

BÁO CÁO THỰC HÀNH

Môn học: Pháp chứng kỹ thuật số

Kỳ báo cáo: Buổi 06 (Session 06)

Tên chủ đề: CTF Final Test

GVHD: Đoàn Minh Trung

Ngày báo cáo: 29/05/2024

Nhóm: 07

1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT334.021.ATCL

STT	Họ và tên	MSSV	Email
1	Nguyễn Tấn Phát	21522447	21522447@gm.uit.edu.vn
2	Nguyễn Đình Bảo Long	21522303	21522303@gm.uit.edu.vn
3	Đào Vĩnh Thịnh	21522632	21522632@gm.uit.edu.vn
4	Ngô Minh Thiên	21522623	21522623@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:¹

STT	Công việc	Thực hiện	Kết quả tự đánh giá
1	Memory	Memory.dmp:0% Dump.raw:30%	15%
2	Network	Tìm IP của web server: Hoàn thành Tìm username và password của một tài khoản sử dụng server: Hoàn thành Hacker tấn công từ đâu: Hoàn thành Tìm lỗ hổng hacker khai thác: Hoàn thành Tìm tài khoản hacker đã login: Hoàn thành Tìm Server hacker dùng để test: Hoàn thành Tìm mật khẩu của admin: Hoàn thành Có nên tình nghi đặc vụ đó là người đã thực hiện cuộc tấn công không? Tại sao? Hoàn thành	100%
3	Android	Bypass_login	100%
4	Steganography	DecaovsDatg	0%

¹ Ghi nội dung công việc, các kịch bản trong bài Thực hành.

BÁO CÁO CHI TIẾT

1. Memory

Câu Memory.dmp

```
(lixsong@kali)-[~/Downloads/volatility]
$ strings memory.dmp | grep -i "distrib_description="
DISTRIB_DESCRIPTION="Ubuntu 20.04.2 LTS"
DISTRIB_DESCRIPTION="Ubuntu 20.04.2 LTS"
DISTRIB_DESCRIPTION="Ubuntu 20.04.2 LTS"
DISTRIB_DESCRIPTION=
DISTRIB_DESCRIPTION=%s
DISTRIB_DESCRIPTION=
DISTRIB_DESCRIPTION=%s
DISTRIB_DESCRIPTION=
DISTRIB_DESCRIPTION=%s
```

Kiểm tra linux version.

```
(lixsong@kali)-[~/Downloads/volatility]
$ strings memory.dmp | grep -i "linux version"
o The intent is to make the tool independent of Linux version dependencies,
o The intent is to make the tool independent of Linux version dependencies,
MESSAGE=Linux version 5.13.0-39-generic (buildd@lcy02-amd64-080) (gcc (Ubuntu 9.4.0-1ubuntu1~20.04.1) 9.4.0, GNU ld (GNU Binutils for Ubuntu) 2.34) #44~20.04.1-Ubuntu SMP Thu Mar 24 16:43:35 UTC 2022 (Ubuntu 5.13.0-39.44~20.04.1-generic 5.13.19)
Linux version 5.13.0-39-generic (buildd@lcy02-amd64-080) (gcc (Ubuntu 9.4.0-1ubuntu1~20.04.1) 9.4.0, GNU ld (GNU Binutils for Ubuntu) 2.34) #44~20.04.1-Ubuntu SMP Thu Mar 24 16:43:35 UTC 20
22 (Ubuntu 5.13.0-39.44~20.04.1-generic 5.13.19)
Linux version 5.13.0-39-generic (buildd@lcy02-amd64-080) (gcc (Ubuntu 9.4.0-1ubuntu1~20.04.1) 9.4.0, GNU ld (GNU Binutils for Ubuntu) 2.34) #44~20.04.1-Ubuntu SMP Thu Mar 24 16:43:35 UTC 20
22 (Ubuntu 5.13.0-39.44~20.04.1-generic 5.13.19)
o The intent is to make the tool independent of Linux version dependencies,
Linux version 5.13.0-39-generic (buildd@lcy02-amd64-080) (gcc (Ubuntu 9.4.0-1ubuntu1~20.04.1) 9.4.0, GNU ld (GNU Binutils for Ubuntu) 2.34) #44~20.04.1-Ubuntu SMP Thu Mar 24 16:43:35 UTC 20
22 (Ubuntu 5.13.0-39.44~20.04.1-generic 5.13.19)
Linux version 5.13.0-39-generic (buildd@lcy02-amd64-080) (gcc (Ubuntu 9.4.0-1ubuntu1~20.04.1) 9.4.0, GNU ld (GNU Binutils for Ubuntu) 2.34) #44~20.04.1-Ubuntu SMP Thu Mar 24 16:43:35 UTC 20
22 (Ubuntu 5.13.0-39.44~20.04.1-generic 5.13.19)
Linux version 5.13.0-39-generic (buildd@lcy02-amd64-080) (gcc (Ubuntu 9.4.0-1ubuntu1~20.04.1) 9.4.0, GNU ld (GNU Binutils for Ubuntu) 2.34) #44~20.04.1-Ubuntu SMP Thu Mar 24 16:43:35 UTC 20
22 (Ubuntu 5.13.0-39.44~20.04.1-generic 5.13.19)
Linux version 5.13.0-39-generic (buildd@lcy02-amd64-080) (gcc (Ubuntu 9.4.0-1ubuntu1~20.04.1) 9.4.0, GNU ld (GNU Binutils for Ubuntu) 2.34) #44~20.04.1-Ubuntu SMP Thu Mar 24 16:43:35 UTC 20
22 (Ubuntu 5.13.0-39.44~20.04.1-generic 5.13.19)
Linux version 5.13.0-39-generic (buildd@lcy02-amd64-080) (gcc (Ubuntu 9.4.0-1ubuntu1~20.04.1) 9.4.0, GNU ld (GNU Binutils for Ubuntu) 2.34) #44~20.04.1-Ubuntu SMP Thu Mar 24 16:43:35 UTC 20
22 (Ubuntu 5.13.0-39.44~20.04.1-generic 5.13.19)
```

Câu Dump.raw

Dùng Volatility check profile của dump.raw

```
(lixsong@kali)-[~/Downloads/volatility]
$ python2 vol.py -f dump.raw imageinfo
Volatility Foundation Volatility Framework 2.6.1
** Failed to import volatility.plugins.malware.apihooks (NameError: name 'distorm3' is not defined)
** Failed to import volatility.plugins.malware.threads (NameError: name 'distorm3' is not defined)
** Failed to import volatility.plugins.mac.apihooks_kernel (ImportError: No module named distorm3)
** Failed to import volatility.plugins.mac.check_syscall_shadow (ImportError: No module named distorm3)
** Failed to import volatility.plugins.ssdt (NameError: name 'distorm3' is not defined)
** Failed to import volatility.plugins.mac.apihooks (ImportError: No module named distorm3)
INFO : volatility.debug : Determining profile based on KDBG search...
Suggested Profile(s): Win7SP1x64, Win7SP0x64, Win2008R2SP1x64_24000, Win2008R2SP1x64_23418, Win2008R2SP1x64_24000, Win7SP1x64_24000, Win7SP1x64_23418
AS Layer1 : WindowsAMD64PagedMemory (Kernel AS)
AS Layer2 : FileAddressSpace (/home/lixsong/Downloads/volatility/dump.raw)
PAE type : No PAE
DTB : 0x187000L
KDBG : 0xf800029f2110L
Number of Processors : 1
Image Type (Service Pack) : 1
KPCR for CPU 0 : 0xfffff800029f3d00L
KUSER_SHARED_DATA : 0xfffff78000000000L
Image date and time : 2022-04-08 19:05:12 UTC+0000
Image local date and time : 2022-04-08 12:05:12 -0700
```

Thử nghiệm lấy thông tin mật khẩu.

Ở đây ta dùng hivelist để lấy ra trường địa chỉ bắt đầu trong bộ nhớ của nơi lưu trữ thông tin đăng ký và quản lý về tài khoản người dùng Windows.

```
(lixsong@kali)-[~/Downloads/volatility]
$ python2 vol.py -f dump.raw --profile=Win7SP1x64 hivelist mel image generic
Volatility Foundation Volatility Framework 2.6.1
*** Failed to import volatility.plugins.malware.apihooks (NameError: name 'distorm3' is not defined)
*** Failed to import volatility.plugins.malware.threads (NameError: name 'distorm3' is not defined)
*** Failed to import volatility.plugins.mac.apihooks_kernel (ImportError: No module named distorm3)
*** Failed to import volatility.plugins.mac.check_syscall_shadow (ImportError: No module named distorm3)
*** Failed to import volatility.plugins.ssd (NameError: name 'distorm3' is not defined)
*** Failed to import volatility.plugins.mac.apihooks (ImportError: No module named distorm3)
Virtual          Physical          Name
-----
0xfffff8a0012a6010 0x000000009e18b010 \??\C:\Users\sshd_server\ntuser.dat
0xfffff8a0012bb270 0x000000004829e270 \??\C:\Users\sshd_server\AppData\Local\Microsoft\Windows\UsrClass.dat
0xfffff8a0017f4010 0x0000000019cda010 \??\C:\Users\TEMP\ntuser.dat
0xfffff8a001882410 0x0000000021a41410 \??\C:\Users\TEMP\AppData\Local\Microsoft\Windows\UsrClass.dat
0xfffff8a0032eb010 0x0000000011ff7a010 \??\C:\Windows\AppCompat\Programs\Amcache.hve
0xfffff8a00484c010 0x00000000a8ca5010 \Device\HarddiskVolume1\Boot\BCD
0xfffff8a004ecd010 0x00000000529bb010 \SystemRoot\System32\Config\DEFAULT
0xfffff8a004ed7010 0x0000000052913010 \SystemRoot\System32\Config\SAM
0xfffff8a00000e010 0x00000000a9537010 [no name]
0xfffff8a000024010 0x00000000a9742010 \REGISTRY\MACHINE\SYSTEM
0xfffff8a000063010 0x00000000a9683010 \REGISTRY\MACHINE\HARDWARE
0xfffff8a0005dc010 0x0000000054799010 \SystemRoot\System32\Config\SECURITY
0xfffff8a0005e6010 0x0000000013a00010 \SystemRoot\System32\Config\SOFTWARE
0xfffff8a000e2b010 0x00000000a4cc8010 \??\C:\System Volume Information\Syscache.hve
0xfffff8a000e61010 0x00000000dc000010 \??\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
0xfffff8a000ef1010 0x000000004b8d9010 \??\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
```

Sau đó ta trích xuất mã băm mật khẩu vào một tập tin text để tiện quan sát

```
(lixsong@kali)-[~/Downloads/volatility]
$ python2 vol.py -f dump.raw --profile=Win7SP1x64 hashdump -y 0xfffff8a000024010 -s 0xfffff8a004ed7010 > pwdhashes.txt
Volatility Foundation Volatility Framework 2.6.1
```

Mật khẩu đã được hash và thông tin chỉ được lưu lại ở dạng hash.

```
(lixsong@kali)-[~/Downloads/volatility]
$ cat pwdhashes.txt
*** Failed to import volatility.plugins.malware.apihooks (NameError: name 'distorm3' is not defined)
*** Failed to import volatility.plugins.malware.threads (NameError: name 'distorm3' is not defined)
*** Failed to import volatility.plugins.mac.apihooks_kernel (ImportError: No module named distorm3)
*** Failed to import volatility.plugins.mac.check_syscall_shadow (ImportError: No module named distorm3)
*** Failed to import volatility.plugins.ssd (NameError: name 'distorm3' is not defined)
*** Failed to import volatility.plugins.mac.apihooks (ImportError: No module named distorm3)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:fc525c9683e8fe067095ba2ddc971889:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
NHK-InsecLab:1000:aad3b435b51404eeaad3b435b51404ee:141be588e38b145c4e1f274b646898eb:::
sshd:1001:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
sshd_server:1002:aad3b435b51404eeaad3b435b51404ee:8d0a16cfc061c3359db455d00ec27035:::
```

Đã thử dùng hashcat và tool hashes online nhưng ko ra kết quả pass ở dạng plaintext.

2. Network

a. Tìm IP của web server.

Sử dụng Statistics cho Ipv4 thì ta thấy tổng số packet của IP 224.0.0.251 và 172.18.0.3 bằng 122213, tổng số packet của 3 IP còn lại cũng bằng 122213

Wireshark · All Addresses · pcap.pcap									
Topic / Item	Count	Average	Min Val	Max Val	Rate (ms)	Percent	Burst Rate	Burst Start	
All Addresses	122213				0.3875	100%	3.4600	306.092	
224.0.0.251	4				0.0000	0.00%	0.0100	0.000	
172.18.0.3	122209				0.3874	100.00%	3.4600	306.092	
172.18.0.2	14027				0.0445	11.48%	2.0800	306.082	
172.18.0.1	8671				0.0275	7.09%	1.3000	306.092	
151.101.78.132	99515				0.3155	81.43%	2.2100	150.720	

Xem qua vài packet thì thấy đa số là IP 172.18.0.3 response, 172.18.0.2 thì response các gói tin qua giao thức MySQL nên có thể đây là database của server, 172.18.0.1 có thể là default gateway. Vậy mô hình mạng theo em suy đoán thì hacker sử dụng máy có IP 151.101.78.132. IP của server là 172.18.0.3

19964 85.295431	172.18.0.3	172.18.0.2	TCP	74	33992 → 3306 [SYN] Seq=0 Win=32120 Len=0 MSS=1460 SACK_PERM TSval=3228849465 TSecr=0 WS=128
19965 85.295462	172.18.0.2	172.18.0.3	TCP	74	3306 → 33992 [SYN, ACK] Seq=0 Ack=1 Win=31856 Len=0 MSS=1460 SACK_PERM TSval=3228849465 TSecr=3228849465 WS=128
19966 85.295469	172.18.0.3	172.18.0.2	TCP	66	33992 → 3306 [ACK] Seq=1 Ack=1 Win=32128 Len=0 TSval=3228849465 TSecr=3228849465 WS=128
19967 85.295690	172.18.0.2	172.18.0.3	MySQL	143	Server Greeting proto=10 version=8.3.0
19968 85.295698	172.18.0.3	172.18.0.2	TCP	66	33992 → 3306 [ACK] Seq=1 Ack=78 Win=32128 Len=0 TSval=3228849465 TSecr=3228849465 WS=128
19969 85.295715	172.18.0.3	172.18.0.2	MySQL	206	Login Request user=root db=chall
19970 85.295723	172.18.0.2	172.18.0.3	TCP	66	3306 → 33992 [ACK] Seq=78 Ack=141 Win=31872 Len=0 TSval=3228849465 TSecr=3228849465 WS=128
19971 85.295749	172.18.0.2	172.18.0.3	MySQL	72	Caching_sha2_password fast_auth_success
19972 85.295797	172.18.0.2	172.18.0.3	MySQL	77	Response OK
19973 85.295801	172.18.0.3	172.18.0.2	TCP	66	33992 → 3306 [ACK] Seq=141 Ack=95 Win=32128 Len=0 TSval=3228849465 TSecr=3228849465 WS=128
19974 85.295873	172.18.0.3	172.18.0.2	MySQL	126	Request Query
19975 85.296151	172.18.0.2	172.18.0.3	MySQL	248	Response TABULAR Response
19976 85.296228	172.18.0.3	172.18.0.2	MySQL	71	Request Quit
19977 85.296238	172.18.0.3	172.18.0.2	TCP	66	33992 → 3306 [FIN, ACK] Seq=206 Ack=277 Win=32000 Len=0 TSval=3228849465 TSecr=3228849465 WS=128
19978 85.296253	172.18.0.2	172.18.0.3	TCP	66	3306 → 33992 [FIN, ACK] Seq=277 Ack=207 Win=31872 Len=0 TSval=3228849465 TSecr=3228849465 WS=128
19979 85.296257	172.18.0.3	172.18.0.2	TCP	66	33992 → 3306 [ACK] Seq=207 Ack=278 Win=32000 Len=0 TSval=3228849465 TSecr=3228849465 WS=128

- b. Tìm username và password của một tài khoản sử dụng server. NHK nghe nói anh ta là một đặc vụ mật.

Sử dụng Ctrl+F tìm string "login", gói tin 3786 có hiện thông tin username là 'agentp' và password là "perrytheplatypu"

3786 20.317082 172.18.0.1 172.18.0.3 HTTP 614 POST /login.php HTTP/1.1 (application/x-www-form-urlencoded)

Source Address: 172.18.0.1
Destination Address: 172.18.0.3
Transmission Control Protocol, Src Port: 34068, Dst Port: 80, Seq: 1, Ack: 1, Len: 548
Source Port: 34068
Destination Port: 80
[Stream index: 1]
[Conversation completeness: Complete, WITH_DATA (31)]
[TCP Segment Len: 548]
Sequence Number: 1 (relative sequence number)
Sequence Number (raw): 2107149488
[Next Sequence Number: 549 (relative sequence number)]
Acknowledgment Number: 1 (relative ack number)
Acknowledgment number (raw): 2896418520
1000 = Header Length: 32 bytes (8)
> Flags: 0x018 (PSH, ACK)
Window: 251
[Calculated window size: 32128]
[Window size scaling factor: 128]
Checksum: 0x5a73 [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
> Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
> [Timestamps]
> [SEQ/ACK analysis]
TCP payload (548 bytes)
Hypertext Transfer Protocol
HTML Form URL Encoded: application/x-www-form-urlencoded
> Form item: "username" = "agentp"
> Form item: "password" = "perrytheplatypu"

```

0000 02 42 ac 12 00 03 02 42 eb 1f c6 4a 08 00 45 00 .B....B...J...E...
0010 02 58 69 f9 40 00 40 06 76 7e ac 12 00 01 ac 12 .Xi...@...v.....
0020 00 03 85 14 00 50 7d 98 8c b0 ac a3 d6 d8 80 18 .....P)...
0030 00 fb 5a 73 00 00 01 01 08 0a 5a 91 91 b8 12 5e ..Zs....Z....^
0040 69 98 50 4f 53 54 20 2f 6c 6f 67 69 6e 2e 70 68 i POST /login.ph
0050 70 20 48 54 54 50 2f 31 2e 31 0d 0a 48 6f 73 74 p HTTP/1.1..Host
0060 3a 20 61 73 38 37 34 32 2e 64 75 63 6b 64 6e 73 : as8742 .duckdns
0070 2e 6f 72 67 3a 32 38 30 38 0d 0a 55 73 65 72 2d .org:280 8..User-
0080 41 67 65 6e 74 3a 20 4d 6f 7a 69 6c 6c 61 2f 35 Agent: Mozilla/5
0090 2e 30 20 28 58 31 31 3b 20 4c 69 6e 75 78 20 78 .0 (X11; Linux x
00a0 38 36 5f 36 3a 3b 20 72 76 3a 31 32 31 2e 30 29 86;4; r v:121.0)
00b0 20 47 65 63 6b 6f 2f 32 30 31 30 30 31 30 31 20 Gecko/2.0.0100101
00c0 46 69 72 65 66 6f 78 2f 31 32 31 2e 30 0d 0a 41 Firefox/ 121.0..A
00d0 63 63 65 70 74 3a 20 74 65 78 74 2f 68 74 6d 6c ccept: t ext/html
00e0 2c 61 70 70 6c 69 63 61 74 69 6f 6e 2f 78 68 74 ,applicati on/xht
00f0 6d 6c 2b 78 6d 6c 2c 61 70 70 6c 69 63 61 74 69 ml+xml,a pplicati
0100 6f 6e 2f 78 6d 6c 3b 71 3d 30 2e 39 2c 69 6d 61 on/xml;q =0.9,ima
0110 67 65 2f 61 76 69 66 2c 69 6d 61 67 65 2f 77 65 ge/avif, image/we
0120 62 70 2c 2a 2f 2a 3b 71 3d 30 2e 38 0d 0a 41 63 bp,*/;q =0.8..Ac
0130 63 65 70 74 2d 4c 61 6e 67 75 61 67 65 3a 20 65 cept-Lan guage: e
0140 6e 2d 55 53 2c 65 6e 3b 71 3d 30 2e 35 0d 0a 41 n-US;q=0.5..A
0150 63 63 65 70 74 2d 45 6e 63 6f 64 69 6e 67 3a 20 ccept-En coding:
0160 67 7a 69 70 2c 20 64 65 66 6c 61 74 65 0d 0a 43 gzip, de flate..C
0170 6f 6e 74 65 6e 74 2d 54 79 70 65 3a 20 61 70 70 ontent-1 ype: app
0180 6c 69 63 61 74 69 6f 6e 2f 78 2d 77 77 2d 66 lication /x-www-f
0190 6f 72 6d 2d 75 72 6c 65 6e 63 6f 64 65 64 0d 0a orm-urle ncoded..
01a0 43 6f 6e 74 65 6e 74 2d 4c 65 6e 67 74 68 3a 20 Content- Length:
01b0 3a 30 0d 0a 4f 72 69 6f 69 6e 3a 20 68 74 70 40 ..Orig in: htt
01c0 3a 2f 2f 61 73 38 37 34 32 64 75 63 6b 64 6e 6e ://as874 2.duckdn
01d0 73 2e 6f 72 67 3a 32 38 30 38 0d 0a 43 6f 6e 6e s.org:28 08..Conn
01e0 65 63 74 69 6f 6e 3a 20 6b 65 65 70 2d 61 6c 69 ection: keep-ali
01f0 76 65 0d 0a 52 65 66 65 72 65 72 3a 20 68 74 74 ve..Refere r: htt

```

- c. Hacker tấn công từ bên ngoài mạng hay là từ bên trong mạng.
Theo câu a thì hacker tấn công từ bên ngoài mạng

- d. Lỗ hổng là hacker dùng để khai thác là gì?

Trong file pcap có rất nhiều packet có payload giống như sau, vậy nên attacker khai thác lỗ hổng SQL injection

- e. Hacker đã login vào tài khoản nào?

Ở packet 4929 ta lại thấy POST request gửi tới server với cùng username và password nhưng lần này password có thêm chữ 's' thay vì bị thiếu như packet trước

```

4928 26.017854 172.18.0.1 172.18.0.3 TCP 66 57126 → 80 [ACK] Seq=1 Ack=1 Win=32128 Len=0 TSval=1519495165 TSecr=308:
+ 4929 26.017907 172.18.0.1 172.18.0.3 HTTP 676 POST /login.php HTTP/1.1 (application/x-www-form-urlencoded)
4930 26.017914 172.18.0.3 172.18.0.1 TCP 66 80 → 57126 [ACK] Seq=1 Ack=611 Win=31872 Len=0 TSval=308183005 TSecr=15:

> Frame 4929: 676 bytes on wire (5408 bits), 676 bytes captured (5408 bits)
> Ethernet II, Src: 02:42:eb:1f:c6:4a (02:42:eb:1f:c6:4a), Dst: 02:42:ac:12:00:03 (02:42:ac:12:00:03)
> Internet Protocol Version 4, Src: 172.18.0.1, Dst: 172.18.0.3
> Transmission Control Protocol, Src Port: 57126, Dst Port: 80, Seq: 1, Ack: 1, Len: 610
> Hypertext Transfer Protocol
  > HTML Form URL Encoded: application/x-www-form-urlencoded
    > Form item: "username" = "agentp"
    > Form item: "password" = "perrytheplatypus"
    [Community ID: 1:URNMx:164CVtrNFtN0/G6oieuxo=]

```

Packet 7073 cũng có 1 POST request gửi với username và password là admin

```

7072 33.103027 172.18.0.3 151.101.78.132 TCP 66 36794 → 80 [ACK] Seq=3013 Ack=10323844 Win=793216 Len=0 TSval=5014:
+ 7073 33.105063 172.18.0.1 172.18.0.3 HTTP 655 POST /login.php HTTP/1.1 (application/x-www-form-urlencoded)
7074 33.106430 172.18.0.3 172.18.0.2 TCP 74 58040 → 3306 [SYN] Seq=0 Win=32120 Len=0 MSS=1460 SACK_PERM TSval=
7075 33.106497 172.18.0.2 172.18.0.3 TCP 74 3306 → 58040 [SYN, ACK] Seq=0 Ack=1 Win=31856 Len=0 MSS=1460 SACK_

> Frame 7073: 655 bytes on wire (5240 bits), 655 bytes captured (5240 bits)
> Ethernet II, Src: 02:42:eb:1f:c6:4a (02:42:eb:1f:c6:4a), Dst: 02:42:ac:12:00:03 (02:42:ac:12:00:03)
> Internet Protocol Version 4, Src: 172.18.0.1, Dst: 172.18.0.3
> Transmission Control Protocol, Src Port: 57126, Dst Port: 80, Seq: 1468, Ack: 1200, Len: 589
> Hypertext Transfer Protocol
  > HTML Form URL Encoded: application/x-www-form-urlencoded
    > Form item: "username" = "admin"
    > Form item: "password" = "admin"
    [Community ID: 1:URNMx:164CVtrNFtN0/G6oieuxo=]

```

f. Server mà hacker dùng để test là gì?

Ở đây server hacker dùng để test là MySQL Database server

95615	252.400330	172.18.0.3	172.18.0.1	TCP	66 80 → 3306
95616	252.403070	172.18.0.3	172.18.0.2	TCP	74 45156 → 3306
95617	252.403158	172.18.0.2	172.18.0.3	TCP	74 3306 → 45156
95618	252.403179	172.18.0.3	172.18.0.2	TCP	66 45156 → 3306
95619	252.403705	172.18.0.2	172.18.0.3	MySQL	143 Server G
95620	252.403730	172.18.0.3	172.18.0.2	TCP	66 45156 → 3306
95621	252.403851	172.18.0.3	172.18.0.2	MySQL	206 Login Re
95622	252.403866	172.18.0.2	172.18.0.3	TCP	66 3306 → 45156
95623	252.403962	172.18.0.2	172.18.0.3	MySQL	72 Caching_
95624	252.404087	172.18.0.2	172.18.0.3	MySQL	77 Response
95625	252.404098	172.18.0.3	172.18.0.2	TCP	66 45156 → 3306
95626	252.404348	172.18.0.3	172.18.0.2	MySQL	234 Request
95627	252.405751	172.18.0.2	172.18.0.3	MySQL	257 Response
95628	252.406172	172.18.0.3	172.18.0.2	MySQL	71 Request
95629	252.406199	172.18.0.3	172.18.0.2	TCP	66 45156 → 3306
95630	252.406219	172.18.0.2	172.18.0.3	TCP	66 3306 → 45156
95631	252.406226	172.18.0.3	172.18.0.2	TCP	66 45156 → 3306
95632	252.406605	172.18.0.3	172.18.0.1	HTTP	418 HTTP/1.1

Như câu a thì ta đoán IP của hacker là 151.101.78.132 nên ta có thể thử dùng curl và nmap để thử xác định server của hacker

```
(lixsong@kali) - [~/Downloads/volatility]
$ curl -I 151.101.78.132
HTTP/1.1 500 Domain Not Found
Connection: keep-alive
Content-Length: 249
Server: Varnish
Retry-After: 0
content-type: text/html
Cache-Control: private, no-cache
X-Served-By: cache-hkg17930-HKG
Accept-Ranges: bytes
Date: Wed, 29 May 2024 16:18:23 GMT
Via: 1.1 varnish

(lixsong@kali) - [~/Downloads/volatility]
$ curl -I 172.18.0.1
^C

(lixsong@kali) - [~/Downloads/volatility]
$ nmap -sV 151.101.78.132
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-29 23:19 +07
Nmap scan report for 151.101.78.132
Host is up (0.048s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
80/tcp    open  http         Varnish
443/tcp   open  ssl/https    Varnish
2 services unrecognized despite returning data. If you know the service/version,
=====
NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)
=====
SF-Port80-TCP:V=7.94SVN%I=7%D=5/29%Time=665755F3%P=x86_64-pc-linux-gnu%r(G
SF:etRequest,1F0,"HTTP/1.1\x20500\x20Domain\x20Not\x20Found\r\nConnection
SF::\x20close\r\nContent-Length:\x20221\r\nServer:\x20Varnish\r\nRetry-Aft
SF:er:\x200\r\ncontent-type:\x20text/html\r\nCache-Control:\x20private,\x2
SF:0no-cache\r\nX-Served-By:\x20cache-hkg17926-HKG\r\nAccept-Ranges:\x20by
SF:tes\r\nDate:\x20Wed,\x2029\x20May\x202024\x2016:21:07\x20GMT\r\nVia:\x2
SF:01\x20varnish\r\n\r\n<html>\n<head>\n<title>Fastly\x20error:\x20un
SF:known\x20domain\x20</title>\n</head>\n<body>\n<p>Fastly\x20error:\x20un
SF:known\x20domain:\x20.\x20Please\x20check\x20that\x20this\x20domain\x20
SF:has\x20been\x20added\x20to\x20a\x20service\.\n</p>\n<p>Details:\x20cache-
SF:hkg17926-HKG</p></body></html>")%r(HTTPOptions,1F0,"HTTP/1.1\x20500\x2
```

Có thể thấy service ở đây là Varnish.

Theo tìm hiểu thì Varnish hoạt động như một reverse proxy, đứng giữa người dùng và server backend. Các yêu cầu từ người dùng sẽ được gửi tới Varnish trước, sau đó Varnish sẽ quyết định có nên trả về nội dung từ cache hay chuyển tiếp yêu cầu tới server backend.

Do vậy hacker đã dùng Varnish để che giấu đi server của hacker thật sự nên chưa thể tìm ra.....

- g. Có vẻ là hacker đã lấy được mật khẩu của admin. Nhưng có người lại bảo là chưa. Vậy hacker đã lấy được mật khẩu của admin chưa? Mật khẩu của admin là gì? Hacker đã lấy được gì?

Từ khoảng packet thứ 87000 trở đi em thấy có rất nhiều packet có payload như sau

```
Wireshark - Packet 87086 - pcap.pcap
Frame 87086: 446 bytes on wire (3568 bits), 446 bytes captured (3568 bits)
Ethernet II, Src: 02:42:eb:1f:c6:4a (02:42:eb:1f:c6:4a), Dst: 02:42:ac:12:00:03 (02:42:ac:12:00:03)
Internet Protocol Version 4, Src: 172.18.0.1, Dst: 172.18.0.3
Transmission Control Protocol, Src Port: 41410, Dst Port: 80, Seq: 1, Ack: 1, Len: 380
Hypertext Transfer Protocol
GET /news.php?name=%22%20UNION%20SELECT%20IF%28SUBSTR%28%28SELECT%20password%20FROM%20users%20WHERE%20username%3D%27admin%27%29%2C1%2C1%29%3D%27a%27%2Csleep%284%29%2Csleep%280%29%29%2C%27a%27%27%27 HTTP/1.1\r\n
Host: as8742.duckdns.org:2808\r\n
User-Agent: python-requests/2.31.0\r\n
Accept-Encoding: gzip, deflate\r\n
Accept: */*\r\n
Connection: keep-alive\r\n
Cookie: PHPSESSID=f66853a47ea490bfe69cc82954439b8d\r\n
[Full request URI: http://as8742.duckdns.org:2808/news.php?name=%22%20UNION%20SELECT%20IF%28SUBSTR%28%28SELECT%20password%20FROM%20users%20WHERE%20username%3D%27admin%27%29%2C1%2C1%29%3D%27a%27%2Csleep%284%29%2Csleep%280%29%29%2C%27a%27%27%27 HTTP/1.1]
[HTTP request 1/1]
[Response in frame: 87104]
[Community ID: 1:EP1VvyRSHGJO5glTcP0hs0KtwG6o]
```

Thử dùng chatgpt với prompt “convert this payload to readable string for me” để hiểu payload dùng để làm gì

```
GET /news.php?name=" UNION SELECT IF(  
    SUBSTR(  
        (SELECT password FROM users WHERE username='admin'), 1, 1  
    ) = '@',  
    sleep(4),  
    sleep(0)  
), 'a' --  
HTTP/1.1\r\n
```

1. `GET /news.php?name=`: This is the URL being requested, where ``name`` is a query parameter.
2. `*** UNION SELECT IF(...), 'a' -- **`: This is the SQL injection payload inserted into the ``name`` parameter.
 - `"`: The closing quote of the ``name`` parameter, potentially terminating any existing SQL query.
 - **UNION SELECT**: Combines the result of this SELECT statement with the result of any existing query.
 - **IF(condition, true_value, false_value)**: Conditional function in SQL.
 - **SUBSTR((SELECT password FROM users WHERE username='admin'), 1, 1)**: Retrieves the first character of the password for the user 'admin'.
 - **= '@'**: Checks if this first character is the '@' symbol.
 - **sleep(4)**: Causes a 4-second delay if the condition is true.
 - **sleep(0)**: Causes no delay if the condition is false.
 - `'a'`: The second part of the UNION SELECT statement, returning 'a'.
 - `--`: SQL comment marker to ignore the rest of the SQL query.
3. `HTTP/1.1\r\n`: The HTTP version used in the request.

Chatgpt còn giải thích giúp thêm về payload, thì em hiểu payload sẽ so sánh kí tự đầu tiên của password cho user 'admin' có phải là '@' hay không, nếu đúng thì sẽ sleep 4s. Lúc này em sẽ đi tìm những khoảng thời gian hacker gửi payload tấn công tới server và bị ngắt khoảng 4s không gửi payload tấn công nữa.

09377	230.807814	172.18.0.3	151.101.78.132	TCP	66 36794 → 80 [ACK] Seq=22804 Ack=137733138 Win=1437440 Len=0 TSval=501608133 TSecr=1402774997
09378	230.893329	172.18.0.1	172.18.0.3	TCP	74 42132 → 80 [SYN] Seq=0 Win=32128 Len=0 MSS=1460 SACK_PERM TSval=1519700041 TSecr=0 WS=128
09379	230.893876	172.18.0.3	172.18.0.1	TCP	74 80 → 42132 [SYN, ACK] Seq=0 Ack=1 Win=31856 Len=0 MSS=1460 SACK_PERM TSval=308387881 TSecr=1519700041 WS=128
09380	230.893905	172.18.0.1	172.18.0.3	TCP	66 42132 → 80 [ACK] Seq=1 Ack=1 Win=32128 Len=0 TSval=1519700041 TSecr=308387881
09381	230.893906	172.18.0.1	172.18.0.3	TCP	448 3306 → 43188 [SYN, ACK] Seq=0 Ack=1 Win=31856 Len=0 MSS=1460 SACK_PERM TSval=797481973 TSecr=3228995064 WS=128
09382	230.893970	172.18.0.3	172.18.0.1	TCP	66 80 → 42132 [ACK] Seq=1 Ack=383 Win=31872 Len=0 TSval=308387881 TSecr=1519700041
09383	230.894778	172.18.0.3	172.18.0.2	TCP	74 43188 → 3306 [SYN] Seq=0 Win=32128 Len=0 MSS=1460 SACK_PERM TSval=3228995064 TSecr=0 WS=128
09384	230.894848	172.18.0.2	172.18.0.3	TCP	74 3306 → 43188 [SYN, ACK] Seq=0 Ack=1 Win=31856 Len=0 MSS=1460 SACK_PERM TSval=797481973 TSecr=3228995064 WS=128
09385	230.894860	172.18.0.3	172.18.0.2	TCP	66 43188 → 3306 [ACK] Seq=1 Ack=1 Win=32128 Len=0 TSval=3228995064 TSecr=797481973
09386	230.895177	172.18.0.2	172.18.0.3	MySQL	143 Server Greeting proto=10 version=8.3.0
09387	230.895185	172.18.0.3	172.18.0.2	TCP	66 43188 → 3306 [ACK] Seq=1 Ack=78 Win=32128 Len=0 TSval=3228995064 TSecr=797481973
09388	230.895203	172.18.0.3	172.18.0.2	MySQL	206 Login Request user=root db=chall
09389	230.895211	172.18.0.2	172.18.0.3	TCP	66 3306 → 43188 [ACK] Seq=78 Ack=141 Win=31872 Len=0 TSval=797481973 TSecr=3228995064
09390	230.895245	172.18.0.2	172.18.0.3	MySQL	72 Caching_sha2_password fast_auth_success
09391	230.895388	172.18.0.2	172.18.0.3	MySQL	77 Response OK
09392	230.895512	172.18.0.3	172.18.0.2	TCP	66 43188 → 3306 [ACK] Seq=141 Ack=95 Win=32128 Len=0 TSval=3228995065 TSecr=797481973
09393	230.895584	172.18.0.3	172.18.0.2	MySQL	234 Request Query
09394	230.898646	151.101.78.132	172.18.0.3	HTTP	2946 Continuation
09395	230.898652	172.18.0.3	151.101.78.132	TCP	66 36794 → 80 [ACK] Seq=22804 Ack=137736818 Win=1437440 Len=0 TSval=501608144 TSecr=1402775015
09396	230.909972	151.101.78.132	172.18.0.3	HTTP	2946 Continuation
09397	230.909991	172.18.0.3	151.101.78.132	TCP	66 36794 → 80 [ACK] Seq=22804 Ack=137738898 Win=1437440 Len=0 TSval=501608155 TSecr=1402775022
09398	230.919383	151.101.78.132	172.18.0.3	HTTP	2946 Continuation
09399	230.919341	172.18.0.3	151.101.78.132	TCP	66 36794 → 80 [ACK] Seq=22804 Ack=137741778 Win=1437440 Len=0 TSval=501608165 TSecr=1402775035
09400	230.936475	151.101.78.132	172.18.0.3	HTTP	2946 Continuation
09401	230.936537	172.18.0.3	151.101.78.132	TCP	66 36794 → 80 [ACK] Seq=22804 Ack=137744658 Win=1437440 Len=0 TSval=501608182 TSecr=1402775054
09402	230.942363	172.18.0.2	172.18.0.3	TCP	66 3306 → 43188 [ACK] Seq=95 Ack=309 Win=31872 Len=0 TSval=797482821 TSecr=3228995065
09403	231.304664	151.101.78.132	172.18.0.3	HTTP	2946 Continuation
09404	231.304670	172.18.0.3	151.101.78.132	TCP	66 36794 → 80 [ACK] Seq=22804 Ack=137747538 Win=1437440 Len=0 TSval=501608550 TSecr=1402775122
09405	231.304681	151.101.78.132	172.18.0.3	HTTP	2946 Continuation
09406	231.304689	151.101.78.132	172.18.0.3	HTTP	2946 Continuation
09407	231.304693	151.101.78.132	172.18.0.3	HTTP	2946 Continuation
09408	231.304697	151.101.78.132	172.18.0.3	HTTP	2946 Continuation
09409	231.304701	151.101.78.132	172.18.0.3	HTTP	2946 Continuation
09410	231.304730	172.18.0.3	151.101.78.132	TCP	66 36794 → 80 [ACK] Seq=22804 Ack=137761938 Win=1424384 Len=0 TSval=501608550 TSecr=1402775132
09411	231.334847	151.101.78.132	172.18.0.3	HTTP	1506 Continuation
09412	231.394451	172.18.0.3	151.101.78.132	TCP	66 36794 → 80 [ACK] Seq=22804 Ack=137763378 Win=1448256 Len=0 TSval=501608640 TSecr=1402775415
09413	231.423613	151.101.78.132	172.18.0.3	HTTP	1506 Continuation
09414	231.423646	172.18.0.3	151.101.78.132	TCP	66 36794 → 80 [ACK] Seq=22804 Ack=137764818 Win=1438848 Len=0 TSval=501608669 TSecr=1402775540
09415	231.425784	151.101.78.132	172.18.0.3	HTTP	2946 Continuation
09416	231.425804	151.101.78.132	172.18.0.3	TCP	66 36794 → 80 [ACK] Seq=22804 Ack=137767698 Win=1437440 Len=0 TSval=501608671 TSecr=1402775545
09417	231.443421	151.101.78.132	172.18.0.3	HTTP	2946 Continuation
09418	231.443458	172.18.0.3	151.101.78.132	TCP	66 36794 → 80 [ACK] Seq=22804 Ack=137770578 Win=1437440 Len=0 TSval=501608689 TSecr=1402775569
09419	231.459430	151.101.78.132	172.18.0.3	HTTP	2946 Continuation
09420	231.459457	172.18.0.3	151.101.78.132	TCP	66 36794 → 80 [ACK] Seq=22804 Ack=137773458 Win=1437440 Len=0 TSval=501608705 TSecr=1402775593

Em tìm được rất nhiều packet như trên và sử dụng chatgpt để chuyển đổi payload thành dạng dễ đọc hơn. Sau đó em ghép các kí tự lại với nhau thì được chuỗi “@dm1n_3r37_!@#~” nhưng bị thiếu kí tự thứ 7 và 9.

Lúc này em đi search lại string ‘password’ ở các packet injection đầu tiên xem có bỏ sót gì không thì gặp packet 49482, packet này rất khác các packet trên, em xem thử các packet response thì packet 49495 trả về tabular

Packet list						Narrow & Wide	Case sensitive	String	password
o.	Time	Source	Destination	Protocol	Length	Info			
49476	130.152375	172.18.0.3	151.101.78.132	TCP	66	36794 → 80 [ACK] Seq=15771 Ack=78084285 Win=1106816 Len=0 TSval=501507398 TSecr=1402674319			
49477	130.159104	151.101.78.132	172.18.0.3	HTTP	2946	Continuation			
49478	130.159181	172.18.0.3	151.101.78.132	TCP	66	36794 → 80 [ACK] Seq=15771 Ack=78087165 Win=1106816 Len=0 TSval=501507404 TSecr=1402674326			
49479	130.160284	172.18.0.1	172.18.0.3	TCP	74	50164 → 80 [SYN] Seq=0 Win=32120 Len=0 MSS=1460 SACK_PERM TSval=1519599307 TSecr=0 WS=128			
49480	130.160314	172.18.0.3	172.18.0.1	TCP	74	80 → 50164 [SYN, ACK] Seq=0 Ack=1 Win=31856 Len=0 MSS=1460 SACK_PERM TSval=308287148 TSecr=1519599307 WS=128			
49481	130.160343	172.18.0.1	172.18.0.3	TCP	66	50164 → 80 [ACK] Seq=1 Ack=1 Win=32128 Len=0 TSval=1519599308 TSecr=308287148			
49482	130.160465	172.18.0.1	172.18.0.3	HTTP	538	GET /news.php?name=%22%20union%20select%20username,password%20from%20users+--+ HTTP/1.1			
49483	130.160490	172.18.0.3	172.18.0.1	TCP	66	80 → 50164 [ACK] Seq=1 Ack=473 Win=31872 Len=0 TSval=308287148 TSecr=1519599308			
49484	130.162207	172.18.0.3	172.18.0.2	TCP	74	57410 → 3306 [SYN] Seq=0 Win=32120 Len=0 MSS=1460 SACK_PERM TSval=3228894331 TSecr=0 WS=128			
49485	130.162248	172.18.0.2	172.18.0.3	TCP	74	3306 → 57410 [SYN, ACK] Seq=0 Ack=1 Win=31856 Len=0 MSS=1460 SACK_PERM TSval=797381240 TSecr=3228894331 WS=128			
49486	130.162257	172.18.0.3	172.18.0.2	TCP	66	57410 → 3306 [ACK] Seq=1 Ack=1 Win=32128 Len=0 TSval=3228894331 TSecr=797381240			
49487	130.162775	172.18.0.2	172.18.0.3	MySQL	143	Server Greeting proto=10 version=8.3.0			
49488	130.162795	172.18.0.3	172.18.0.2	TCP	66	57410 → 3306 [ACK] Seq=1 Ack=78 Win=32128 Len=0 TSval=3228894332 TSecr=797381241			
49489	130.162858	172.18.0.3	172.18.0.2	MySQL	206	Login Request user=root db=chall			
49490	130.162867	172.18.0.2	172.18.0.3	TCP	66	3306 → 57410 [ACK] Seq=78 Ack=141 Win=31872 Len=0 TSval=797381241 TSecr=3228894332			
49491	130.162941	172.18.0.2	172.18.0.3	MySQL	72	Caching_sha2_password fast_auth_success			
49492	130.163129	172.18.0.2	172.18.0.3	MySQL	77	Response OK			
49493	130.163137	172.18.0.3	172.18.0.2	TCP	66	57410 → 3306 [ACK] Seq=141 Ack=95 Win=32128 Len=0 TSval=3228894332 TSecr=797381241			
49494	130.163525	172.18.0.3	172.18.0.2	MySQL	169	Request Query			
49495	130.164202	172.18.0.2	172.18.0.3	MySQL	347	Response TABULAR Response			
49496	130.164409	172.18.0.3	172.18.0.2	MySQL	71	Request Quit			
49497	130.164422	172.18.0.3	172.18.0.2	TCP	66	57410 → 3306 [FIN, ACK] Seq=249 Ack=376 Win=31872 Len=0 TSval=3228894334 TSecr=797381242			
49498	130.164437	151.101.78.132	172.18.0.3	HTTP	2946	Continuation			
49499	130.164446	172.18.0.2	172.18.0.3	TCP	66	3306 → 57410 [FIN, ACK] Seq=376 Ack=250 Win=31872 Len=0 TSval=797381243 TSecr=3228894334			

Mở ra xem thử thì có toàn bộ username và password luôn

Packet list Narrow & Wide Case sensitive String password

No.	Time	Source	Destination	Protocol	Length	Info
49495	130.164202	172.18.0.2	172.18.0.3	MySQL	347	Response TABULAR Respo

```

Length: 16000
Type: FIELD_TYPE_BLOB (252)
  Flags: 0x0010
  Decimals: 0
MySQL Protocol - intermediate EOF
  Packet Length: 5
  Packet Number: 4
  Response Code: EOF Packet (0xfe)
  EOF marker: 254
  Warnings: 0
  Server Status: 0x0022
MySQL Protocol - row packet
  Packet Length: 23
  Packet Number: 5
  text
    text: Science news
  text
    text: 1 + 1 = 3
MySQL Protocol - row packet
  Packet Length: 55
  Packet Number: 6
  text
    text: More science news
  text
    text: 88 + 22 is not equal to 100
MySQL Protocol - row packet
  Packet Length: 23
  Packet Number: 7
  text
    text: admin
  text
    text: @dm1n_s3cr37_!@#
MySQL Protocol - row packet
  Packet Length: 24
  Packet Number: 8
  text
    text: agentp
  text
    text: perrytheplatypus
MySQL Protocol - row packet
  Packet Length: 22
  Packet Number: 9
  text
    text: doofenshmirtz
  text
    text: noooooo
MySQL Protocol - row packet
  Packet Length: 13
  Packet Number: 10
  text
    text: guest
  text
    text: 123456
MySQL Protocol - response EOF
  Packet Length: 5
  Packet Number: 11
  Response Code: EOF Packet (0xfe)
  EOF marker: 254
  Warnings: 0
  Server Status: 0x0022
Ready to load or capture

```

Vậy hacker đã lấy được mật khẩu admin là “@dm1n_s3cr37_!@#”

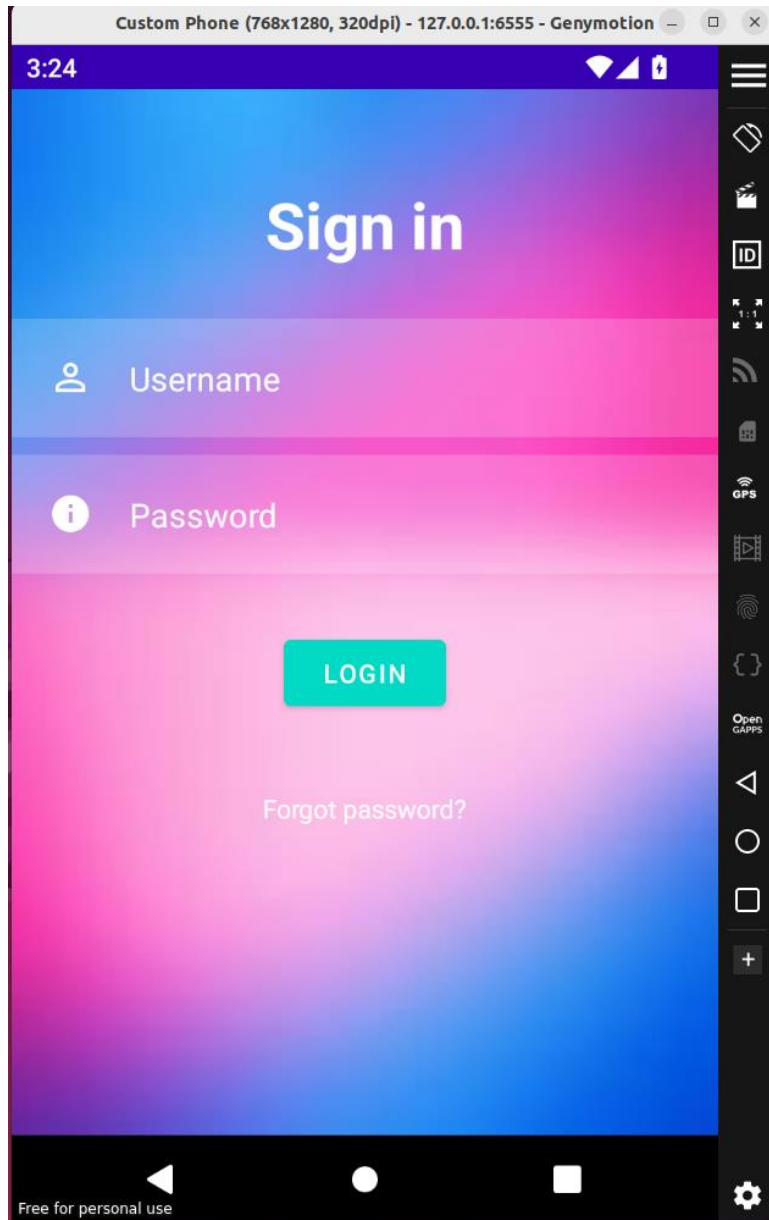
h. Có nên tình nghi đặc vụ đó là người đã thực hiện cuộc tấn công không? Tại sao?

Theo em thì hacker không phải Agent P, hacker không biết pass admin nên không phải là người get TABULAR ở packet 49495, nếu đã biết account thì không cần attack, thêm 1 tí bằng chứng ngoài lề là Agent P là đặc vụ giữ an toàn cho khu vực Tri-State Area nên khá chắc đặc vụ không làm việc này.

3. Android

Mở app thấy giao diện như này

Có vẻ yêu cầu ta Login thành công sẽ có flag



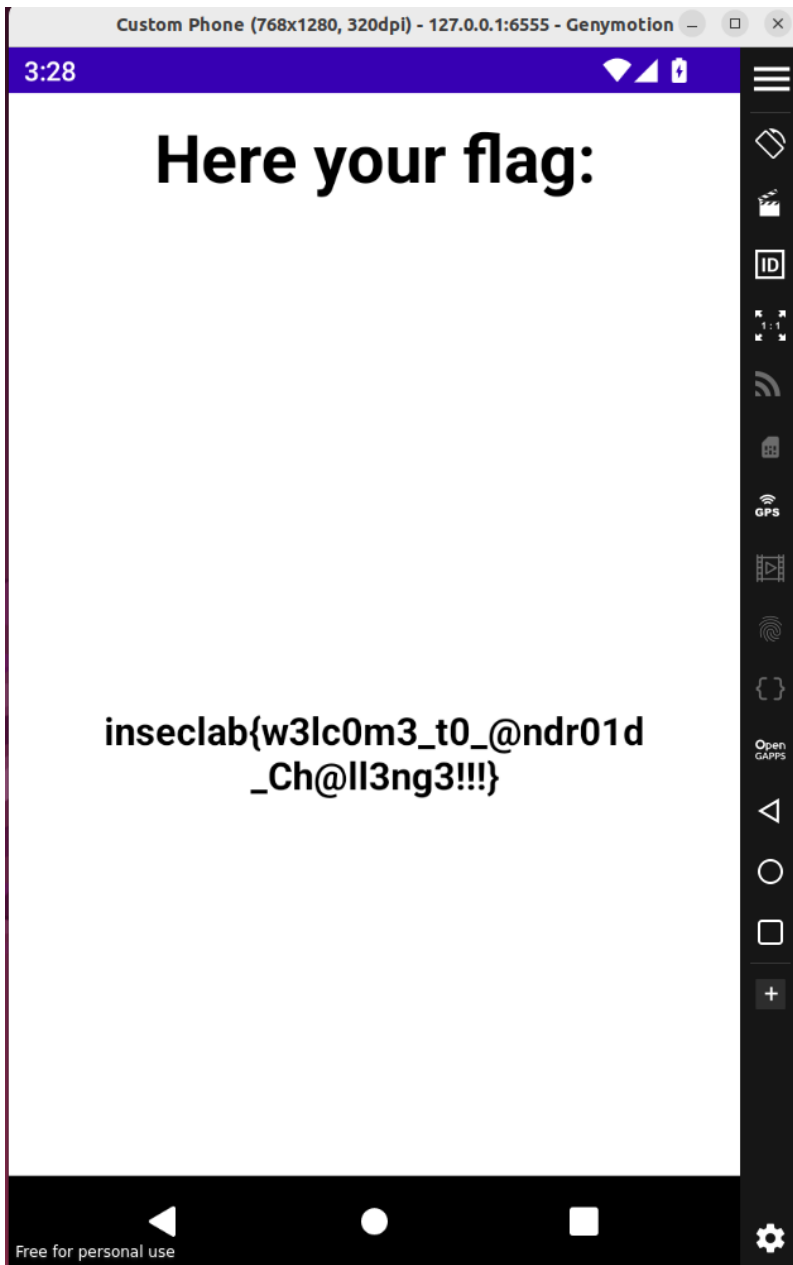
Dùng ByteCode-Viewer để đọc code.

Đọc và phân tích code hàm MainActivity

```
Work Space
com/example/trungquyenapp/MainActivity.class x
FernFlower Decompiler
[Download] [Upload] [Search] Match Case

1 package com.example.trungquyenapp;
2
3 import android.content.Intent;
4 import android.os.Bundle;
5 import android.view.View;
6 import android.widget.TextView;
7 import android.widget.Toast;
8 import androidx.appcompat.app.AppCompatActivity;
9 import com.google.android.material.button.MaterialButton;
10
11 public class MainActivity extends AppCompatActivity {
12     // $FF: synthetic method
13     public void lambda$onCreate$0$com_example_trungquyenapp_MainActivity(TextView var1, TextView var2, View var3) {
14         if (var1.getText().toString().equals("admin") && var2.getText().toString().equals("admin")) {
15             Toast.makeText(this, "LOGIN SUCCESSFUL", 0).show();
16             this.startActivity(new Intent(this, User.class));
17         } else {
18             Toast.makeText(this, "LOGIN FAILED !!!", 0).show();
19         }
20     }
21 }
22
23 protected void onCreate(Bundle var1) {
24     super.onCreate(var1);
25     this setContentView(2131427356);
26     TextView var2 = (TextView) this.findViewById(2131231157);
27     TextView var3 = (TextView) this.findViewById(2131231021);
28     ((MaterialButton) this.findViewById(2131230938)).setOnClickListener(new .ExternalSyntheticLambda0(this, var2, var3));
29 }
30 }
31
```

Ta thấy username và password là admin thì login thành công.



Ta có được flag: `inseclab{w3lc0m3_t0_@ndr01d_Ch@ll3ng3!!!}`

Đoạn mã bạn cung cấp chứa một số lỗi bảo mật nghiêm trọng liên quan đến việc xử lý thông tin đăng nhập và bảo vệ thông tin nhạy cảm. Dưới đây là phân tích về các lỗi và biện pháp khắc phục:

Lỗi bảo mật

1. Thông tin đăng nhập cứng (Hard-coded credentials):

- Thông tin đăng nhập `admin` được mã hóa cứng trong mã nguồn. Điều này có nghĩa là bất kỳ ai có quyền truy cập vào mã nguồn đều có thể dễ dàng nhìn thấy thông tin đăng nhập này.

```
if (var1.getText().toString().equals("admin") && var2.getText().toString().equals("admin")) {  
    Toast.makeText(this, "Login successful", Toast.LENGTH_SHORT).show();  
}
```

2. So sánh chuỗi không an toàn:

- So sánh chuỗi sử dụng phương thức ``equals`` mà không có biện pháp bảo vệ trước các tấn công thời gian (timing attacks).

- Điều này có thể bị lợi dụng để đoán mật khẩu dựa trên thời gian phản hồi của ứng dụng.

Biện pháp khắc phục

1. Loại bỏ thông tin đăng nhập cứng:

- Không bao giờ lưu trữ thông tin đăng nhập hoặc thông tin nhạy cảm trực tiếp trong mã nguồn. Thay vào đó, sử dụng cơ sở dữ liệu hoặc các dịch vụ xác thực bên ngoài.

- Sử dụng các phương pháp xác thực an toàn như OAuth, JWT, hoặc các phương pháp xác thực hai yếu tố (2FA).

2. Sử dụng các hàm băm và mã hóa an toàn:

- Mật khẩu nên được lưu trữ dưới dạng băm (hash) với một salt ngẫu nhiên và sử dụng các thuật toán băm mạnh như bcrypt, Argon2, hoặc PBKDF2.

- Khi người dùng đăng nhập, so sánh mật khẩu đã băm của họ với mật khẩu đã băm được lưu trữ.

3. Sử dụng biện pháp bảo vệ trước các tấn công thời gian:

- Sử dụng các thư viện bảo mật đã được kiểm chứng để so sánh các chuỗi an toàn, như thư viện ``MessageDigest`` hoặc ``SecretKeySpec`` trong Java.

Đề xuất mã sửa đổi

Dưới đây là một ví dụ về cách cải thiện đoạn mã đăng nhập:

```
public class MainActivity extends AppCompatActivity {

    private UserService userService; // Giả định có một lớp dịch vụ người dùng để xử lý xác thực

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main); // Sử dụng tên tài nguyên rõ ràng
        TextView usernameTextView = findViewById(R.id.username);
        TextView passwordTextView = findViewById(R.id.password);
        MaterialButton loginButton = findViewById(R.id.login_button);

        userService = new UserService(); // Khởi tạo dịch vụ người dùng

        loginButton.setOnClickListener(view -> {
            String username = usernameTextView.getText().toString();
            String password = passwordTextView.getText().toString();
            handleLogin(username, password);
        });

        private void handleLogin(String username, String password) {
            boolean loginSuccess = userService.authenticate(username, password);
            if (loginSuccess) {
                Toast.makeText(this, "LOGIN SUCCESSFUL", Toast.LENGTH_SHORT).show();
                startActivity(new Intent(this, UserActivity.class));
            } else {
                Toast.makeText(this, "LOGIN FAILED !!!", Toast.LENGTH_SHORT).show();
            }
        }
    }
}
```

Tóm tắt

- Loại bỏ thông tin đăng nhập cứng khỏi mã nguồn.
- Sử dụng cơ sở dữ liệu để lưu trữ thông tin người dùng và mật khẩu đã băm.
- Sử dụng các thư viện băm mạnh và so sánh chuỗi an toàn.
- Triển khai các biện pháp bảo mật như xác thực hai yếu tố (2FA) để tăng cường bảo mật.

4. Steganography

Sinh viên đọc kỹ yêu cầu trình bày bên dưới trang này

YÊU CẦU CHUNG

- Sinh viên tìm hiểu và thực hành theo hướng dẫn.
- Nộp báo cáo kết quả chi tiết những việc (**Report**) bạn đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (nếu có); giải thích cho quan sát (nếu có).
- Sinh viên báo cáo kết quả thực hiện và nộp bài.

Báo cáo:

- File **.DOCX và .PDF**. Tập trung vào nội dung, không mô tả lý thuyết.
- Nội dung trình bày bằng **Font chữ Times New Romans/ hoặc font chữ của mẫu báo cáo này (UTM Neo Sans Intel/UTM Viet Sach)– cỡ chữ 13. Canh đều (Justify) cho văn bản. Canh giữa (Center) cho ảnh chụp.**
- Đặt tên theo định dạng: [Mã lớp]-SessionX_GroupY. (trong đó X là Thứ tự buổi Thực hành, Y là số thứ tự Nhóm Thực hành đã đăng ký với GVHD-TH).

Ví dụ: [NT101.H11.1]-Session1_Group3.

- Nếu báo cáo có nhiều file, nén tất cả file vào file .ZIP với cùng tên file báo cáo.
- **Không đặt tên đúng định dạng – yêu cầu, sẽ KHÔNG chấm điểm bài Lab.**
- Nộp file báo cáo trên theo thời gian đã thống nhất tại courses.uit.edu.vn.

Đánh giá: Sinh viên hiểu và tự thực hiện được bài thực hành. Khuyến khích:

- Chuẩn bị tốt và đóng góp tích cực tại lớp.
- Có nội dung mở rộng, ứng dụng trong kịch bản phức tạp hơn, có đóng góp xây dựng bài thực hành.

Bài sao chép, trộm, ... sẽ được xử lý tùy mức độ vi phạm.

HẾT