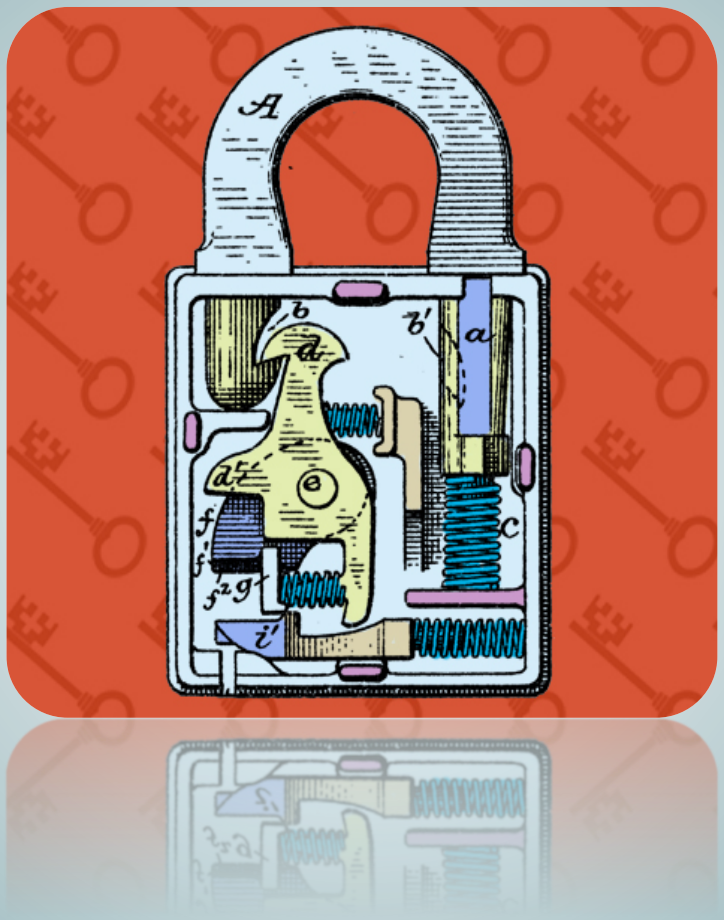
Garantía y Seguridad en Sistemas y Redes

Práctica 5. Denegación de Servicio en Apache



Esteban Stafford

Departamento de Ingeniería
Informática y Electrónica

Práctica 5

Denegación de servicio en apache

1. Introducción

2. Objetivos

Los objetivos fundamentales que persigue esta práctica son los siguientes:

- Conocer de primera mano el comportamiento de un ataque de denegación de servicio.
- Explorar maneras de paliar los efectos de este tipo de ataques.

3. Material Necesario

En esta práctica se realizan una serie de ataques de denegación de servicio al servidor apache que tenemos instalado en la máquina virtual. La herramienta que usaremos para los ataques es Switchblade. Ésta esta listada en el Open Web Application Security Project (OWASP), que es una entidad dedicada a la mejora de la seguridad de los servicios WEB. El programa ya compilado se puede descargar de `http://aulaserver/http_dos_cli`. A continuación se muestra un estracto de la ayuda con las opciones que se usarán en la práctica.

```
HTTP load tester for slow headers and slow POST attacks.
  Version: 4.0
  URL: http://code.google.com/p/owasp-dos-http-post/
Usage:
  --host <dns_name_of_webserver>
      Web server to connect to (just DNS, no URI). Defaults to localhost.
  --slow-headers
      Run slow-headers attack.
  --connections <num>
      Number of connections to spawn.
  --rate <num>
      Number of connections to create per second. If set to 1000 or greater,
      will create connections as fast as possible.
  --timeout <num>
      Timeout, in seconds, between each write of header data or POST data.
      Defaults to 100 seconds, and may include fractional seconds, e.g. 1.5
      for one and a half seconds.
  --report-interval <num>
      Interval (in seconds) between each report of statistics. This defaults
```

```
    to 1.0.
--log-connection <num>
    Print out diagnostic information about a single connection: all data
    transferred will be printed out. The number specified is the number of
    the created connection, starting at one. For example, if 1 is specified
    the first connection created will have it's information printed.
```

4. Desarrollo

4.1. Estudio de la herramienta de ataque

La máquina de ataque será la máquina física, por lo que habrá que descargar el programa `http_dos_cli` a nuestra cuenta en esta máquina. La herramienta permite hacer varios tipos de ataque a servidores WEB. Por defecto utiliza un ataque de encabezados lentos. Prueba con diferentes parámetros de `timeout`, `connections` y `rate`, y familiarizate con el efecto de cada uno. Al principio es recomendable empezar con una sola conexion. Para ver el el comportamiento de ésta puedes indicarlo con la opción `log-connection`. Aunque también puedes usar un analizador de tráfico como `tcpdump`, `ngrep` o wireshark.

¿De qué manera consigue la herramienta tener la conexión HTTP abierta de manera indefinida? ¿Es correcta la cabecera HTTP?

4.2. Monitorización de apache

Antes de realizar un ataque conviene ser capaz de monitorizar el estado interno de apache. Esto se puede hacer visitando la página `/server-status`. Por defecto apache no permite que se visite esta página desde fuera de la máquina local. Para mayor comodidad modifica la configuración del módulo `status` para que se pueda visitar desde la máquina anfitrión.

Observa el contenido de esta página. En particular nos interesa el estado de los differentes threads de trabajo de apache.

```
1 requests currently being processed, 74 idle workers

.........................________________________________________
____W_______________________________.............................
......................

Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process
```

Haz varias pruebas con `http_dos_cli`, con 10 o 15 conexiones y observa como se van ocupando los threads y en qué estado se ponen. Prueba también con clientes normales, como firefox o `wget`. Para hacer muchas conexiones, puedes usar bucles en `bash`.

¿En qué se diferencia el efecto de las conexiones de `http_dos_cli` de las de otros clientes no maliciosos?

4.3. Al ataque

Ahora prueba varias configuraciones de `http_dos_cli` para conseguir que apache no pueda responder a conexiones legítimas. Es lo más normal es que apache no responda ni siquiera a peticiones de `/server-status`. Entonces se puede parar el ataque y refrescar.

¿Cual es la carga de la máquina virtual durante el ataque? ¿Y la ocupación de los enlaces de red?

4.4. Defensa de apache

Se ha visto que un ataque de este tipo hace que apache deje de responder. Activa y configura el módulo `reqtimeout` de apache para reducir el efecto de este tipo de ataques.