1. labos koji ulazi u ZI

Tablica s particijama → počinje od retka 1B0, od stupca 0E

Iz datoteke od 2. labosa:

80 01 01 00 **83** FE 3F 08 **3F 00 00 00** **8A 34 02 00** 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

80 – označava da je particija aktivna (00 da nije aktivna)

01 01 00 – označavaju početni odnosno završni cilindar, glavu i sektor tvrdog diska

83 – označava datotečni sustav (kojim datotečnim sustavom je formirana particija) → u ovom slučaju je Linux Native Partition → https://www.win.tue.nl/~aeb/partitions/partition_types-1.html

3F 00 00 00 – označava početni sektor particije (zapisan u little-endian formatu) → particija počinje u sektoru 63 (3F pretvoren u decimalni broj)

8A 34 02 00 – Ukupan broj sektora koje sadrži particija → 2348A = 144522 sektora

Veličina u bajtovima: 144522 * 512 = 73 995 264 bajtova veličina sektora = 72 261 kilobajta

Početak particije = Broj početne particije * 512 = 63 * 512 = 32256 = 7E00 (HEX)

SUPERBLOCK start = početak particije + 1024 = 32256 + 1024 = 33280 = 8200 (HEX)

Selimo se na početak particije koji smo gore izračunali, tj. na SUPERBLOCK start zato što prvih 1024 bajtova nije iskorišteno → nakon pronalaska pratimo pomake u sljedećoj tablici

Offset	Size	Name	Description
0x0	__le32	s_inodes_count	Total inode count.
0x4	__le32	s_blocks_count_lo	Total block count.
0x8	__le32	s_r_blocks_count_lo	This number of blocks can only be allocated by the super-user.
0xC	__le32	s_free_blocks_count_lo	Free block count.
0x10	__le32	s_free_inodes_count	Free inode count.
0x14	__le32	s_first_data_block	First data block. This must be at least 1 for 1k-block filesystems and is typically 0 for all other block sizes.
0x18	__le32	s_log_block_size	Block size is 2 ^ (10 + s_log_block_size).
0x1C	__le32	s_log_cluster_size	Cluster size is (2 ^ s_log_cluster_size) blocks if bigalloc is enabled, zero otherwise.
0x20	__le32	s_blocks_per_group	Blocks per group.
0x24	__le32	s_clusters_per_group	Clusters per group, if bigalloc is enabled.
0x28	__le32	s_inodes_per_group	Inodes per group.

60 23 00 00 **91 46 00 00** **87 03 00 00** **AF 00 00 00** **CC 21 00 00** 00 00 00 00 **04 00 00 00** 02 00 00 00 91 46 00 00 91 46 00 00 60 23 00 00 00 00 00 00

inode count = 60 23 00 00 = 2360 (HEX) = 9056

blocks count = 91 46 00 00 = 4691 (HEX) = 18065

no of blocks that can only be allocated by the super-user = 87 03 00 00 = 387 (HEX) = 903

free block count = AF 00 00 00 = AF (HEX) = 175

free inode count = CC 21 00 00 = 21CC (HEX) = 8652

first data block = 00 00 00 00 = 0

block size = 04 00 00 00 = 4 (HEX) = 2 ^ (10 + block size) = 2 ^ (10 + 4) = 16384

cluster size = 02 00 00 00 = 2

blocks per group = 91 46 00 00 = 18065

clusters per group = 91 46 00 00 = 18065

inodes per group = 60 23 00 00 = 9056

mount time = 00 00 00 00 = 0

Block size **NIJE** ispravan, piše da se ovako računa:

(veličine particije – superblock veličina) / broj blokova

(73 995 264 – 1024) / 18 065 = 4096 → 2^12 što znači da umjesto 4 mora pisati 2

Autopsy:

MOGUĆE UBRZANJE → U Autopsy se mogu odabrati npr. sve datoteke pa napraviti Extract, onda u Notepad++ pomoću Searcha možete napraviti Find in Files i tražiti pojam koji može pomoći (i odabrati folder di ste exportali), **ispod ću navoditi na primjeru datoteke s labosa kako ubrzati**

Moguće je i unutar Autopsyja tražiti

(Moodle) What types and model of the device is this?

Tražiti **ro.product** pa se dobije npr.:

```
  D:\FER\9. SEMESTAR\RAČUNALNA FORENZIKA\LABORATORIJSKE
VJEŽBE\LAB02\Lab02NovaVjezba\Export\Vol2\404-system\build.prop (11 hits)
	Line 15: ro.product.model=Ascend Y300
	Line 16: ro.product.brand=huawei
	Line 17: ro.product.name=u8833
	Line 18: ro.product.device=u8833
	Line 19: ro.product.board=u8833
	Line 20: ro.product.cpu.abi=armeabi-v7a
	Line 21: ro.product.cpu.abi2=armeabi
	Line 22: ro.product.manufacturer=HUAWEI
	Line 23: ro.product.locale.language=en
	Line 24: ro.product.locale.region=US
	Line 27: # ro.build.product is obsolete; use ro.product.device
```

(Moodle) What is the device MAC address?

Ovo nisam siguran kako otkriti, jedino šta mi pada napamet je tražiti **Tag="Address"** šta izbaci onu MAC adresu šta mi je izbacilo adresu koja je bila točna.

(Moodle) Name at least 3 applications which were installed on the device?

vol2 → data → app i samo očitati apk datoteke

(Moodle) Write down SSID:password pairs of all known WiFi networks this device was connected to?

data → misc → wifi → kopati po datotekama i otkrije se da se WiFi mreže na koje smo se spajali nalaze u wpa_supplicant.conf

(Moodle) Find portable WiFi hotspot SSID and password?

data → misc → wifi → kopati po datotekama i otkrije se da se se WiFi hotspot mreža nalazi u hostapd.conf

(Moodle) When was the contact 'Ivan BBB' called and what was the duration of the call?

Pogledati Call Logs u Extracted Content

(Moodle) Which movie did the device user googled twice?

data → data → com.android.browser → databases → pregledati db datoteke i tamo se nalazi povijest svih pretraga

(Moodle) At which football match was picture in DCIM folder taken?

Pogledati metapodatke za sliku unutar DCIM foldera te ovisno o toj slici tražiti utakmicu koristeći Google.

2. labos koji ulazi u ZI

Ja sam kod sebe stavio volatility_2.5.win.standalone koji mi je skroz OK napravio posao i poslužio je (+ još je kao Windows executable datoteka pa nema potrebe za nekim dodatnim namještavanjima / kompajliranjima). Ja sam napravio opet na primjeru labosa (3. zadatak).

Task 1: There are 4 applications of interest running on this PC. Identify them.

ime_exe_datoteke -f **imeDatoteke** pslist > 1.txt

Identificirati procese koji se tiču baš pokrenutih aplikacija i zapisati njihove PID-ove.
U ovom slučaju:
mspaint.exe (520), IEXPLORE.exe (1612), notepad.exe (1944), calc.exe (1772)

Task 2: The suspect used one of the applications for the calculation of his profits. What is his profit going to be?

ime_exe_datoteke -f **imeDatoteke** editbox -p **PID od calc.exe** > 2.txt

Onda iz datoteke treba samo očitati rezultat koji je 12300

Task 3: The suspect used one of the applications for jotting down the secret codeword that will be used during the meeting. What is the codeword?

ime_exe_datoteke -f **imeDatoteke** editbox -p **PID od notepad.exe** > 3.txt
ILI
ime_exe_datoteke -f **imeDatoteke** notepad > 3.txt

Onda iz datoteke treba samo očitati rezultat koji je: Secret codeword is Q1W2E3ASDF

Task 4: The suspect used one of the applications for drawing a map. Determine where the meeting will take place.

ime_exe_datoteke -f **imeDatoteke** memdump -p **PID od mspaint.exe** --dump-dir **odabrana destinacija**

Promijeniti ekstenziju iz dmp u data, otvoriti u GIMP-u, postaviti širinu na 1394, okrenuti sliku da se može pročitati da se vidi rješenje **Cafe Maksimir – Mimice.**

Task 5: The suspect was reading an email which contained the exact time of the meeting. When will the meeting take place?

Ovdje ide yara. Prvo stvorite yara fajl po uzoru na objašnjenje u pdf-u. Nakon toga napravite search.

yara.txt

```
rule secret {
        strings:
                $secret = "meeting" wide ascii nocase
        condition:
                $secret
}
```

```
ime_exe_datoteke -f imeDatoteke yarascan --yara-file=yara.txt -p PID od IEXPLORE.exe > 5.txt
```

Onda je potrebno samo čitati stvorenu datoteku u koju je pisano i očitati točno vrijeme sastanka koje je 13:36.

3. labos koji ulazi u ZI

Otvoriti si Wireshark i 6. prezentaciju **Forenzika mreža** gledati pogotovo od 70. slajda

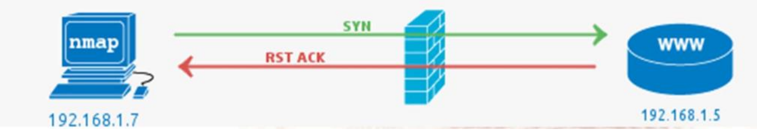
Rješenja:

	1	2	3	4	5
TYPE OF SCAN:	SYN stealth scan	Fin Scan	Null Scan	Xmas Tree Scan	IP Protocol Scan
COVERED PORTS:	23, 135	123, 445	2303, 2869	2303, 2869	80, 443, 40125 ili *
ATTACKER:	10.0.2.15	10.0.2.15	10.0.2.15	10.0.2.15	10.0.2.15
VICTIM:	192.168.1.3	192.168.1.3	192.168.1.3	192.168.1.3	192.168.1.3
OPEN PORTS:	135	/	/	/	/

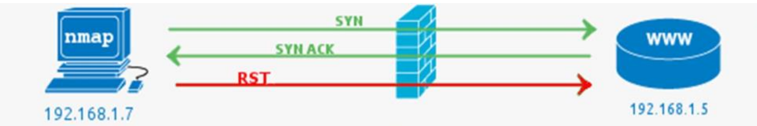
	6	7	8	9	10
ATTACK:	Password Guessing	ARP Poisoning	Teardrop	Heartbleed	Reverse Shell
SUCCESSFUL?	Yes	Yes	No	Yes	Yes

Važno još za napomenuti:

- Kad se prema nekom portu šalje paket, šalje se SYN
    - Ako dođe kao odgovor RST, ACK (i win je obično = 0) → port je **zatvoren**



    - Ako dođe kao odgovor SYN, ACK (i win je obično =/= 0) → port je **otvoren**



- Pazi s portovima 80 & 443 → ne idu u rješenje ako se ne izvršava skeniranje kako se obavlja za pojedini tip!!! (osim kod IP skeniranja zato što se tamo svi portovi skeniraju)

1. zadatak

SYN or Stealth scanning makes use of this procedure by sending a SYN packet and looking at the response. If SYN/ACK is sent back, the port is open and the remote end is trying to open a TCP connection. The scanner then sends an RST to tear down the connection before it can be established fully; often preventing the connection attempt appearing in application logs. If the port is closed, an RST will be sent. If it is filtered, the SYN packet will have been dropped and no response will be sent. In this way, Nmap can detect three port states - open, closed and filtered. Filtered ports may require further probing since they could be subject to firewall rules which render them open to some IPs or conditions, and closed to others.

Attacker & victim – gledamo tko prvi šalje zahtjev kome i tko odgovara

Po SYN, ACK prepoznajemo da je port 135 otvoren.

2. – 4. zadatak

XMAS - XMAS scans send a packet with the FIN, URG, and PSH flags set. If the port is open, there is no response; but if the port is closed, the target responds with a RST/ACK packet. XMAS scans work only on target systems that follow the RFC 793 implementation of TCP/IP and don't work against any version of Windows.

FIN - A FIN scan is similar to an XMAS scan but sends a packet with just the FIN flag set. FIN scans receive the same response and have the same limitations as XMAS scans. FIN A FIN scan is similar to an XMAS scan but sends a packet with just the FIN flag set. FIN scans receive the same response and have the same limitations as XMAS scans.

NULL - A NULL scan is also similar to XMAS and FIN in its limitations and response, but it just sends a packet with no flags set.

5. zadatak

IP scan – Šalje čisti IP packet bez dodatnih zaglavlja protokola na svaki port na ciljnom uređaju.

- Primitak ICMP Protocol Unreachable poruke nam kaže da se protokol ne koristi, inače se podrazumijeva da se koristi
- Ne šalju svi sustavi ICMP Protocol Unreachable poruke

The IP Protocol Scans attempt to determine the IP protocols supported on a target. Nmap sends a raw IP packet without any additional protocol header (see a good TCP/IP book for information about IP packets), to each protocol on the target machine. Receipt of an ICMP Protocol Unreachable message tells us the protocol is not in use, otherwise it is assumed open. Not all hosts send ICMP Protocol Unreachable messages. These may include firewalls, AIX, HP-UX and Digital UNIX). These machines will report all protocols open.

Covered Ports je UVIJEK * (svi).

6. zadatak

Password Guessing je dosta trivijalan za prepoznati – puno requesta i responsova s različitim lozinkama.

Uspješnost: Ako ima liniju RESPONSE: User Admin (ili bilo koji username) logged in.

7. zadatak

ARP spoofing is a type of attack in which a malicious actor sends falsified ARP (Address Resolution Protocol) messages over a local area network. This results in the linking of an attacker's MAC address with the IP address of a legitimate computer or server on the network. Once the attacker's MAC address is connected to an authentic IP address, the attacker will begin receiving any data that is intended for that IP address. ARP spoofing can enable malicious parties to intercept, modify or even stop data in-transit. ARP spoofing attacks can only occur on local area networks that utilize the Address Resolution Protocol.

Uglavnom, prepoznajemo ga po veeeelikom broju ARP poruka koje se šalju.

Uspješnost: Napad je uspješan ukoliko napadač dobije odgovor na neku od poruka (tj. ako je prihvaćen lažni upit s odgovorom tako da žrtva pošalje paket na neku od lažnih adresa).

8. zadatak

- **Napad fragmentacijom – "Teardrop"**
    - napadač šalje dva fragmenta koji se djelomično prekrivaju
        - "crash" kernela nakon sastavljanja fragmenata.
    - kad kernel primi drugi fragment uspoređuje ga s krajem prvog fragmenta
        - ako je pomak drugog fragmenta manji od kraja prvog, pokušava poravnati drugi frag. tako da se eliminira prekrivanje, "overlap"
        - pomak se postavi na kraj prvog te se izračunava nova duljina drugog segmenta kao razlika između nove pozicije kraja fragmenta i kraja starog fragmenta
            - no ako su originalni pomak drugog fragmenta i kraj manji od kraja prvog fragmenta tada nema dovoljno podatka kojim bi drugi fragment prekrio poravnanje te je nova duljina drugog fragmenta negativna
        - kad se ta negativna vrijednost preda funkciji memcpy koja očekuje broj bez predznaka to se interpretira kao veliki pozitivni broj
        - rezultat je reboot ili halt ovisno o količini fizičke memorije

Teardrop napad možemo prepoznati po Fragmented IP Protocol i BAD UDP LENGTH, a napad je uspješan tek kad dođe do zaustavljanja ili ponovnog pokretanja sustava (tj. do Denial of Service), a u ovom primjeru nismo sigurni jer ne znamo što se dalje dogodilo.

9. zadatak

The Heartbleed Bug is a serious vulnerability in the popular OpenSSL cryptographic software library. This weakness allows stealing the information protected, under normal conditions, by the SSL/TLS encryption used to secure the Internet. SSL/TLS provides communication security and privacy over the Internet for applications such as web, email, instant messaging (IM) and some virtual private networks (VPNs).

The Heartbleed bug allows anyone on the Internet to read the memory of the systems protected by the vulnerable versions of the OpenSSL software. This compromises the secret keys used to identify the service providers and to encrypt the traffic, the names and passwords of the users and the actual content. This allows attackers to eavesdrop on communications, steal data directly from the services and users and to impersonate services and users.

Heartbleed koristi Heartbeat za izvršavanje napada te tako i prepoznajemo da je riječ o Heartbleed napadu. Napadač može kontrolirati veličinu heartbeat-a i strukturirati ga da bude veći od očekivanog. Kroz vrijeme, napadač može rekonstruirati cijelu žrtvinu memoriju.

Uspješnost: Napad je uspješan ukoliko je napadač uspio dobiti podatke od žrtve. To možemo vidjeti tako da pogledamo zapise u datoteci gdje vidimo da se šalju segmenti (a i također možemo vidjeti pomoću praćenja TCP streama da su svi ti segmenti kad se spoje zapravo podatci s računala žrtve).

10. zadatak

A reverse shell is a type of shell in which the target machine communicates back to the attacking machine. The attacking machine has a listener port on which it receives the connection, which by using, code or command execution is achieved.

Napad možemo prepoznati po tome što nakon 3-way handshakea napadač šalje prema žrtvi PUSH i ACK zastavice čime zapravo inicira uspostavljanje veze.

Uspješnost: Možemo pratiti TCP stream i ukoliko vidimo da se naredba izvršila na računalu žrtvi, znamo da je napad bio uspješan.