ENG

BUILD WEEK 1

BYTE REBELS

All Designed by:
CANNAVACCIUOLO DAVIDE
DI MAIO PAOLO
FORLENZA SIMONE
RUSSO FEDERICO - LEADER
TIZZI FEDERICO
VAN ZWAM ARJEN

TASK

ENG 02

We were hired by the Theta company to perform security assessments on some of their critical data center infrastructure.

The scope of activities is mainly focused on:

-A Web server that exposes various services on the internet (and therefore accessible to the public)

-An Application server that exposes on the internal network an e-commerce application accessible only by employees of the Theta company (therefore not accessible from external sources, i.e. the internet) Based on the information above, the head of IT security of Theta, also called CISO (chief information security officer), requires us:

1. To propose a network model (design) to secure the two critical components, including in the analysis the security devices that could be used to increase network protection.
2. To carry out specific tests on the two critical components to evaluate their safety status. In this case, the CISO asks us to carry out the checks reported in the next slide.

On the Web Server:

● Scan for active services on the machine.

● Possible enumeration of HTTP methods enabled on the HTTP service listening on port 80. On the application server:

● Enumeration of enabled HTTP methods.

● Evaluation of the robustness of the login page to Brute Force attacks. The CISO explicitly asked us not to carry out any invasive tests in the production environment, and therefore we proposed to him to reproduce the two components in our test laboratories, so as to be able to carry out the tests safely, separating the test environments from the Work.

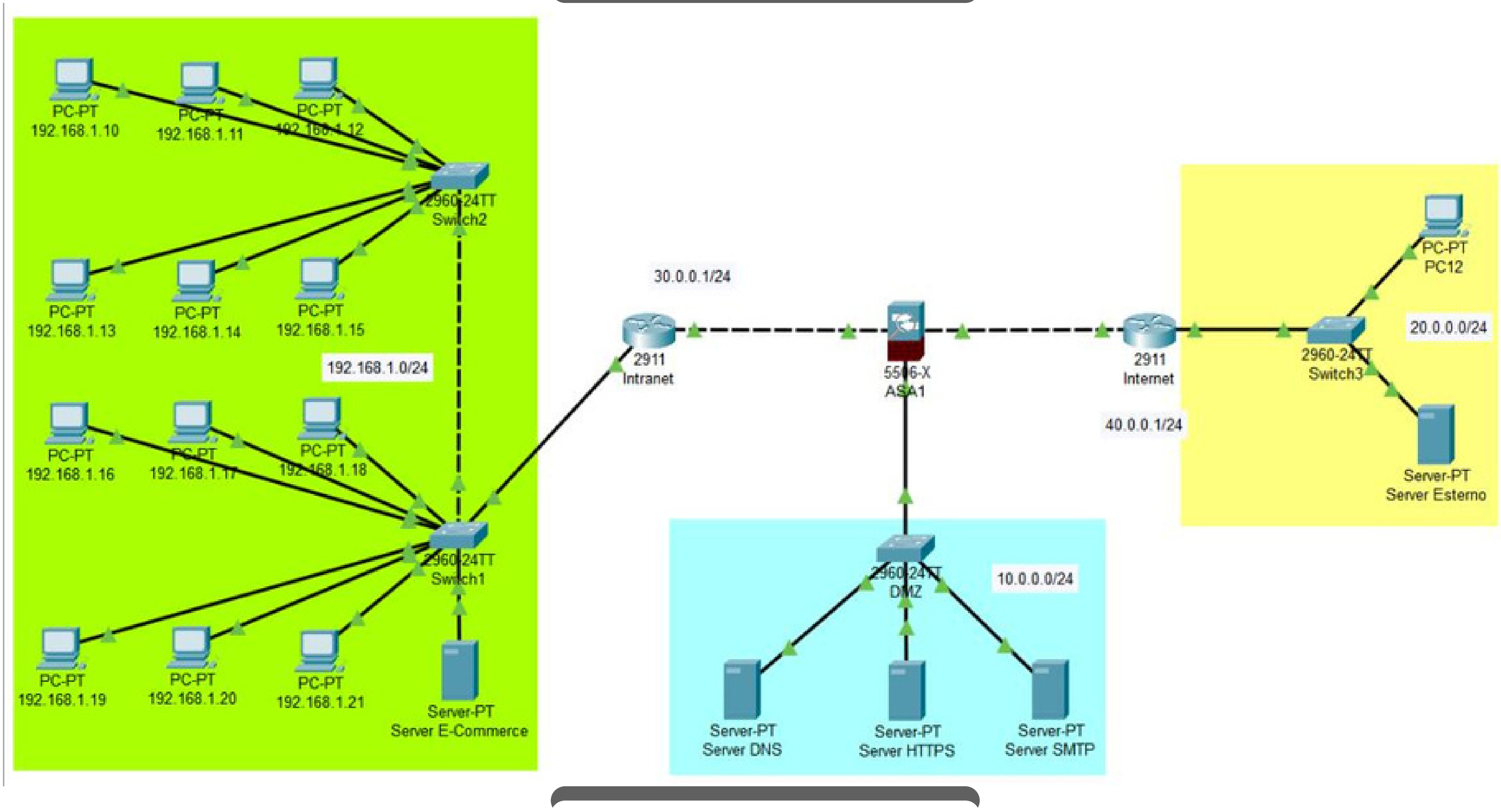
THETA STRUCTURE

The Theta corporate IT structure has been reconstructed to make it easier to understand.

There are **Intranet** areas (where it is not accessible from the outside, therefore from the internet, but only from the company network)

There is an internal e-commerce **server** (also accessible only from the intranet.

Finally, a **DMZ** with a **web server** accessible to all.



To drastically and immediately improve network security, we hypothesize the insertion of a **perimeter Firewall**, positioning it between the internal network and the external network (for example the Internet) and acting as a **defense barrier** that *controls* and *filters* incoming network traffic and output. In summary it will be used for: monitor traffic, filter packets, proxy, VPN and to create custom security rules

ENG

04

WEB SERVER

SERVER THAT EXPOSES VARIOUS SERVICES ON THE INTERNET

[redacted]

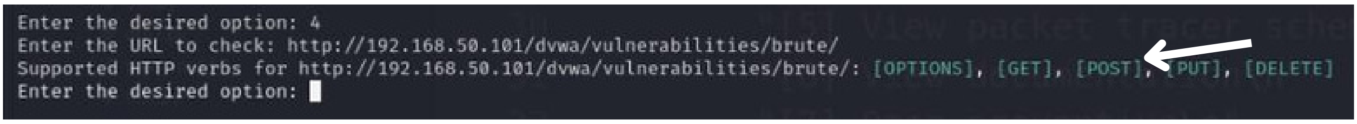
SCAN OF ACTIVE SERVICES ON THE MACHINE

The **Metasploitable2** machine was simulated on the **Theta** Web Server.
We therefore simulated a port scanner of the services on the **Theta** server with output the list of open and closed ports.

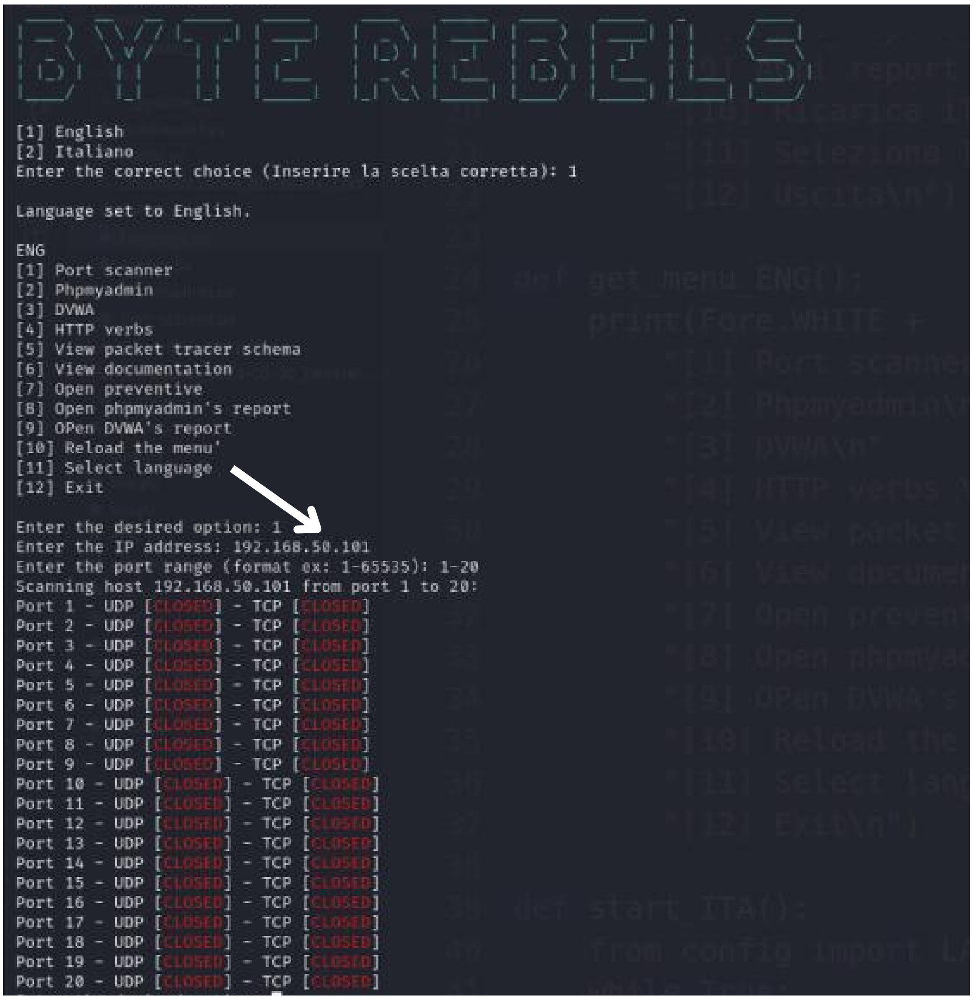
[redacted]

ENUMERATION OF ENABLED HTTP METHODS

We simulated a series of **HTTP** requests to the server in order to determine which HTTP **verbs** are supported for further analysis.



```
Enter the desired option: 4
Enter the URL to check: http://192.168.50.101/dvwa/vulnerabilities/brute/
Supported HTTP verbs for http://192.168.50.101/dvwa/vulnerabilities/brute/: [OPTIONS], [GET], [POST], [PUT], [DELETE]
Enter the desired option: 
```



```
[1] English
[2] Italiano
Enter the correct choice (Inserire la scelta corretta): 1

Language set to English.

ENG
[1] Port scanner
[2] Phpmyadmin
[3] DVWA
[4] HTTP verbs
[5] View packet tracer schema
[6] View documentation
[7] Open preventive
[8] Open phpmyadmin's report
[9] OPen DVWA's report
[10] Reload the menu'
[11] Select language
[12] Exit

Enter the desired option: 1
Enter the IP address: 192.168.50.101
Enter the port range (format ex: 1-65535): 1-20
Scanning host 192.168.50.101 from port 1 to 20:
Port 1 - UDP [CLOSED] - TCP [CLOSED]
Port 2 - UDP [CLOSED] - TCP [CLOSED]
Port 3 - UDP [CLOSED] - TCP [CLOSED]
Port 4 - UDP [CLOSED] - TCP [CLOSED]
Port 5 - UDP [CLOSED] - TCP [CLOSED]
Port 6 - UDP [CLOSED] - TCP [CLOSED]
Port 7 - UDP [CLOSED] - TCP [CLOSED]
Port 8 - UDP [CLOSED] - TCP [CLOSED]
Port 9 - UDP [CLOSED] - TCP [CLOSED]
Port 10 - UDP [CLOSED] - TCP [CLOSED]
Port 11 - UDP [CLOSED] - TCP [CLOSED]
Port 12 - UDP [CLOSED] - TCP [CLOSED]
Port 13 - UDP [CLOSED] - TCP [CLOSED]
Port 14 - UDP [CLOSED] - TCP [CLOSED]
Port 15 - UDP [CLOSED] - TCP [CLOSED]
Port 16 - UDP [CLOSED] - TCP [CLOSED]
Port 17 - UDP [CLOSED] - TCP [CLOSED]
Port 18 - UDP [CLOSED] - TCP [CLOSED]
Port 19 - UDP [CLOSED] - TCP [CLOSED]
Port 20 - UDP [CLOSED] - TCP [CLOSED]
```

APPLICATION SERVER

E-COMMERCE SERVER ONLY ON INTRANET

ENG

05

[redacted]

ENUMERATION OF ENABLED HTTP METHODS

We simulated a series of **HTTP** requests to the server in order to determine which HTTP **verbs** are supported for further analysis.

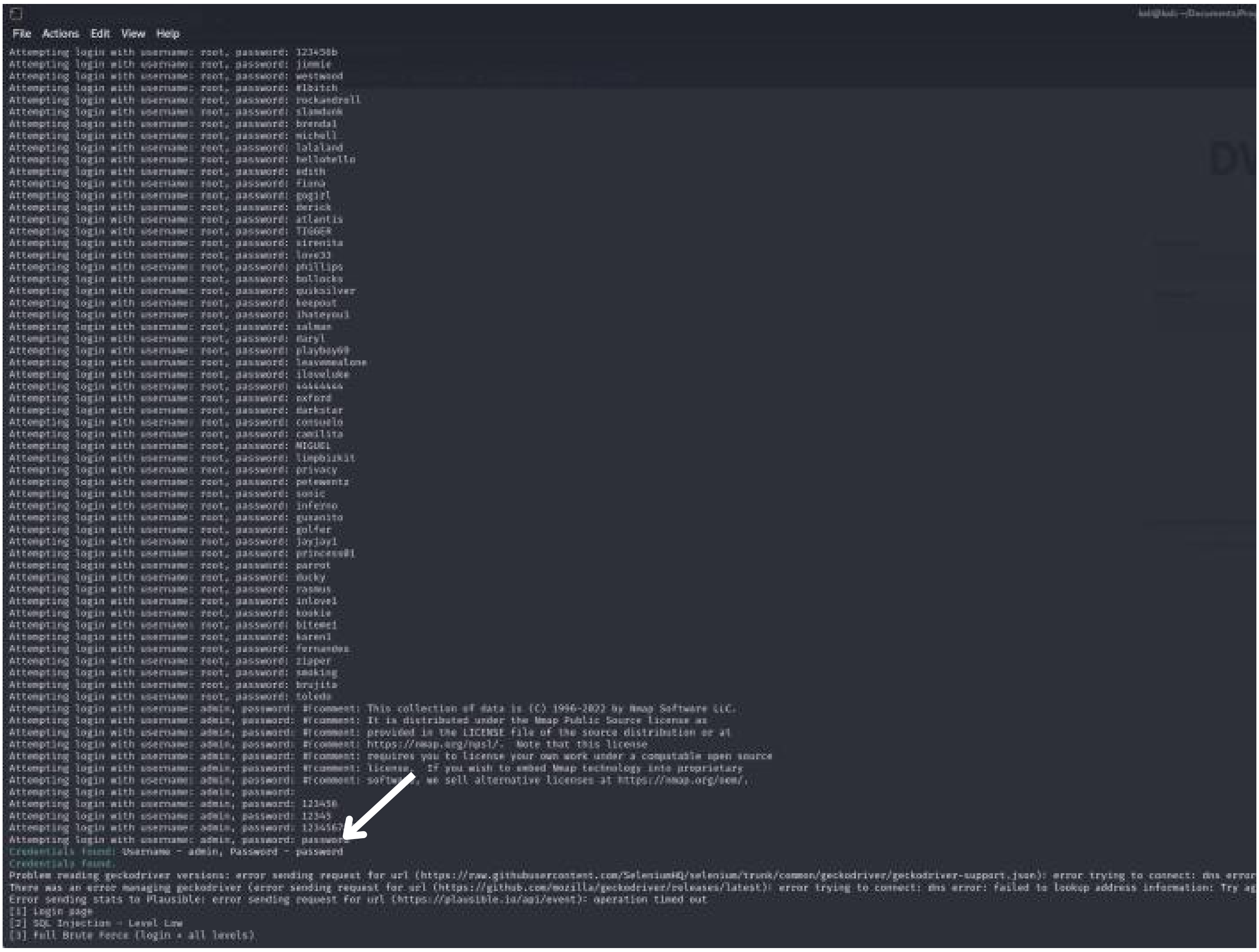
[redacted]

EVALUATION OF THE ROBUSTNESS OF THE LOGIN PAGE TO BRUTE FORCE ATTACKS

We simulated several **Brute Force** attacks to test the actual security of the **login forms**.

[redacted]

We have detected multiple fragilities IN ALL the company's login forms, we will subsequently explain why and how to resolve them.



```
Attempting login with username: root, password: 123456b
Attempting login with username: root, password: jimmie
Attempting login with username: root, password: westwood
Attempting login with username: root, password: #1bitch
Attempting login with username: root, password: rockandroll
Attempting login with username: root, password: slamdunk
Attempting login with username: root, password: brenda1
Attempting login with username: root, password: michell
Attempting login with username: root, password: lalaland
Attempting login with username: root, password: hellohello
Attempting login with username: root, password: edith
Attempting login with username: root, password: fiona
Attempting login with username: root, password: gogirl
Attempting login with username: root, password: derick
Attempting login with username: root, password: atlantis
Attempting login with username: root, password: TIGGER
Attempting login with username: root, password: sirenita
Attempting login with username: root, password: love33
Attempting login with username: root, password: phillips
Attempting login with username: root, password: bollocks
Attempting login with username: root, password: quiksilver
Attempting login with username: root, password: keepout
Attempting login with username: root, password: ihateyou1
Attempting login with username: root, password: salman
Attempting login with username: root, password: daryl
Attempting login with username: root, password: playboy69
Attempting login with username: root, password: leavemealone
Attempting login with username: root, password: iloveluke
Attempting login with username: root, password: 4444444
Attempting login with username: root, password: oxford
Attempting login with username: root, password: darkstar
Attempting login with username: root, password: consuelo
Attempting login with username: root, password: camilita
Attempting login with username: root, password: MIGUEL
Attempting login with username: root, password: limpbizkit
Attempting login with username: root, password: privacy
Attempting login with username: root, password: petewentz
Attempting login with username: root, password: sonic
Attempting login with username: root, password: inferno
Attempting login with username: root, password: gusanito
Attempting login with username: root, password: golfer
Attempting login with username: root, password: joyjoy1
Attempting login with username: root, password: princess01
Attempting login with username: root, password: parrot
Attempting login with username: root, password: ducky
Attempting login with username: root, password: rasmus
Attempting login with username: root, password: inlove1
Attempting login with username: root, password: kookie
Attempting login with username: root, password: biteme1
Attempting login with username: root, password: karen1
Attempting login with username: root, password: fernandes
Attempting login with username: root, password: zipper
Attempting login with username: root, password: smoking
Attempting login with username: root, password: brujita
Attempting login with username: root, password: toledo
Attempting login with username: admin, password: #!comment: This collection of data is (C) 1996-2022 by Nmap Software LLC.
Attempting login with username: admin, password: #!comment: It is distributed under the Nmap Public Source license as
Attempting login with username: admin, password: #!comment: provided in the LICENSE file of the source distribution or at
Attempting login with username: admin, password: #!comment: https://nmap.org/npsl/.  Note that this license
Attempting login with username: admin, password: #!comment: requires you to license your own work under a compatable open source
Attempting login with username: admin, password: #!comment: license.  If you wish to embed Nmap technology into proprietary
Attempting login with username: admin, password: #!comment: software, we sell alternative licenses at https://nmap.org/oem/.
Attempting login with username: admin, password:
Attempting login with username: admin, password: 123456
Attempting login with username: admin, password: 12345
Attempting login with username: admin, password: 1234567
Attempting login with username: admin, password: password
Credentials found: Username - admin, Password - password
Credentials found.
Problem reading geckodriver versions: error sending request for url (https://raw.githubusercontent.com/SeleniumHQ/selenium/trunk/common/geckodriver/geckodriver-support.json): error trying to connect: dns error
There was an error managing geckodriver (error sending request for url (https://github.com/mozilla/geckodriver/releases/latest): error trying to connect: dns error: failed to lookup address information: Try ag
Error sending stats to Plausible: error sending request for url (https://plausible.io/api/event): operation timed out
[1] Login page
[2] SQL Injection - Level Low
[3] Full Brute Force (login + all levels)
```

APPLICATION SERVER

ENG

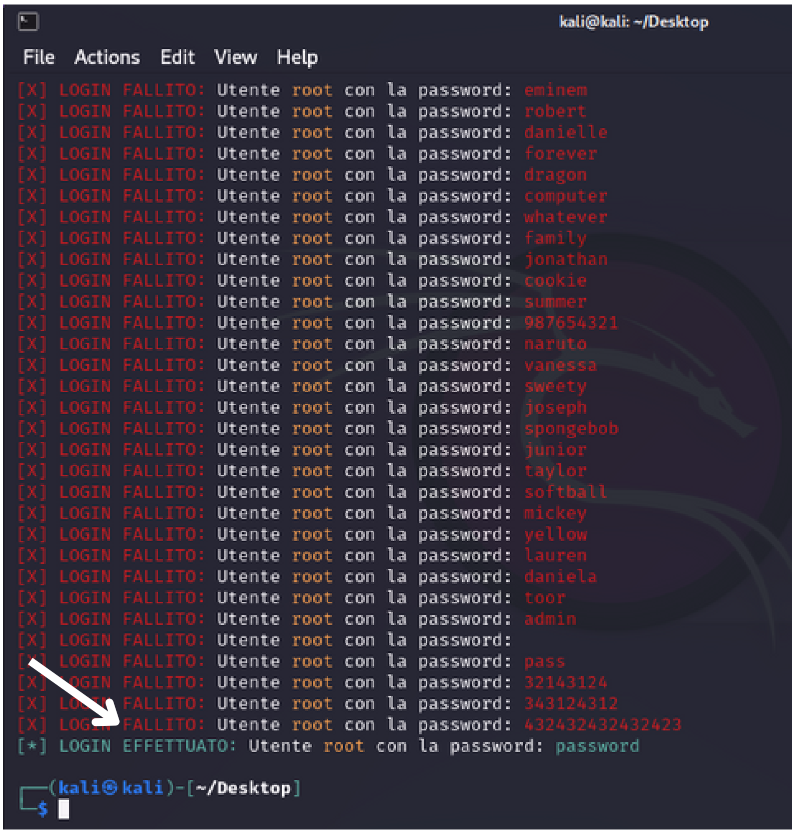
06

E-COMMERCE SERVER ONLY ON INTRANET

RESULTS: VERY VULNERABLES

With the tests carried out in complete **safety** in our virtual machines we can state that the login pages are totally vulnerable to ANY **attack**, here are some screenshots:

Brute Force attack inside the index.php in the login **form**

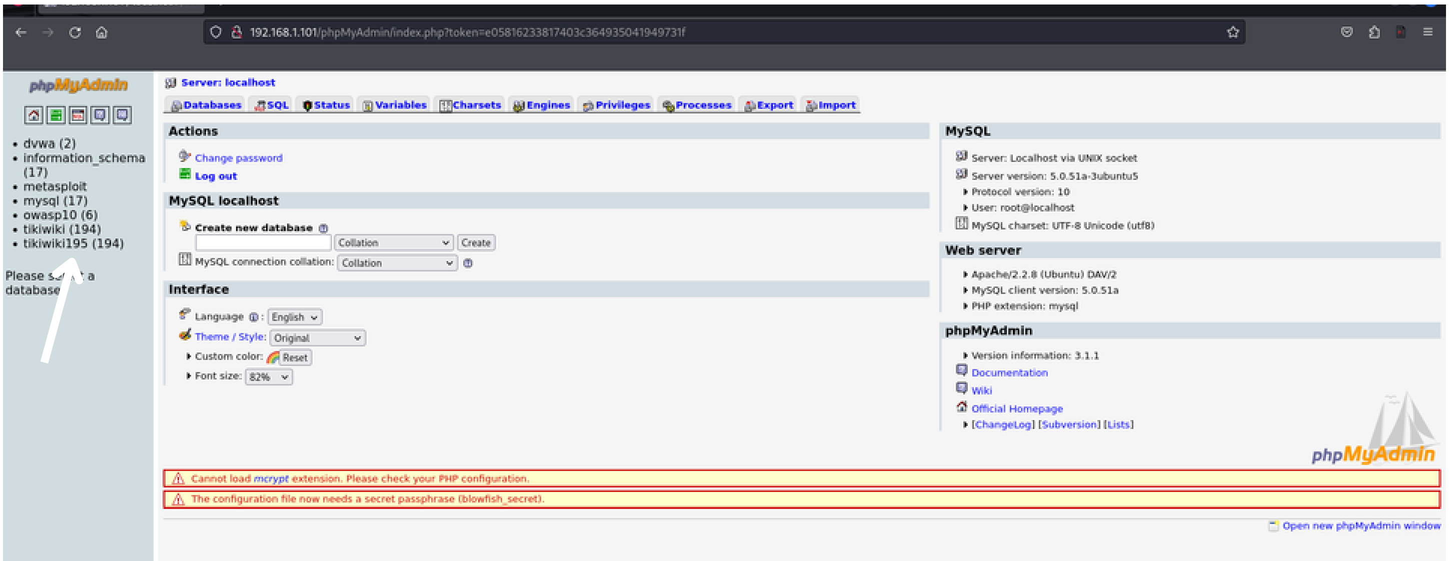


```
[X] LOGIN FALLITO: Utente root con la password: eminem
[X] LOGIN FALLITO: Utente root con la password: robert
[X] LOGIN FALLITO: Utente root con la password: danielle
[X] LOGIN FALLITO: Utente root con la password: forever
[X] LOGIN FALLITO: Utente root con la password: dragon
[X] LOGIN FALLITO: Utente root con la password: computer
[X] LOGIN FALLITO: Utente root con la password: whatever
[X] LOGIN FALLITO: Utente root con la password: family
[X] LOGIN FALLITO: Utente root con la password: jonathan
[X] LOGIN FALLITO: Utente root con la password: cookie
[X] LOGIN FALLITO: Utente root con la password: summer
[X] LOGIN FALLITO: Utente root con la password: 987654321
[X] LOGIN FALLITO: Utente root con la password: naruto
[X] LOGIN FALLITO: Utente root con la password: vanessa
[X] LOGIN FALLITO: Utente root con la password: sweety
[X] LOGIN FALLITO: Utente root con la password: joseph
[X] LOGIN FALLITO: Utente root con la password: spongebob
[X] LOGIN FALLITO: Utente root con la password: junior
[X] LOGIN FALLITO: Utente root con la password: taylor
[X] LOGIN FALLITO: Utente root con la password: softball
[X] LOGIN FALLITO: Utente root con la password: mickey
[X] LOGIN FALLITO: Utente root con la password: yellow
[X] LOGIN FALLITO: Utente root con la password: lauren
[X] LOGIN FALLITO: Utente root con la password: daniela
[X] LOGIN FALLITO: Utente root con la password: toor
[X] LOGIN FALLITO: Utente root con la password: admin
[X] LOGIN FALLITO: Utente root con la password:
[X] LOGIN FALLITO: Utente root con la password: pass
[X] LOGIN FALLITO: Utente root con la password: 32143124
[X] LOGIN FALLITO: Utente root con la password: 343124312
[X] LOGIN FALLITO: Utente root con la password: 432432432432423
[*] LOGIN EFFETTUATO: Utente root con la password: password

┌──(kali㉿kali)-[~/Desktop]
└─$ 
```

[redacted]

Login with the relevant **credentials**



192.168.1.101/phpMyAdmin/index.php?token=e05816233817403c364935041949731f

- dvwa (2)
- information_schema (17)
- metasploit
- mysql (17)
- owasp10 (6)
- tikiwiki (194)
- tikiwiki195 (194)

Please select a database

Server: localhost

Actions

MySQL localhost

Create new database: Collation

MySQL connection collation: Collation

Interface

Language: English

Theme / Style: Original

Custom color: Reset

Font size: 82%

MySQL

- Server: Localhost via UNIX socket
- Server version: 5.0.51a-3ubuntu5
- Protocol version: 10
- User: root@localhost
- MySQL charset: UTF-8 Unicode (utf8)

Web server

- Apache/2.2.8 (Ubuntu) DAV/2
- MySQL client version: 5.0.51a
- PHP extension: mysql

phpMyAdmin

- Version information: 3.1.1

⚠ Cannot load *mcrypt* extension. Please check your PHP configuration.

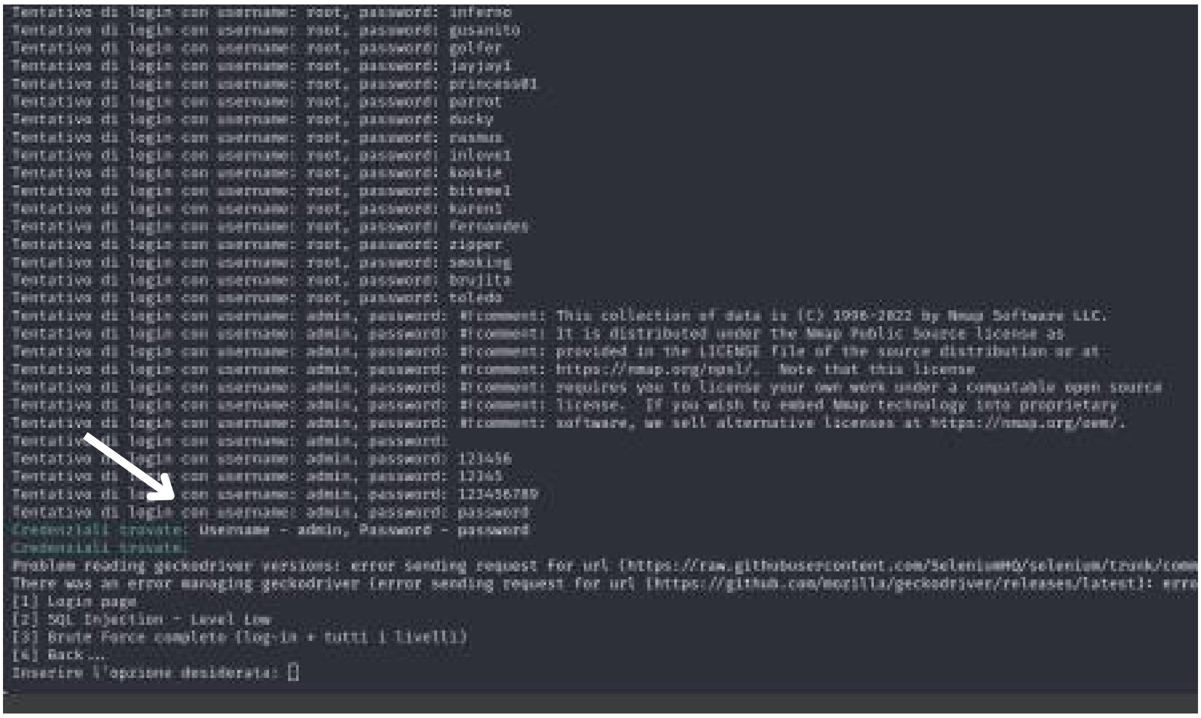
⚠ The configuration file now needs a secret passphrase (blowfish_secret).

APPLICATION SERVER

E-COMMERCE SERVER ONLY ON INTRANET

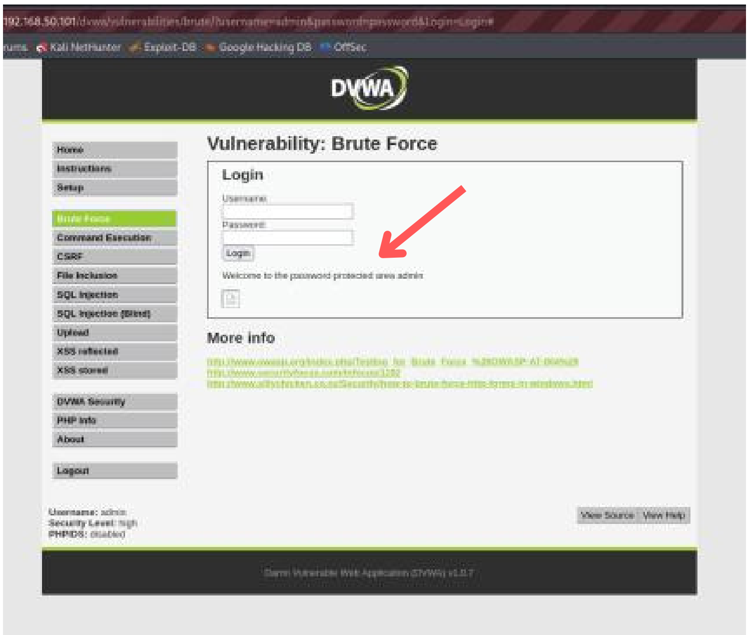
RESULTS: VERY VULNERABLES

The server that hosts the e-commerce has also highlighted very obvious flaws in the **DVWA** login page, where it is also possible to perform a **sql injection** inside as well as using Brute Force



```
Tentativo di login con username: root, password: inferno
Tentativo di login con username: root, password: gusanito
Tentativo di login con username: root, password: golfer
Tentativo di login con username: root, password: jayjay1
Tentativo di login con username: root, password: princesa01
Tentativo di login con username: root, password: parrot
Tentativo di login con username: root, password: ducky
Tentativo di login con username: root, password: rasmus
Tentativo di login con username: root, password: inlove1
Tentativo di login con username: root, password: kookie
Tentativo di login con username: root, password: biteme1
Tentativo di login con username: root, password: karen1
Tentativo di login con username: root, password: fernandes
Tentativo di login con username: root, password: zipper
Tentativo di login con username: root, password: smoking
Tentativo di login con username: root, password: brujita
Tentativo di login con username: root, password: toledo
Tentativo di login con username: admin, password: #comment: This collection of data is (C) 1996-2022 by Nmap Software LLC.
Tentativo di login con username: admin, password: #comment: It is distributed under the Nmap Public Source license as
Tentativo di login con username: admin, password: #comment: provided in the LICENSE file of the source distribution or at
Tentativo di login con username: admin, password: #comment: https://nmap.org/npsl/.  Note that this license
Tentativo di login con username: admin, password: #comment: requires you to license your own work under a compatable open source
Tentativo di login con username: admin, password: #comment: license.  If you wish to embed Nmap technology into proprietary
Tentativo di login con username: admin, password: #comment: software, we sell alternative licenses at https://nmap.org/oem/.
Tentativo di login con username: admin, password:
Tentativo di login con username: admin, password: 123456
Tentativo di login con username: admin, password: 12345
Tentativo di login con username: admin, password: 123456789
Tentativo di login con username: admin, password: password
Credenziali trovate: Username - admin, Password - password
Credenziali trovate.
Problem reading geckodriver versions: error sending request for url (https://raw.githubusercontent.com/SeleniumHQ/selenium/trunk/comm
There was an error managing geckodriver (error sending request for url (https://github.com/mozilla/geckodriver/releases/latest): erro
[1] Login page
[2] SQL Injection - Level Low
[3] Brute Force completo (log-in + tutti i livelli)
[4] Back ...
Inserire l'opzione desiderata: 
```

Brute Force **attack** inside the **DVWA** login page



192.168.50.101/dvwa/vulnerabilities/brute/?username=admin&password=password&Login=Login#

Vulnerability: Brute Force

Login

Username:

Password:

Login

Welcome to the password protected area admin

More info

http://www.owasp.org/index.php/Testing_for_Brute_Force_%28OWASP-AT-004%29
http://www.securityfocus.com/infocus/1192
http://www.sillychicken.co.nz/Security/how-to-brute-force-http-forms-in-windows.html

Username: admin
Security Level: high
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.0.7

Furthermore, we have also implemented an **AUTOMATIC** login via the python script which **logs** in once the correct credentials are found!

HOW TO IMPROVE

The services we tested in totally secure environments highlighted multiple flaws.
Below we have compiled a list of how to improve security:

In the **LOW** security level of the DVWA (accessible with sql injection)
1. **Using Query Parameterization or ORM**: Use parameterized SQL statements or an Object-Relational Mapping (ORM) to avoid SQL injection attacks.
2. **Using Secure Hash Algorithms**: Replace MD5 hashing with more secure hashing algorithms such as bcrypt or Argon2 to protect passwords from rainbow table hash attacks.
3. **Lockout Mechanism**: Implement an account lockout mechanism after a certain number of failed attempts.

In the **MEDIUM** safety level of the DVWA
1. **Using Modern Sanitization Features:** Use modern sanitization features such as `mysqli_real_escape_string` or query parameterization to prevent SQL injection attacks.
2. **Using Secure Hash Algorithms**: Replace MD5 hashing with more secure hashing algorithms such as bcrypt or Argon2 to protect passwords from rainbow table hash attacks.

In the **HIGH** security level of the DVWA
1. **Using Modern Sanitization Features**: Use modern sanitization features such as `mysqli_real_escape_string` or query parameterization to prevent SQL injection attacks.
2. **Using Secure Hash Algorithms**: Replace MD5 hashing with more secure hashing algorithms such as bcrypt or Argon2 to protect passwords from rainbow table hash attacks.
3. **Implement Anti-Brute Force Controls**: Implement brute force attack detection and mitigation mechanisms, such as login attempt limits, captchas, or escalating delays.

HOW TO IMPROVE

The services we tested in totally secure environments highlighted multiple flaws.
Below we have compiled a list of how to improve security:

phpMyAdmin's index.php page
There can be multiple _solutions_:

1) **Access monitoring**: Implementation of controls on requests from the same IP and consequently a temporary block/ban of that particular IP
2) **Attempt Limitations**: If a login page does not enforce any limitations on login attempts, an attacker can make an unlimited number of attempts without restrictions. This allows them to carry out brute force attacks without hindrance.
3) **Weak passwords**: If users use weak or easily guessable passwords, it becomes easier for an attacker to identify the correct combinations via a brute force attack.
4) **Attack detection mechanisms**: Login pages must have brute force attack detection, not having them could allow an attacker to carry out the attack without being detected or blocked by the system.
5) **Use multi-factor authentication (MFA) measures**: Using MFA, such as sending a verification code via SMS or using an authenticator application, can make it more difficult for an attacker to obtain the even if he can guess the password.

We remind you that the quote we have already sent lasts 30 days