

PRACTICA 3

Pareja:06

Daniel Garduño Hernandez

Mario Garcia Roque

Detalles sobre el programa:

Aunque pedíais que el programa no fuese general y que respondiese a los requisitos en varios programas debido a que era más sencillo, hemos hecho un programa. ¿El motivo? Básicamente lo hemos entendido al revés, por un despiste. Pero el programa funciona bien, y en cada ejecución crea 3 ficheros que dada una traza cambiarán si se modifican los filtros.

Recomendamos que si se va a ejecutar el programa se redireccione el tráfico a un archivo, es decir, añadir “> 'archivo' ” al final del programa. Lo mismo si se desea ejecutar el script ya que este contiene casi 10 llamadas al programa.

El script llama repetidamente al programa cambiando los datos que se nos piden: puerto, máscaras, etc.

EJERCICIO 1:

IP:54333, 99.43 %
NOIP:314, 0.57 %

UDP:7606, 14.00 %
TCP:46449, 85.49 %
otros:278, 0.51 %

EJERCICIO 2:

Como se puede ver está implementado el caso IP. La implementación elegida para almacenar los datos de IP ha sido la de una lista de punteros ya que si no ocupa demasiado tamaño de memoria (si no recordamos mal creemos que puede llegar a ocupar toda la memoria de un sistema de 32 bits). Para los puertos también usamos esta implementación.

IP más usada:

referencias

1º:198.183.101.139,	count:34986
2º:136.94.42.18,	count:15454
3º:198.56.233.139,	count:11462
4º:129.100.116.235,	count:6423
5º:129.227.242.229,	count:6423

tam:

1º:198.183.101.139,	tam:50345143
2º:136.94.42.18,	tam:23098463
3º:136.94.42.21,	tam:6917980
4º:136.94.42.21,	tam:6917980
5º:136.94.42.21,	tam:6917980

Puerto más usado:

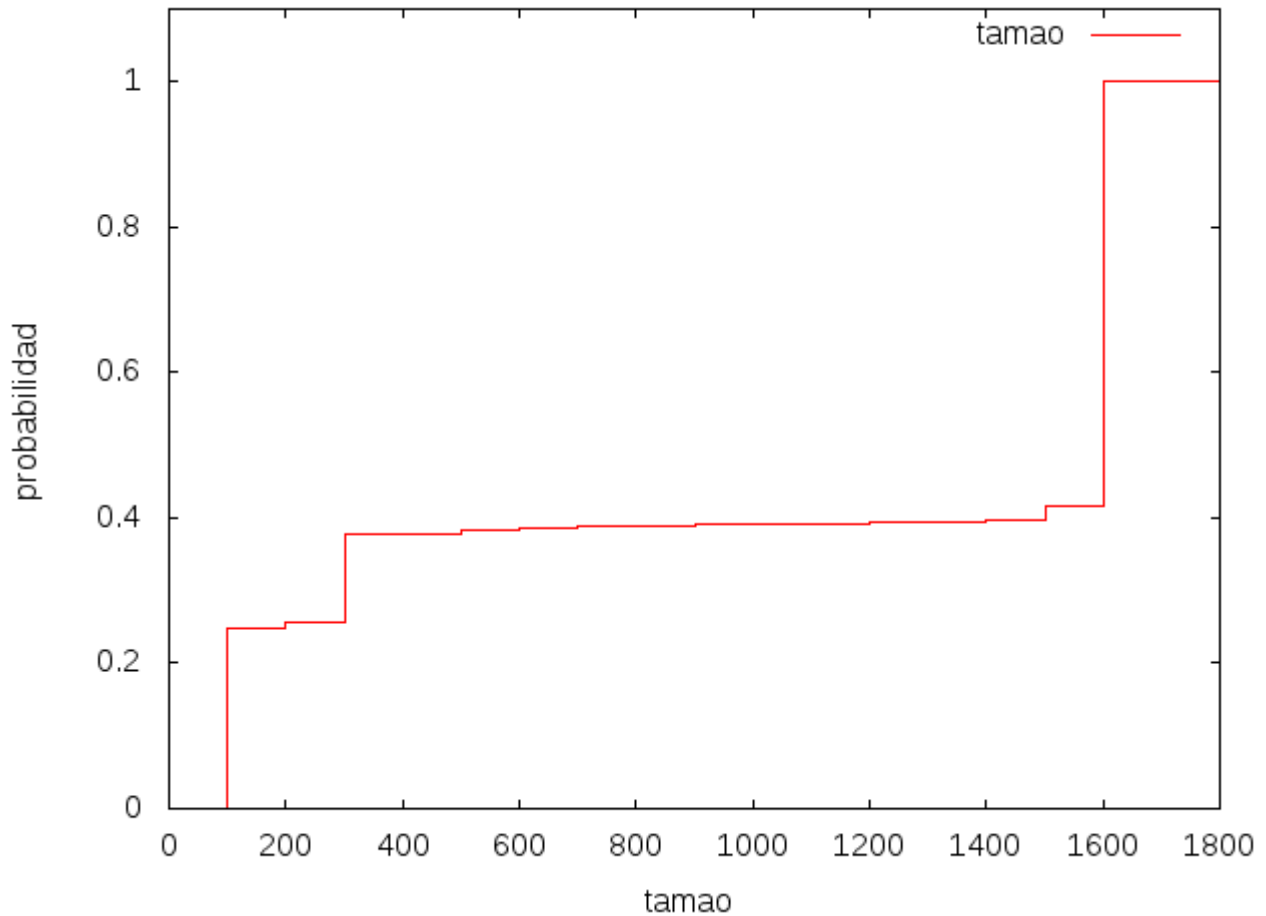
referencias

1º:80,	count:46054
2º:55934,	count:6909
3º:180,	count:6424
4º:36969,	count:6423
5º:55860,	count:5410

tam:

1º:80,	tam:51130785
2º:55934,	tam:8324498
3º:55860,	tam:6505361
4º:55860,	tam:6505361
5º:55860,	tam:6505361

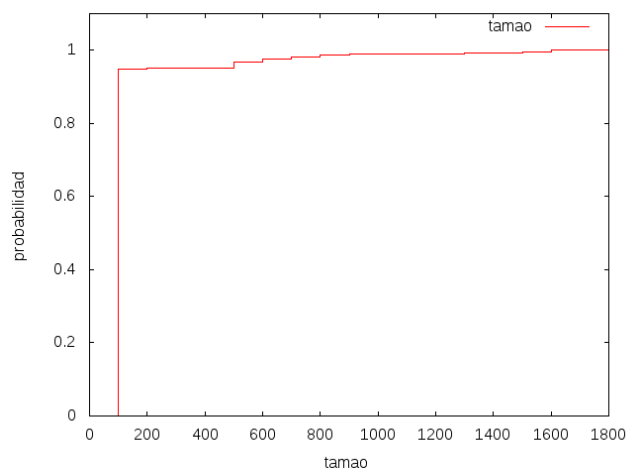
EJERCICIO 3:



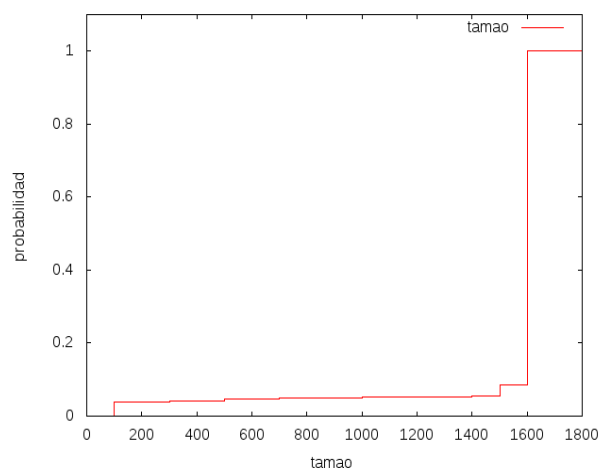
Como se puede observar en la gráfica la mayoría del tráfico está por encima de los 1500 bytes de tamaño y además este tráfico en su mayoría es de tipo TCP. Por tanto, como podemos ver en los porcentajes del ejercicio anterior, tiene sentido ya que la mayoría del tráfico es de este tipo.

EJERCICIO 4

Tráfico de bajada:



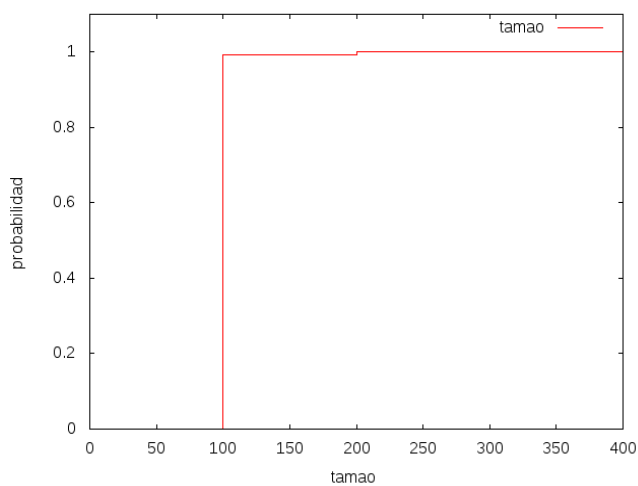
Tráfico subida:



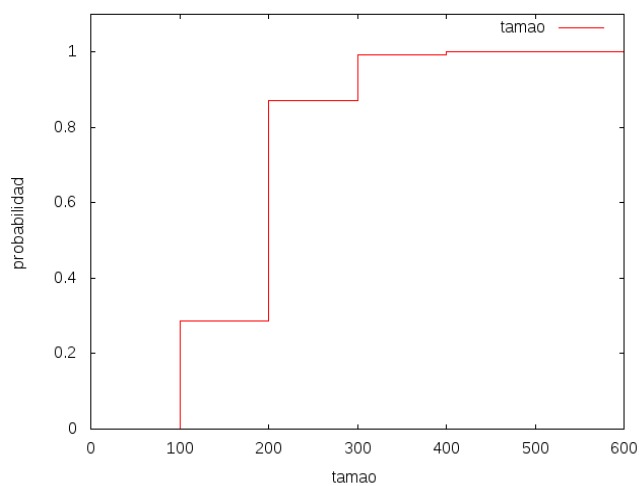
No entendemos mucho qué hay que comentar sobre estas gráficas. Tenemos un tráfico de subida HTTP y un tráfico de bajada. Cómo tampoco sabemos en qué se basa el generador de la traza para crear tráfico no queda muy claro qué explicar. Quizá resulta llamativo que el de subida mande en general mucha información, lo cual quizá se deba a que se ha querido simular una red con servidores y por eso el de subida sea más alto.

EJERCICIO 5:

Tráfico bajada:



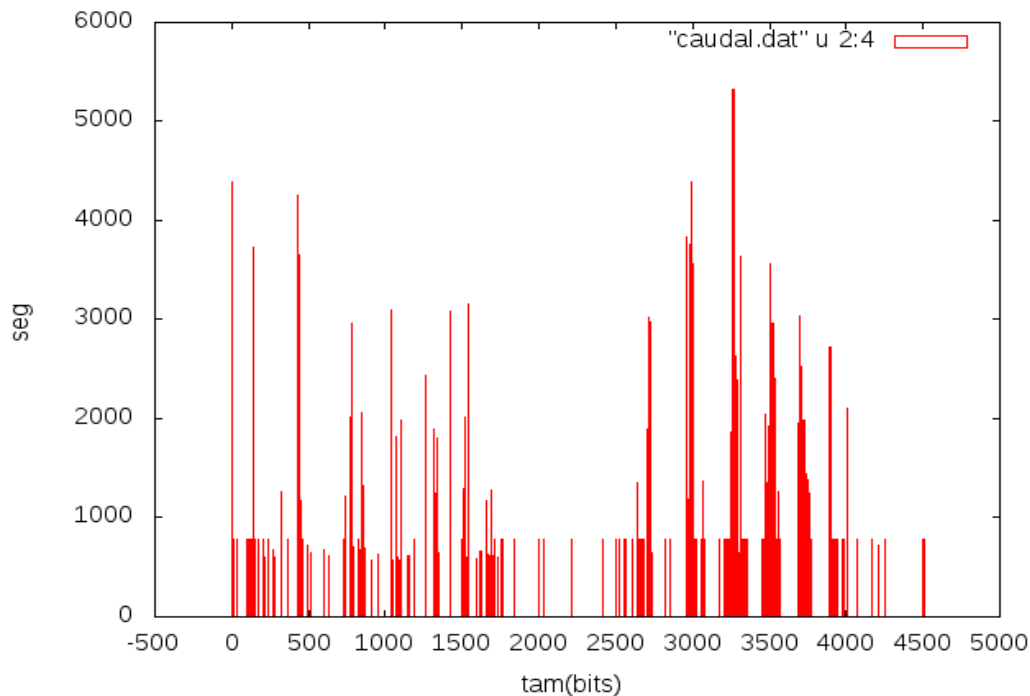
Tráfico subida:



Similar a lo que hemos explicado en el apartado anterior.

EJERCICIO 6:

No existe tráfico con la MAC origen que tenemos, por lo que no pondremos esta gráfica. Un detalle: la etiqueta del eje x debería estar en y.

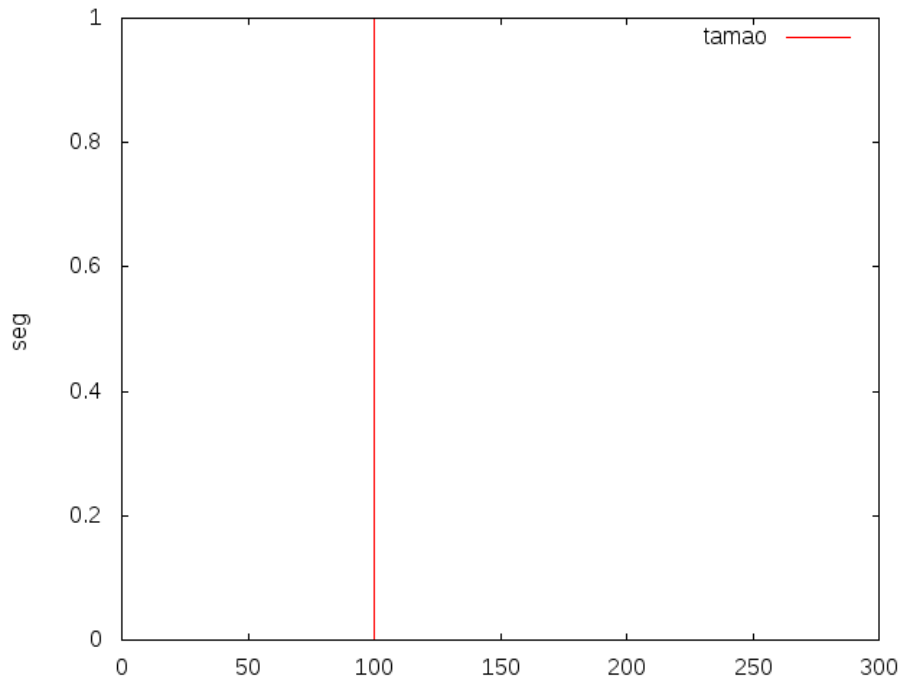


Esta gráfica ya tiene un sentido más importante. Es útil por varios motivos, quizá el más importante sea que nos indica el uso que damos en general a la red, pudiendo calcular a partir de estos datos dónde debemos situar estadísticamente el máximo ancho de banda para un usuario, aunque en realidad este estudio tiene más sentido para una dirección IP. Pero digamos que si tienes muchos usuarios y pocos recursos entonces esta gráfica te puede resultar útil para saber qué tráfico usa cada usuario y actuar en consecuencia.

También nos muestra otros detalles, como los periodos de actividad de un cierto usuario, aunque otra vez, sería más interesante saber el de una IP para saber el periodo de actividad de un cierto usuario de la red, puede ser interesante.

EJERCICIO 7:

El motivo para que la gráfica salga con un intervalo tan raro es que hemos creado nuestro propio creador de ECDF, que coge intervalos de 100. Por tanto, a pesar de que los datos se concentran en su mayoría en los decimales de 0, nuestra gráfica no lo representa bien.



EJERCICIO 8:

Ocurre lo mismo que en el caso anterior.

