Práctica 3: Análisis de Tráfico

Javier Ramos de Santiago
José Luis García Dorado

Introducción

- ¿Qué es el análisis de tráfico?
  - Conjunto de técnicas que dada una traza (captura de paquetes) o el tráfico de una interfaz de red, obtiene conclusiones y datos a partir de los datos contenidos en los paquetes.
- ¿Por qué es importante?
  - Monitorización
  - Seguridad
  - Dimensionado de redes
  - Detección de problemas

Conceptos Previos

- ¿Qué es un Flujo?
  - Se define flujo como el conjunto de paquetes que tienen la misma quíntupla (ip origen, ip destino, puerto origen, puerto destino y protocolo) en un intervalo de tiempo dado
- ¿Qué es una sesión?
  - Se define sesión como la agrupación de un flujo entrante y otro saliente en un intervalo de tiempo

Wireshark

- ¿Qué es?
  - Analizador de tráfico que permite tanto capturar de una interfaz como abrir una traza capturada previamente con Wireshark o tcpdump
- ¿Qué hay que conocer?
  - Filtros de captura y visualización
  - Análisis y estadísticas:
    - Summary: estadísticas generales
    - IOGraphs: gráficas (tasa de bytes, paquetes, tamaño, etc)
    - Flow Graph: Gŕafico por sentidos
    - TCP Stream Graphs: tasa de transferencia de un flujo TCP

Libpcap(I)

- ¿Qué es?
  - Librería C que permite la captura y procesado de paquetes así como el almacenamiento en el formato propietario pcap
- ¿Cómo se usa?
  - Programar en C usando funciones pcap
  - Compilar con -lpcap

Libpcap (II)

- ¿Qué funciones tenemos?
  - pcap_t * pcap_open_live(const char *device, int snaplen, int promisc, int to_ms, char *errbuf)
  - pcap_t * pcap_open_offline (const char *fname, char *errbuf)
  - const u_char * pcap_next(pcap_t *p, struct pcap_pkthdr *h)
  - void pcap_close(pcap_t *p)

Estructura básica de un programa pcap

```
pcap_t *handle
char errbuf[PCAP_ERRBUF_SIZE]
struct pcap_pkthdr header;
handle=pcap_open_offline(“traza.pcap”,errbuf);
if(handle==NULL)
{
    fprintf(stderr,”No se puede abrir traza.pcap %s\n”,errbuf);
    return -1;
}
while((packet=pcap_next(handle,&header))!=NULL)
//procesar paquete
pcap_close(handle);
```

¿Cómo empezar?

- Leer y entender el enunciado de la práctica.
- Comprender qué hacen las funciones pcap que se explican en el enunciado. Para ello mirar las descripciones y ejemplo.
- Mirar y entender bien las cabeceras de los protocolos Ethernet, IP y UDP/TCP. Se observar el orden y para qué sirve cada campo.
- Planificar cómo se van a partir los paquetes para obtener los campos. (Sugerencia: memcpy y un puntero de lectura sizeRead)

Ejercicios

- Para la realización de la practica se deben completar los archivos analisis_trafico.c y functions.c. Se deben LEER TODOS los comentarios que contiene el código e implementar lo que se indica en ellos.
- Abre/lee/cierra un fichero de captura (2 puntos).
- Abre/recorre/cierra un fichero de captura aplicándole filtros BPF (2 puntos)
- Análisis de protocolos a nivel de paquete (6 puntos)

Ejercicios(II)

- Además del código se debe entregar una memoria BREVE respondiendo a las preguntas del enunciado y razonando las respuestas que sean necesarias.
- Analizar trafico no es solo «picar» código también hay que pensar y analizar los resultados.

Funciones útiles

- memcpy
- qsort: Ordena vectores y datos en C. Útil para obtener puertos mas populares (Top 5 por ejemplo). Ver ejemplos en Internet y man.
- ntohs