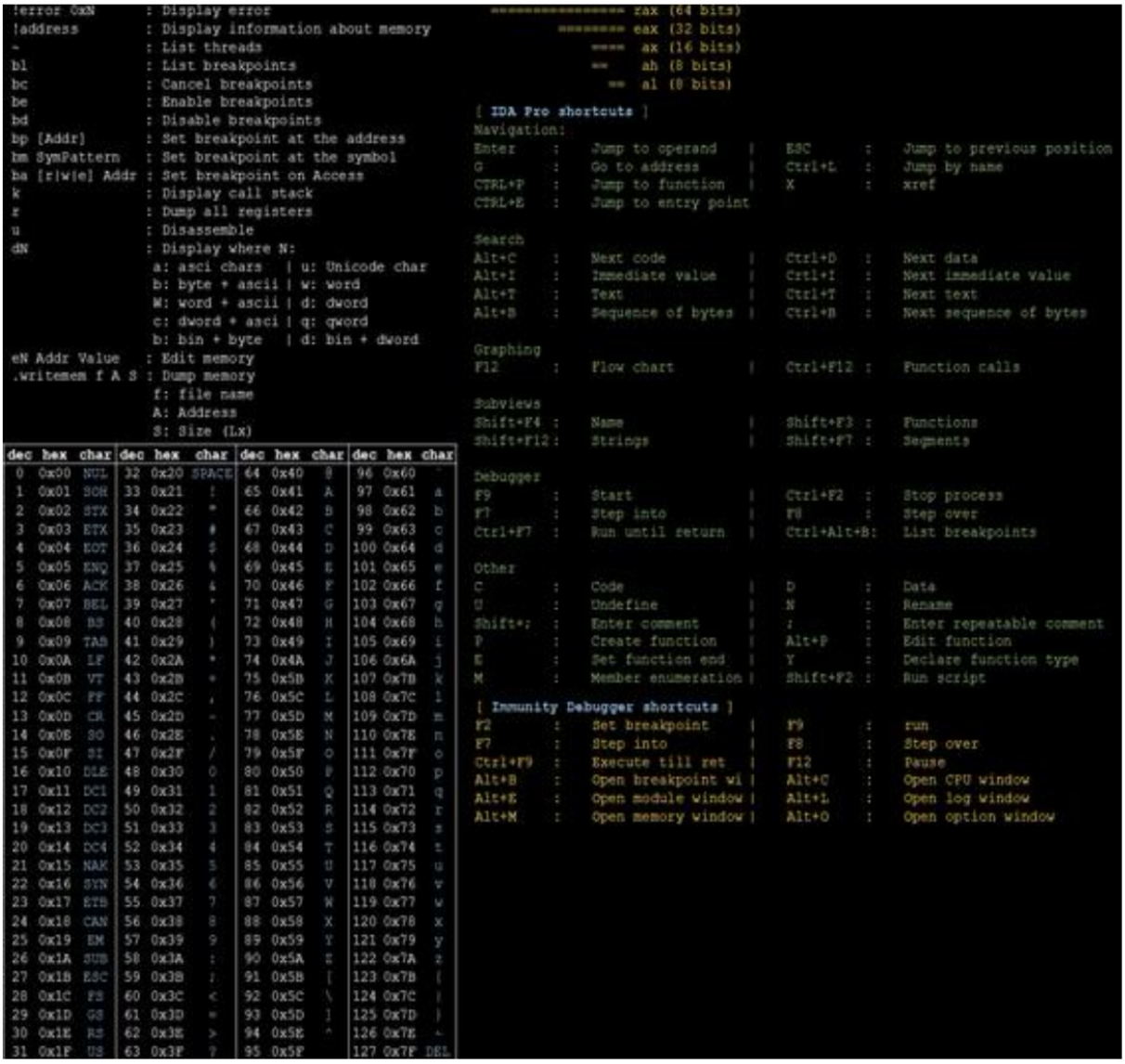
MODULO 5- REVERSING DE REDES Y PROTOCOLOS



```
!error 0xN        : Display error
!address          : Display information about memory
~                 : List threads
bl                : List breakpoints
bc                : Cancel breakpoints
be                : Enable breakpoints
bd                : Disable breakpoints
bp [Addr]         : Set breakpoint at the address
bm SymPattern     : Set breakpoint at the symbol
ba [r|w|e] Addr   : Set breakpoint on Access
k                 : Display call stack
r                 : Dump all registers
u                 : Disassemble
dN                : Display where N:
                    a: asci chars   | u: Unicode char
                    b: byte + ascii | w: word
                    W: word + ascii | d: dword
                    c: dword + asci | q: qword
                    b: bin + byte   | d: bin + dword
eN Addr Value     : Edit memory
.writemem f A S   : Dump memory
                    f: file name
                    A: Address
                    S: Size (Lx)
```

dec	hex	char	dec	hex	char	dec	hex	char	dec	hex	char
0	0x00	NUL	32	0x20	SPACE	64	0x40	@	96	0x60	`
1	0x01	SOH	33	0x21	!	65	0x41	A	97	0x61	a
2	0x02	STX	34	0x22	"	66	0x42	B	98	0x62	b
3	0x03	ETX	35	0x23	#	67	0x43	C	99	0x63	c
4	0x04	EOT	36	0x24	$	68	0x44	D	100	0x64	d
5	0x05	ENQ	37	0x25	%	69	0x45	E	101	0x65	e
6	0x06	ACK	38	0x26	&	70	0x46	F	102	0x66	f
7	0x07	BEL	39	0x27	'	71	0x47	G	103	0x67	g
8	0x08	BS	40	0x28	(	72	0x48	H	104	0x68	h
9	0x09	TAB	41	0x29	)	73	0x49	I	105	0x69	i
10	0x0A	LF	42	0x2A	*	74	0x4A	J	106	0x6A	j
11	0x0B	VT	43	0x2B	+	75	0x5B	K	107	0x7B	k
12	0x0C	FF	44	0x2C	,	76	0x5C	L	108	0x7C	l
13	0x0D	CR	45	0x2D	-	77	0x5D	M	109	0x7D	m
14	0x0E	SO	46	0x2E	.	78	0x5E	N	110	0x7E	n
15	0x0F	SI	47	0x2F	/	79	0x5F	O	111	0x7F	o
16	0x10	DLE	48	0x30	0	80	0x50	P	112	0x70	p
17	0x11	DC1	49	0x31	1	81	0x51	Q	113	0x71	q
18	0x12	DC2	50	0x32	2	82	0x52	R	114	0x72	r
19	0x13	DC3	51	0x33	3	83	0x53	S	115	0x73	s
20	0x14	DC4	52	0x34	4	84	0x54	T	116	0x74	t
21	0x15	NAK	53	0x35	5	85	0x55	U	117	0x75	u
22	0x16	SYN	54	0x36	6	86	0x56	V	118	0x76	v
23	0x17	ETB	55	0x37	7	87	0x57	W	119	0x77	w
24	0x18	CAN	56	0x38	8	88	0x58	X	120	0x78	x
25	0x19	EM	57	0x39	9	89	0x59	Y	121	0x79	y
26	0x1A	SUB	58	0x3A	:	90	0x5A	Z	122	0x7A	z
27	0x1B	ESC	59	0x3B	;	91	0x5B	[	123	0x7B	{
28	0x1C	FS	60	0x3C	<	92	0x5C	\	124	0x7C	|
29	0x1D	GS	61	0x3D	=	93	0x5D	]	125	0x7D	}
30	0x1E	RS	62	0x3E	>	94	0x5E	^	126	0x7E	~
31	0x1F	US	63	0x3F	?	95	0x5F	_	127	0x7F	DEL

```
================ rax (64 bits)
        ======== eax (32 bits)
            ====  ax (16 bits)
            ==    ah (8 bits)
              ==  al (0 bits)
```

```
[ IDA Pro shortcuts ]
Navigation:
Enter     :   Jump to operand    |   ESC      :   Jump to previous position
G         :   Go to address      |   Ctrl+L   :   Jump by name
CTRL+P    :   Jump to function   |   X        :   xref
CTRL+E    :   Jump to entry point

Search
Alt+C     :   Next code          |   Ctrl+D   :   Next data
Alt+I     :   Immediate value    |   Ctrl+I   :   Next immediate value
Alt+T     :   Text               |   Ctrl+T   :   Next text
Alt+B     :   Sequence of bytes  |   Ctrl+B   :   Next sequence of bytes

Graphing
F12       :   Flow chart         |   Ctrl+F12 :   Function calls

Subviews
Shift+F4  :   Name               |   Shift+F3 :   Functions
Shift+F12 :   Strings            |   Shift+F7 :   Segments

Debugger
F9        :   Start              |   Ctrl+F2  :   Stop process
F7        :   Step into          |   F8       :   Step over
Ctrl+F7   :   Run until return   |   Ctrl+Alt+B:  List breakpoints

Other
C         :   Code               |   D        :   Data
U         :   Undefine           |   N        :   Rename
Shift+;   :   Enter comment      |   ;        :   Enter repeatable comment
P         :   Create function    |   Alt+P    :   Edit function
E         :   Set function end   |   Y        :   Declare function type
M         :   Member enumeration |   Shift+F2 :   Run script

[ Immunity Debugger shortcuts ]
F2        :   Set breakpoint     |   F9       :   run
F7        :   Step into          |   F8       :   Step over
Ctrl+F9   :   Execute till ret   |   F12      :   Pause
Alt+B     :   Open breakpoint wi |   Alt+C    :   Open CPU window
Alt+E     :   Open module window |   Alt+L    :   Open log window
Alt+M     :   Open memory window |   Alt+O    :   Open option window
```

Máster en Análisis de Malware, Reversing y Bug Hunting

ENIIT INNOVA IT BUSINESS SCHOOL

excellence inNova Consultoría y Formación S.L.

UCAM Universidad Católica de Murcia

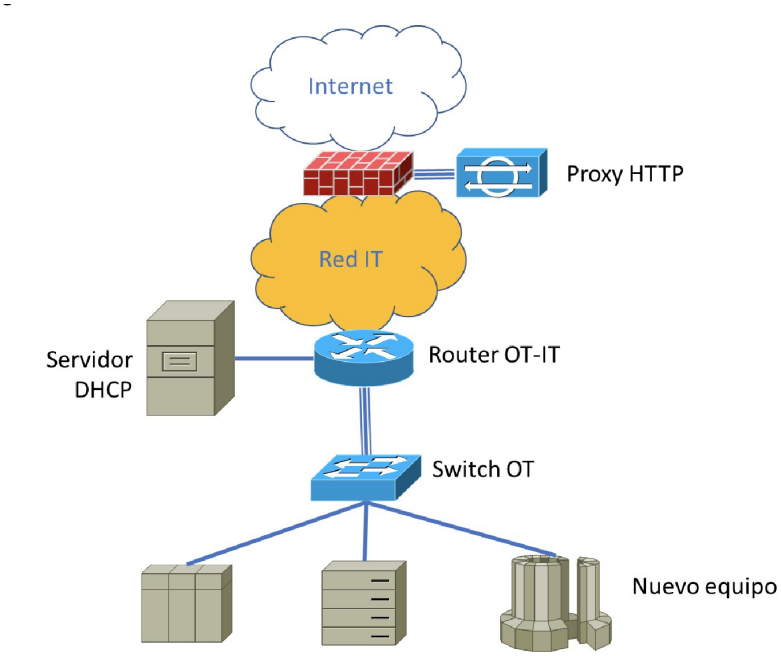
Campus Internacional CIBERSEGURIDAD

Ramon Gonzalez Gaztelupe

1.- Introducción

Se nos pide proponer el mecanismo o mecanismos de captura de tráfico que cumplan con todos los requisitos del cliente. Debemos realizar un análisis de cada una de las técnicas de captura de trafico y a partir del análisis proponer los que considere necesario
para este caso, explicando en detalle en qué consistirían (incluyendo la configuración necesaria en la estación de captura) y cómo se desplegarían.

Se nos proporciona este esquema de red:



Y los requisitos de cliente que nos especifica ACME S.A son los siguientes:

-Se desea capturar todo trafico entrante y saliente del nuevo equipo
-Impacto minimo en la red OT
-Solo se accede a internet a traves del proxy web
-La red OT obtiene la configuracion de red y Ipv4 del servidor DHCP
-Toda la infraestructura salvo el switch OT disponen de la capacidad port mirroring
-Tendremos una ventana de mantenimiento para aprovechar la instalacion y/o configuracion del sistema de monitoreo de trafico, una vez cerrada la ventana no podremos reconfigurar nada.

2.- Captura de trafico

Nombraremos las distintas formas de capturar el trafico para optar por la mas optima según las especificaciones del cliente ACME.

2.1- En el propio equipo

Si bien el sitio idóneo para realizar una captura de trafico eficiente de entrada y salida, este método también requiere de un acceso completo al sistema y que ademas el sistema operativo nos permita la instalación de herramientas externas tanto pa la captura como para el análisis.

En este caso desconocemos que tipo de sistema lleva instalado el nuevo nodo de la red OT, sin embargo si que sabemos que es un sistema operativo privativo el cual solo podremos configurar a través de un panel de configuración.

Por lo tanto queda descartada la captura en el propio equipo.

2.2- Nivel físico

Para realizar captura a nivel físico en la topología de red que nos presentan nos encontramos con varios inconvenientes.

El primero es que el switch OT en el que esta conectado el nuevo equipo carece de la capacidad de port mirroring, por lo tanto descartaremos esa posibilidad dentro las capturas físicas.

No podremos usar una sonda física en el enlace uplink del switch ya que no obtendremos los paquetes locales y ademas es un método para captura de mas de un equipo.

El problema de realizar port mirroring en el router es que dependiendo del tráfico que estemos monitoreando podemos llegar a colapsar nuestro Router provocando pérdida de paquetes o incluso que pare de funcionar completamente (oversubscription), hay que tener en cuenta que si por ejemplo estamos reenviando el tráfico de 3 puertos full-duplex de 100Mbps a un puerto de 100Mps full-duplex, éste puerto podría recibir 600Mbps, con lo que estaríamos sobrepasando su ancho de banda en 500Mbps, calculando las tramas que envía y recibe por puerto.

Conectar el nuevo equipo y el equipo de monitoreo de trafico a un Hub que a su vez este conectado al switch lo descartamos completamente ya que degrada el rendimiento Ethernet convirtiendo el enlace en semiduplex y ademas ya no se fabrican Hub´s.

Port Stealing tampoco lo usaremos por que tendríamos que estar inyectando tramas falsas continuamente para manipular la tabla de reenvío del switch.

2.3- Nivel de red

A nivel de red manejamos diferentes tipos de posible monitoreo de trafico, pero realmente no son una forma profesional de realizar una captura, mas bien podríamos usarlos como ataques tras detectar una intrusión en nuestra infraestructura.

- Redireccionamiento ICMP
- Suplantación DHCP
- ARP spoofing

2.4- Nivel de aplicación

Este método no es del todo concluyente en nuestro caso, si bien el nodo tendrá la capacidad de encapsular trafico a nivel de aplicación no tenemos datos suficientes sobre que monitorear salvo el trafico completo del nodo.

Una de las formas seria asignar desde el servidor DHCP una IP fija tanto al nuevo nodo como a la estación de captura (*el router trabaja en la capa 3 y utiliza la IP de los nodos para manejar los paquetes a diferencia de los switch que trabajan en la capa 2 y utilizan la dirección MAC*), agregar la estación directamente al router y modificar en ese router la DNAT para cambiar el puerto y ip destino de los paquetes y redirigirlos a nuestra estación.

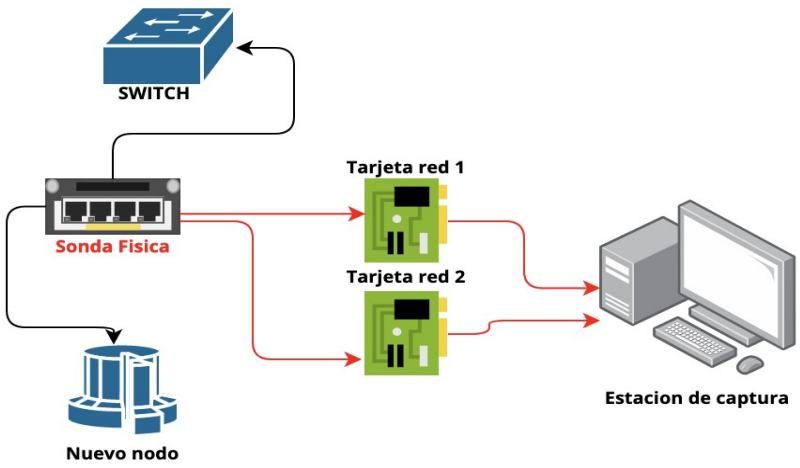
3.- Solución de Captura

Finalmente realizaremos la captura a nivel físico mediante un TAP no agregado para mayor flexibilidad y garantizar los paquetes de entrada y salida por puerto.

Para la instalación necesitaremos:

- TAP no agregado con 2 puertos, uno de entrada y otro para la tramas de envío
- Estación de captura con soporte para 2 tarjetas de red
- 2 tarjetas de red para la estación de captura
- 4 cables par trenzados
- 8 conectores RJ45

De tal forma que nos quedaría como el siguiente diagrama:



Las 2 tarjetas de red van integradas en la misma estación de captura.

La sonda física también requiere de batería o corriente.