Atizapán de Zaragoza Septiembre 17 del 2007

Instituto Tecnológico y de Estudios Superiores de Monterrey
Campus Estado de México
Departamento de Ciencias Computacionales

Proyecto de Segundo Parcial, Operativos II

Descripción del proyecto:

El proyecto se divide en dos partes. La parte teórica que cuenta con un valor del 40% y la parte práctica que cuenta con un valor del 60%. Fecha de entrega: día del segundo examen parcial. Para la parte teórica se requerirá lo siguiente:

Parte Teórica

El alumno investigará los conceptos relacionados con los sockets de tipo crudo o raw. Se investigará todo lo relacionado con su estructura, construcción diseño e implementación. Se realizará un reporte completo donde se explique cada uno de los conceptos mencionados anteriormente. Se pedirá al alumno que cite ejemplos y aplicaciones de los sockets tipo raw, así como éstos han impactado el uso y diseño de los sistemas operativos modernos. Se recomienda al alumno a investigar también lo que se está haciendo a la fecha, o state of the art en cuanto a la materia de raw sockets y sistemas operativos. Se recomienda consultar bases de datos de IEEE y ACM.

Requisitos Parte teórica:

- Investigación de papers de IEEE y ACM
- Reporte detallado de raw sockets como descrito arriba (mínimo 6 hojas)
- Referencias (mínimos: 5 libros, 15 referencias web, 2 contenidos de biblioteca digital).
- Aplicaciones en reporte.

Parte Práctica

Para la parte práctica se le pedirá al alumno que desarrolle e implemente un programa de ***inyección de paquetes.*** El programa deberá ser capaz como mínimo inyectar paquetes de tipo TCP o ICMP. Se recomienda al alumno utilizar un sniffer como Ethereal para determinar si su programa inyecta correctamente las tramas al ambiente de red.

Se presentará a continuación una sintaxis recomendada para la elaboración del programa. Quedará a decisión del alumno en utilizar dicha sintaxis o elaborar una que el considere conveniente.

Sintaxis para programa inyector de paquetes:

paquete -tcp -D <direccion IP destino> -x <puerto destino>

[-v] [-a ack-number] [-P payload-file] [-s sequence-number]
[-S source-IP-address] [-T IP-TTL] [-y source-port]

paquete -icmp -D <direccion IP destino> -x <puerto destino>

[-v] [-c ICMP-code] [-S source-IP-address]

Aclaraciones:

- [-T IP-TTL] especifica el valor del campo time to live (TTL) dentro del encabezado IP.
- [-a ack-number] especifica el numero de ACK dentro del campo de encabezado TCP
- [-s sequence-number] especifica el numero de secuencia dentro del encabezado TCP
- [ICMP-type] especifica el ICMP-code dentro del encabezado ICMP
- [-c ICMP-type] Specify the ICMP-code within the ICMP header.
- [-v] Despliega todo lo inyectado en el paquete.

Ejemplo:

paquete -tcp -v -S 192.168.1.1 -D 192.168.2.2 -a 1234 -s8989 -x 9898 -y 22 -P ivan

Valores paquete inyectado:

[IP] 192.168.1.1 > 192.168.2.2
[IP ID] 13566
[IP Proto] TCP (6)
[IP TTL] 255
[IP TOS] 00
[IP Frag offset] 0000
[IP Frag flags]
[TCP Ports] 9898 > 22
[TCP Flags] SYN
[TCP Urgent Pointer] 0
[TCP Window Size] 4096
[TCP Seq number] 898

Se escribieron 114 bytes de un paquete TCP.
FECHA DE ENTREGA: Día del examen de segundo parcial (codigo fuente, reporte, pruebas de funcionamiento capturadas de ethereal).