

## SWAP: EJERCICIOS TEMA 5

Mónica Jiménez Montañés

### Ejercicio 5.1: Buscar información sobre cómo calcular el número de conexiones por segundo.

Si tenemos un servidor Nginx, podemos calcular fácilmente gracias a un módulo de Nginx el número de conexiones por segundo. En primer lugar tenemos que activar la recopilación de estadísticas en el fichero **nginx.conf** añadiendo la sentencia **stub\_status** on en location **/nginx\_status** y hacerla accesible sólo para la máquina del administrador.

Entonces tendremos disponible en el navegador agregando a la dirección de la página cuyo estado queremos comprobar **"/nginx\_status"** el número de conexiones abiertas, de conexiones aceptadas, manejadas, y peticiones manejadas.

Si dividimos el número de peticiones manejadas entre el número de conexiones manejadas, tendremos el número de conexiones abiertas por segundo.

Otra forma de comprobar el número de conexiones es con **netstat | grep -c http**, es decir, estamos mostrando la salida correspondiente a las conexiones de red, tablas de enrutamiento y estadísticas, y buscando (y contando) las conexiones relativas al protocolo HTTP.

### Ejercicio 5.2: Instalar wireshark y observar cómo fluye el tráfico de red en uno de los servidores web mientras se le hacen peticiones HTTP.

Ya había usado wireshark en la asignatura de Redes bajo Ubuntu, pero como esta vez me ha dado problemas con la máquina virtual, he decidido instalar la versión para Windows.

Es muy fácil de usar, y podemos comprobar el tráfico de manera evidente.

Capturing from Wi-Fi 2						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
Apply a display filter ... <Ctrl-/>						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.198	216.58.211.227	SSL	55	Continuation Data
2	0.097569	216.58.211.227	192.168.0.198	TCP	66	443 → 50259 [ACK] Seq=1 Ack=2 Win=361 Len=0 SLE=1 SRE=2
3	2.242549	31.13.83.8	192.168.0.198	TLSv1.2	387	Application Data
4	2.244273	31.13.83.8	192.168.0.198	TLSv1.2	109	Application Data
5	2.244321	192.168.0.198	31.13.83.8	TCP	54	50230 → 443 [ACK] Seq=1 Ack=389 Win=255 Len=0
6	2.270341	192.168.0.198	31.13.83.36	TLSv1.2	310	Application Data
7	2.270414	192.168.0.198	31.13.83.36	TLSv1.2	100	Application Data
8	2.270444	192.168.0.198	31.13.83.36	TLSv1.2	464	Application Data
9	2.321389	192.168.0.198	31.13.83.36	TLSv1.2	164	Application Data
10	2.321462	192.168.0.198	31.13.83.36	TLSv1.2	547	Application Data
11	2.333786	31.13.83.36	192.168.0.198	TLSv1.2	96	Application Data
12	2.334737	31.13.83.36	192.168.0.198	TLSv1.2	100	Application Data
# Frame 1: 55 bytes on wire (440 bits), 55 bytes captured (440 bits) on interface 0 Interface id: 0 (\Device\NPF_{8FE93F93-0355-4B98-B8FE-B6E462B6AB93}) Encapsulation type: Ethernet (1) Arrival Time: Jun 8, 2016 17:35:15.142627000 Hora de verano romance [Time shift for this packet: 0.000000000 seconds] Epoch Time: 1465400115.142627000 seconds [Time delta from previous captured frame: 0.000000000 seconds] [Time delta from previous displayed frame: 0.000000000 seconds] [Time since reference or first frame: 0.000000000 seconds] Frame Number: 1 Frame Length: 55 bytes (440 bits) Capture Length: 55 bytes (440 bits) [Frame is marked: False]						
0000	ec 23 3d 70 2c 4e 00 c0	ca 10 3d 85 08 00 45 00	.#=p,N.. ..=...E.			
0010	00 29 05 5f 40 00 80 06	87 e3 c0 a8 00 c6 d8 3a	.)_.@... .....			
0020	d3 e3 c4 53 01 bb af da	41 0a a6 0e 93 b2 50 10	...S..... A.....P.			
0030	00 ff 50 93 00 00 00		..P....			

Capturing from Wi-Fi 2						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
Apply a display filter ... <Ctrl-/>						
No.	Time	Source	Destination	Protocol	Length	Info
343	105.398991	131.253.61.80	192.168.0.198	TCP	54	443 → 50462 [ACK] Seq=11617 Ack=7742 Win=65536 Len=0
344	108.084961	173.255.112.173	192.168.0.198	TLSv1.2	105	Application Data
345	108.128421	192.168.0.198	173.255.112.173	TCP	54	50285 → 443 [ACK] Seq=1 Ack=205 Win=254 Len=0
346	109.201405	216.58.211.238	192.168.0.198	TLSv1.2	117	Application Data
347	109.201638	192.168.0.198	216.58.211.238	TCP	54	50403 → 443 [FIN, ACK] Seq=2 Ack=64 Win=258 Len=0
348	109.201899	216.58.211.238	192.168.0.198	TCP	54	443 → 50403 [FIN, ACK] Seq=64 Ack=2 Win=1456 Len=0
349	109.201928	192.168.0.198	216.58.211.238	TCP	54	50403 → 443 [ACK] Seq=3 Ack=65 Win=258 Len=0
350	109.261034	216.58.211.238	192.168.0.198	TCP	54	443 → 50403 [ACK] Seq=65 Ack=3 Win=1456 Len=0
351	109.864245	fe80::f1aa:d50d:58d...	ff02::1:2	DHCPv6	148	Solicit XID: 0xa59550 CID: 0001000119f778ff201a062c1b2f
352	110.678546	192.168.0.198	158.85.224.178	TLSv1.2	92	Application Data
353	110.847209	158.85.224.178	192.168.0.198	TLSv1.2	99	Application Data
354	110.912723	192.168.0.198	158.85.224.178	TCP	54	50283 → 443 [ACK] Seq=267 Ack=316 Win=257 Len=0
355	110.953557	162.125.17.131	192.168.0.198	TLSv1.2	388	Application Data
356	110.955951	192.168.0.198	162.125.17.131	TLSv1.2	477	Application Data
357	111.153979	162.125.17.131	192.168.0.198	TCP	54	443 → 50250 [ACK] Seq=1003 Ack=1270 Win=83 Len=0
358	113.086588	HuaweiTe_70:2c:4e	Alfa_10:3d:85	ARP	42	Who has 192.168.0.198? Tell 192.168.0.1
359	113.086615	Alfa_10:3d:85	HuaweiTe_70:2c:4e	ARP	42	192.168.0.198 is at 00:c0:ca:10:3d:85
360	124.370576	192.168.0.198	74.125.206.188	TCP	55	[TCP Keep-Alive] 50227 → 443 [ACK] Seq=1 Ack=1 Win=254 Len=1
361	124.451140	74.125.206.188	192.168.0.198	TCP	66	[TCP Keep-Alive ACK] 443 → 50227 [ACK] Seq=1 Ack=2 Win=363 Len=0 SLE=1 SRE=2

### Ejercicio 5.3: Buscar información sobre herramientas para monitorizar las prestaciones de un servidor.

Podemos usar usando las más usadas en Linux:

- top
- Vmstat
- Netstat

**Con top** podemos tener una vista dinámica de un sistema en funcionamiento. Puede mostrar un resumen y una lista de procesos manejados por el kernel Linux. Tiene una limitada interfaz interactiva y una interfaz más amplia para la configuración personal.

**Vmstat** muestra información sobre procesos, memoria, discos, E/S y actividad de la CPU.

**Netstat** muestra conexiones de red, tablas de enrutamiento, las máscaras de las diferentes conexiones y estadísticas de las diferentes interfaces.

**Munin.** Monitor de servidores para Linux cuya principal virtud es su extensibilidad mediante plugins. También posee una estructura basada en nodos que nos permite recopilar información de varios servidores de forma simultánea.

En el caso de Windows 2008 también tenemos desde “Inicio/Herramientas Administrativas” podemos acceder al Monitor de Confiabilidad y Rendimiento.

**Nagios.** Monitor creado para Linux capaz de monitorizar los servicios de red, los recursos hardware y posee diversas características de personalización como el acceso remoto, el servicio de notificaciones, el chequeo de servicios paralelizados o la visualización en tiempo real del estado de la red.

**Ganglia.** Sistema de monitoreo de computadores de alto rendimiento como clusters. Gracias a sus algoritmos consume bastante poco por nodo y utiliza una alta concurrencia a la hora de obtener toda la información tanto a nivel de servicios como hardware de nuestros servidores.