm9dK5WDPa0aNiIjmjYrhbcwCkzTPttVBA5hyykxCAvPioMpgMZlsq8GW8Jyn4VLAi
pyEy5z3x3dRgCpZ9vBMs5IstEE4fxRAznk0hHNRr+MQPSFhGhgsRoAbwr+rdw0A2TRfQV13dfWKS1NZIMfkrTxfof1hCDAHQw/ed8G9IeN+TTruRggrGnK0WlVdY67o46uiKMN7ugpB0+OzUi+QEBMNEdS4hs/xpG5BHDYq/06Pqzo!:18317:0:99999::
```

@jaylagorio // jay@lagor.io
void initACMUtiMI(fwInfo_s *fwInfo)
{
    char *ftpPath;

    fwInfo->fwUpgradeMethod = 1;
    fwInfo->ftpPort = 0x15;
    fwInfo->ftpID[0] = 'z';
    fwInfo->ftpID[1] = 'y';
    fwInfo->ftpID[3] = 'w';
    fwInfo->ftpID[5] = '\0';
    fwInfo->ftpID[5] = '\0';

    *(undefined0 *)fwInfo->ftpPwd = 0x50724f77216145f;
    fwInfo->ftpPwd[8] = 'f';
    fwInfo->ftpPwd[9] = 'X';
    fwInfo->ftpPwd[10] = 'p';
    fwInfo->ftpPwd[0xb] = '\0';
    fwInfo->ftpPwd[0x7f] = '\0';

    snprintf(fwInfo->ftpPath,0x40,\"UNK_10098018,UNK_10097ff8\\");
    fwInfo->ftpPath[0x3f] = '\0';
    fwInfo->failAction = '\0';
    return;
}
FUTURE WORK

Stuff I’m never going to have time for
THE FUTURE

- Make a Python script
  - The deobfuscation function is at 0x10009F00 and is called from what calls 7za at 0x10002DE0
THE FUTURE

- It turns out all this was way easier than I made it.
- Firmware updates include the firmware, some release notes, a .ri file for recovery.
- It’s unencrypted, you can binwalk it directly, and it contains zld_fsextract.
- Thanks to HN Security for pointing out what should have been obvious.
## The Future

<table>
<thead>
<tr>
<th>DECIMAL</th>
<th>HEXADECIMAL</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0x0</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;.&quot;, file name length: &quot;0x00000002&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>112</td>
<td>0x70</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit&quot;, file name length: &quot;0x00000007&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>232</td>
<td>0x148</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/zyinit&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>266064</td>
<td>0x40F50</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/etc_inittab&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>269196</td>
<td>0x41B8C</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/e2fsck&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>709204</td>
<td>0x1AD254</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/zld_mrd.ko&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>715356</td>
<td>0xAAE5C</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/rw.zip&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>2879904</td>
<td>0x2BF1A0</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/mke2fs&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>3151288</td>
<td>0x3015B8</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/switchdev.ko&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>3196236</td>
<td>0x30C54C</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/zld_fsextract&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>3290728</td>
<td>0x323668</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/cn60xx.ko&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>3312388</td>
<td>0x328B04</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/include&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>3391488</td>
<td>0x33C000</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/unzip&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>3396772</td>
<td>0x33D4A4</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/gpl&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>3745356</td>
<td>0x39264C</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/fwversion&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>3745816</td>
<td>0x392B18</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/switchdev_char.ko&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>3768480</td>
<td>0x3980A0</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/unzip&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>3946956</td>
<td>0x3C39CC</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/1km.list&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>3947164</td>
<td>0x3C3A9C</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;zyinit/platform_support.ko&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>3966960</td>
<td>0x3C87F0</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;init&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
<tr>
<td>3967092</td>
<td>0x3C8874</td>
<td>ASCII cpio archive (SVR4 with no CRC), file name: &quot;TRAILER!!!&quot;, file name length: &quot;0x00000000&quot;, file size: &quot;0x00000000&quot;</td>
</tr>
</tbody>
</table>
○ Make a Python script
  □ The deobfuscation function is at 0x10009F00 and is called from what calls 7za at 0x10002DE0
○ Run the decrypted firmware through EMBA
  □ compress.img is your friend
THE FUTURE
## THE FUTURE

### STRC Py - top 10 results:

<table>
<thead>
<tr>
<th>Rank</th>
<th>File Name</th>
<th>Linux File</th>
<th>RELRO</th>
<th>Canary</th>
<th>NX</th>
<th>Symbols</th>
<th>Networking</th>
</tr>
</thead>
<tbody>
<tr>
<td>93</td>
<td>libreadline.so</td>
<td>yes</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>73</td>
<td>libnetsnmpmibs</td>
<td>no</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>51</td>
<td>libkrb5.so.3</td>
<td>yes</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>41</td>
<td>pam_uam.so</td>
<td>no</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>38</td>
<td>pro</td>
<td>yes</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>30</td>
<td>libuser_profile</td>
<td>no</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>30</td>
<td>libnetsnmp.so.1</td>
<td>no</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>29</td>
<td>zyshd</td>
<td>no</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>29</td>
<td>wtp_zyshd</td>
<td>no</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>29</td>
<td>uam-query</td>
<td>no</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>Symbols</td>
<td>No</td>
</tr>
</tbody>
</table>

### SYSTEM - top 10 results:

<table>
<thead>
<tr>
<th>Rank</th>
<th>File Name</th>
<th>Linux File</th>
<th>RELRO</th>
<th>Canary</th>
<th>NX</th>
<th>Symbols</th>
<th>Networking</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>libmyzyxel_serv</td>
<td>no</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>12</td>
<td>libnetsnmpmibs</td>
<td>no</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>9</td>
<td>lib2FA.so</td>
<td>no</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>5</td>
<td>winbindd</td>
<td>no</td>
<td>RELRO</td>
<td>No</td>
<td>No</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>5</td>
<td>rlm_zy_rid.so</td>
<td>no</td>
<td>No</td>
<td>No</td>
<td>NX</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>5</td>
<td>net</td>
<td>yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>rlm_check_net.s</td>
<td>no</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>ntlm_auth</td>
<td>yes</td>
<td>RELRO</td>
<td>No</td>
<td>No</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>nmbd</td>
<td>yes</td>
<td>RELRO</td>
<td>No</td>
<td>No</td>
<td>Symbols</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>libmtd.so</td>
<td>no</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Symbols</td>
<td>No</td>
</tr>
</tbody>
</table>

@jaylagorio // jay@lagor.io
### THE FUTURE

Collect CVE and exploit details.

<table>
<thead>
<tr>
<th>kernel</th>
<th>version</th>
<th>CVE-2022-2380</th>
<th>STAT</th>
<th>No exploit available</th>
</tr>
</thead>
<tbody>
<tr>
<td>kernel</td>
<td>3.10.87</td>
<td>CVE-2021-4135</td>
<td>STAT</td>
<td>No exploit available</td>
</tr>
<tr>
<td>kernel</td>
<td>3.10.87</td>
<td>CVE-2019-15504</td>
<td>STAT</td>
<td>No exploit available</td>
</tr>
<tr>
<td>kernel</td>
<td>3.10.87</td>
<td>CVE-2019-15505</td>
<td>STAT</td>
<td>No exploit available</td>
</tr>
<tr>
<td>kernel</td>
<td>3.10.87</td>
<td>CVE-2019-15502</td>
<td>STAT</td>
<td>No exploit available</td>
</tr>
<tr>
<td>kernel</td>
<td>3.10.87</td>
<td>CVE-2019-14001</td>
<td>STAT</td>
<td>No exploit available</td>
</tr>
<tr>
<td>kernel</td>
<td>3.10.87</td>
<td>CVE-2018-5703</td>
<td>STAT</td>
<td>No exploit available</td>
</tr>
<tr>
<td>kernel</td>
<td>3.10.87</td>
<td>CVE-2018-20961</td>
<td>STAT</td>
<td>Exploit (Github: thdusdl1219_CVE-Study)</td>
</tr>
<tr>
<td>kernel</td>
<td>3.10.87</td>
<td>CVE-2017-7895</td>
<td>STAT</td>
<td>No exploit available</td>
</tr>
<tr>
<td>kernel</td>
<td>3.10.87</td>
<td>CVE-2017-13715</td>
<td>STAT</td>
<td>Exploit (Github: thdusdl1219_CVE-Study)</td>
</tr>
<tr>
<td>kernel</td>
<td>3.10.87</td>
<td>CVE-2016-9555</td>
<td>STAT</td>
<td>Exploit (Github: R081M4N1N_linux-kernel-exploitation)</td>
</tr>
<tr>
<td>kernel</td>
<td>3.10.87</td>
<td>CVE-2016-7117</td>
<td>STAT</td>
<td>Exploit (Github: xairy_linux-kernel-exploitation)</td>
</tr>
</tbody>
</table>
THE FUTURE

Experimental WSL support #274

BenediktMKuehne merged 21 commits into e-m-b-a:master from m-1-k-3:master 11 days ago

Shadow file detection, mipsn32 in user mode emulation, fixes...

m-1-k-3 wants to merge 7 commits into e-m-b-a:master from m-1-k-3:master

Conversation 0 Commits 7 Checks 1 Files changed 9

m-1-k-3 commented 7 hours ago • edited

• What kind of change does this PR introduce? (Bug fix, feature, docs update, ...)

feature (better shadow file detection)
feature (mipsn32 in user mode emulator)
fixes in s13, s14 and s115

Thanks to your support jaylagorio

@jaylagorio // jay@lagor.io
THE FUTURE

○ Make a Python script
  □ The deobfuscation function is at 0x10009F00 and is called from what calls 7za at 0x10002DE0

○ Run the decrypted firmware through EMBA
  □ compress.img is your friend

○ Track whether and when Zyxel makes a breaking change to this process
THE FUTURE

Plain-Text  Transition  Encrypted
Like, to summarize: If your firmware images are encrypted, it generally just means you don’t have faith in your security.

8:41 AM · 4/23/22 · Twitter Web App

3 Retweets 3 Quote Tweets 35 Likes

stacksmashing 🐍 @ghidran... · 4/23/22

Replying to @daveaitel

It’s also in most cases just a nuisance, not really something that stops an attacker 😞

(Though having a slide where you can make fun of bad fw encryption is always good for a laugh)
Happy Bug Hunting!
AGENDA

○ Introduction
○ How it Started
○ The Firmware
○ The Hardware
○ The Software
○ How it’s Going
○ Future Work
○ Questions
ACKNOWLEDGEMENTS

- My friends and family
- DEF CON CFP Review Team
- Niels Teusink (@NielsTeusink)
- Michael Messner for EMBA
- Author of Every Other Tool Mentioned
- The Ghidra Project (ghidra-sre.org)
- Everyone who previewed my talk

@jaylagorio // jay@lagor.io
THANKS!

Questions?

Twitter: @jaylagorio
Github: @jaylagorio
Email: jay@lagor.io