How Sercomm saved my Easter!

Another backdoor in my router: when Christmas is NOT enough!
I don't know about you, but I love Easter!

And with Sercomm, it's Easter every day!
Remember the TCP/32764 router backdoor?

- Introduced by Sercomm
- Gives root shell, no authentication
- Dump entire configuration
- 4 affected manufacturers (Cisco, Linksys, NetGear, Diamond)
- 24 router models confirmed vulnerable
- 6000 vulnerable routers on the Internet

(more info: https://github.com/elvanderb/TCP-32764)
It was patched!

zmaile commented 11 days ago

I brought this issue up with netgear support (2014/01/17), and just in the last few days they have released a new firmware version that resolves the port 32764 issue. The new firmware is available on their website (http://downloadcenter.netgear.com/other/)

I've confirmed that the below version works correctly.
http://www.downloads.netgear.com/files/GDC/DGN1000/DGN1000-V1.1.00.49WW.zip

If the original backdoor was a planned 'feature', then its possible that there is a knocking sequence required to unlock port 32764 (that is, port 32764 opens after trying port 5000, then 8000 before 32764 as an example).
No, it can't be a *feature*!
It was a simple mistake... wasn't it?

I brought this issue up with netgear support (2014/01/17), and just in the last few days they have released a new firmware version that resolves the port 32764 issue. The new firmware is available on their website (http://downloadcenter.netgear.com/other/)

I've confirmed that the below version works correctly.
http://www.downloads.netgear.com/files/GDC/DGN1000/DGN1000-V1.1.00.49WW.zip

If the original backdoor was a planned 'feature', then it's possible that there is a knocking sequence required to unlock port 32764 (that is, port 32764 opens after trying port 5000, then 8000 before 32764 as an example).
Let's have a look!

- 'binwalk -e' to extract the file system
- `scfgmgr` (the backdoor binary) is still present...
- But it's now started with a new -l option
What's this -l option?

- `scfgmgr` now listens on a Unix domain socket :(
Wait... what?

- There is an alternate option: `-f` that makes `scfgmgr` listen on TCP.
Let's see if it's used...

```bash
:~/DGN1000_1.1.00.55 NA.img.extracted$ grep -r "scfgmgr -f" ./squashfs-root/usr/sbin/ft_tool
```
What's this 'ft_tool'?

- Opens a raw socket
- Waits for packets
  - with ethertype = 0x8888
  - coming from the Ethernet card or broadcasted (check of the destination MAC address)
- Packet format

```
00000000 packet_struct struc  # (sizeof=0x228)
00000000 header:  ether_header  ?
00000008 type:   .half  ?
00000010 sequence: .half  ?
00000012 offset: .half  ?
00000014 chunk:  .half  ?
00000016 payload_len: .half  ?
00000018 payload: .byte 528 dup(?)
00000228 packet_struct ends
```
If payload == md5("DGN1000")...
And if packet type == 0x201...

```assembly
And if packet type == 0x201...
```

```assembly
system("scfgmgr -f &")!!!
```
So you can reactivate the backdoor again...

- If you're on the LAN
- Or if you're an Internet provider (if you're one-hop away, you can craft Ethernet headers)

- It's **DELIBERATE**

- You can also use the 0x200 packet type to ping the router (it will respond with its MAC address) and 0x202 to change its LAN IP address
I don't always patch backdoors...

I DON'T ALWAYS PATCH BACKDOORS

BUT WHEN I DO, I ADD ANOTHER ONE
Because a root shell is not enough...

- You can now (among other things) make the router LEDs flash with the 33, 34 and 35th message :)

```asm
jalr $t9 ; set_led_on
addiu $a0, (aPower_red - 0x800000) # 'power_red'
lw $sp, 0x10670+var_16670($sp)
jalr $t9 ; set_led_on
addiu $a0, (aPower_green - 0x800000) # 'power_green'
jalr $t9 ; set_led_off
addiu $a0, (aInternet_green - 0x800000) # 'internet_green'
jalr $t9 ; set_led_on
addiu $a0, (aInternet_red - 0x800000) # 'internet_red'
jal $t9 ; set_led_off
addiu $a0, (aInternet_green - 0x800000) # 'internet_green'
```
But where does it come from?

- The 0x8888 ethertype and packet structure is used in an old Sercomm update tool:
  http://wiki.openwrt.org/_media/toh/netgear/dg834.g.v4/nftp.c
  - lazy guys, they didn't even code their new backdoor from scratch ;)

- It may be present in other hardware but hard to tell:
  - No easy way to scan
  - MD5 signature will certainly be different as it's based on the router commercial name
How to detect it?

For DGN1000, simply use the PoC from your LAN

For other routers, the simplest way is to:
- Use 'binwalk -e' to extract the file system
- Search for 'ft_tool' or grep -r 'scfgmgr -f'
- Use IDA to confirm
We hope you enjoyed this presentation :)  

PoC is available here:  
http://synacktiv.com/ressources/ethercomm.c