

## ESERCIZIO W13D1

L'obiettivo viene esposto dalla seguente traccia in piattaforma:

- Configurare il vostro laboratorio virtuale in modo tale che la macchina Metasploitable sia raggiungibile dalla macchina Kali Linux. Assicuratevi che ci sia comunicazione tra le due macchine.
- Lo scopo dell'esercizio di oggi è sfruttare la vulnerabilità di «file upload» presente sulla DVWA per prendere controllo della macchina ed eseguire dei comandi da remoto tramite una shell in PHP.
- Inoltre, per familiarizzare sempre di più con gli strumenti utilizzati dagli Hacker Etici, vi chiediamo di intercettare ed analizzare ogni richiesta verso la DVWA con BurpSuite.


### Preparazione macchine virtuali

Configuriamo le nostre macchine su rete interna in modo che riescano a comunicare tra di loro.

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# System
auto eth0
#iface eth0 inet dhcp
iface eth0 inet static
address 192.168.32.100/24
netmask 255.255.255.0
gateway 192.168.32.1
```



```
GNU nano 2.0.7 File: /etc/network/interfaces

* Reloading OpenBSD Secure Shell server's configuration sshdr system
# and how to activate them. For more information, see interfa ...done.
* Reloa

ding Postfix configuration...ace
auto lo ...done.
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
#iface eth0 inet dhcp
address 192.168.32.101
netmask 255.255.255.0
network 192.168.50.0
broadcast 192.168.50.255
gateway 192.168.32.1

[ Wrote 16 lines ]
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

Settaggi rete  
Metasploitable2

```
(kali@kali)-[~]
$ ping 192.168.32.101
PING 192.168.32.101 (192.168.32.101) 56(84) bytes of data.
64 bytes from 192.168.32.101: icmp_seq=1 ttl=64 time=0.942 ms
64 bytes from 192.168.32.101: icmp_seq=2 ttl=64 time=0.762 ms
64 bytes from 192.168.32.101: icmp_seq=3 ttl=64 time=0.784 ms
64 bytes from 192.168.32.101: icmp_seq=4 ttl=64 time=0.689 ms
^C
--- 192.168.32.101 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3071ms
rtt min/avg/max/mdev = 0.689/0.794/0.942/0.092 ms

(kali@kali)-[~]
$
```

Kali ping Meta

```
msfadmin@metasploitable:~$ ping 192.168.32.100
PING 192.168.32.100 (192.168.32.100) 56(84) bytes of data.
64 bytes from 192.168.32.100: icmp_seq=1 ttl=64 time=0.936 ms
64 bytes from 192.168.32.100: icmp_seq=2 ttl=64 time=0.647 ms
64 bytes from 192.168.32.100: icmp_seq=3 ttl=64 time=0.657 ms
64 bytes from 192.168.32.100: icmp_seq=4 ttl=64 time=0.784 ms
--- 192.168.32.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2999ms
rtt min/avg/max/mdev = 0.647/0.756/0.936/0.117 ms
msfadmin@metasploitable:~$
```

Meta ping  
Kali

## Intercettazione richieste

Grazie all'aiuto dell'intercepting proxy Burpsuite, possiamo intercettare le richieste inviate dal nostro browser fino al web server.

Richiesta GET  
durante  
immissione  
credenziali

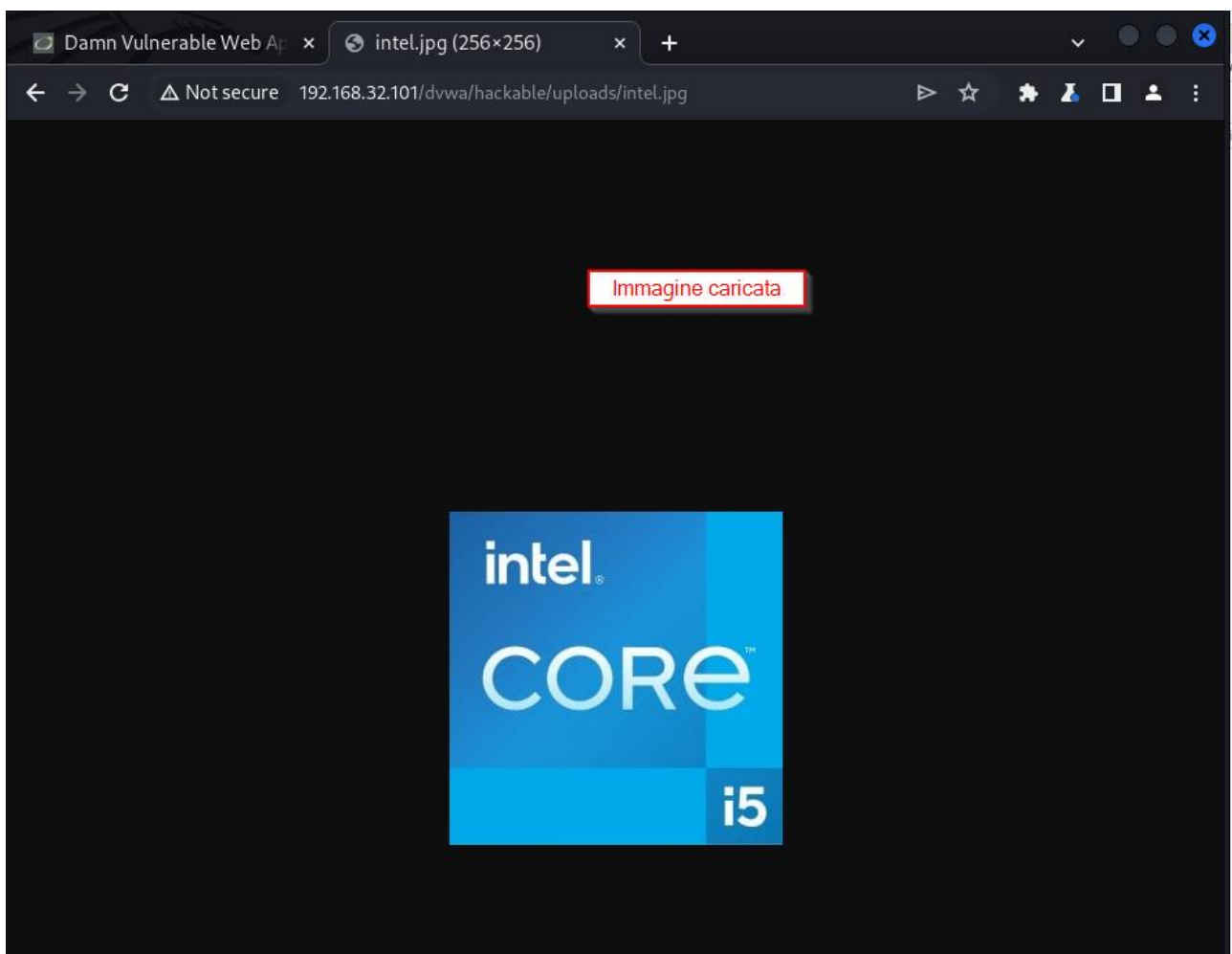
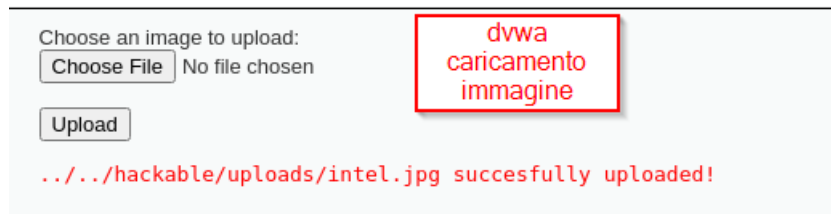
La prima ci mostra il passaggio di credenziali da nostro browser fino alla pagina di login

Rlchiesta POST  
inserimento low  
e submit

La seconda ci dà un'idea di ciò che viene consegnato al web server quando schiacciamo un pulsante, come ad esempio submit per il cambio di sicurezza. Scegliamo low.

Immagine caricata

La terza slide ci mostra una richiesta POST che passa un'immagine di tipo jpg alla pagina di upload. Se il caricamento va a buon fine, dovrebbe uscire un avviso rosso come questo, contenente la parte di percorso che ci permette di accedere all'immagine.



## Inserimento file .php

Inserendo il nostro file .php dentro upload, possiamo lanciare un attacco Cross scripting. Dopo averlo caricato con successo, possiamo eseguire comandi di varia natura, ad esempio ls(file presenti) oppure pwd(percorso Web server).

