



Welcome to RetroNet

jueves, noviembre 28 2002 @ 02:30 ART

Secciones

[Home](#)
[Herramientas](#) (32/0)
[Recursos](#) (17/0)
[Documentos](#) (73/0)
[Actualidad](#) (111/0)
[Exploits](#) (40/0)
[Bugs](#) (126/0)
[criptografía](#) (12/0)
[Hardware](#) (11/0)
[Privacidad](#) (29/0)
[Virus](#) (39/0)
[Incidentes](#) (31/0)
[Microsoft](#) (81/0)
[Linux](#) (33/0)
[Miscelaneas](#) (28/0)
[Debian](#) (11/0)
[Opinion](#) (26/0)

User Functions

Username:

Password:

Don't have an account yet? Sign up as a [New User](#)

Eventos

There are no upcoming events

Visitantes

[Anonymous](#)

[Anonymous](#)

[Anonymous](#)

Anonimos: 3

Planeta Linux

- [DHCP: De cómo configurar la red sin moverse](#)
- [Cuando Red Hat se viste de Windows](#)
- [Red Hat 7.3: Primeras impresiones](#)
- [KDE 3.0: Has recorrido un largo camino](#)
- [Viendo videos DivX en Linux](#)
- [Lindows 2.0](#)
- [FreeBSD 4.7 Full \(4 CDs\)](#)
- [FreeBSD 4.7 Install](#)
- [Red Hat 8.0 \(Psyche\) + Docs + Fuentes \(6 CDs\)](#)
- [Red Hat 8.0 \(Psyche\) + Fuentes \(5 CDs\)](#)

Vivalinux!

GeekLog can not reach the supplied RDF file at 2002-11-28 13:57:04. Please double check the URL provided. Make sure your url is correctly entered and it begins with http://. GeekLog will try in one hour to fetch the file again.

Ciberlinea

Control de ancho de banda con CBQ-INIT

Este script sirve para simplificar la disposición y la gerenciamiento del control de tráfico de manera relativamente simple usando un sistema basado en CBQ en Linux. El acceso a las características avanzadas de red del núcleo de Linux es proporcionado por la utilidades "ip" y "tc" del paquete de iproute2 de A. Kuznetsov, disponible en <ftp://ftp.inr.ac.ru/ip-routing>.

Dado que estas utilidades sirven específicamente para traducir deseos del usuario a comandos de RTNETLINK, su interfaz es algo espartana, intolerante y requiere la escritura de largas cadenas de comandos con parametros. Esta tediosa tarea de escritura de comandos es lo que este script procura reducir:).

El manejo de las facilidades avanzadas de una red en Linux es bastante flexibles y este script trae algunas de estas características a los usuarios de Linux de corazón no tan duro. Por supuesto, hay una relación inversamente proporcional entre la simplicidad y la flexibilidad y usted pueden pensar que la flexibilidad sufrió demasiado en favor de la simplicidad y que no cubre todas sus necesidades, entonces es hora de hacer frente al interfaz "ip" y "tc".

Si usted quiere solamente que `cbq.init` traduzca sus configuración a comandos "tc", use el parametro "compile" esto, generara los comandos "tc" requeridos para construir su configuración. Considere que "compile" no comprueba si los comandos "tc" son correctos a, se utiliza esto (en ciertos lugares) solamente cuando el comando "start nocache" es utilizado, el que es también útil al crear la configuración para comprobar si es totalmente válido. Todos los parámetros de CBQ son válidos para los interfaces Ethernet solamente, el script fue probado en varias versiones del núcleo de Linux de las series 2,1 a 2,4.

¿Cómo se configura CBQ-init?

- [Certificados gratuitos de C&A para la firma digital y el cifrado con SignStation](#)
- [La contaminación de los Ordenadores](#)
- [Un estudio encargado por Microsoft da por perdida la batalla contra la pirateria](#)
- [La electricidad del futuro no necesitará cables y se enviará vía satélite](#)
- [Supervirus: Estados Unidos Vs Irak](#)
- [Manu Chao: "Si un chaval no se baja tu disco en Internet, es que es gilipollas"](#)
- [Un científico sueco asegura que los ordenadores portátiles pueden quemar la piel](#)
- [Un año más de Congreso de HispaLinux](#)
- [Entrenamiento en creación de aplicaciones web y aplicaciones de escritorio.](#)
- [Software para GNU/Linux :: #002](#)
- [...](#)

Cada clase de tráfico se debe describir con un archivo en \$CBQ_PATH en el directorio (/etc/sysconfig/cbq por defecto), un archivo por clase. Los nombres del archivo de la configuración deben obedecer un formato obligatorio: cbq-<clsid>.<name> donde <clsid> es un numero hexadecimal de dos bytes dentro del rango <0002-FFFF> (que de hecho es una identificación de la clase de CBQ) y <name> es el nombre de la clase, cualquier cosa que le ayude a distinguir el archivo de configuración. Para una cantidad pequeña de clases que es posiblemente la mayoría de los casos deje que el <clsid> refleje el ancho de banda de la clase.

Parametros de dispositivo

Parametro:

DEVICE=<ifname>,<bandwidth>[,<weight>]

Ejemplo:

DEVICE=eth0,10Mbit,1Mbit

Descripcion:

1. <ifname>: es el nombre del interfaz de la que usted desea controlar el tráfico, e.g. eth0.
2. <ancho de banda>: es el ancho de banda física del dispositivo, e.g. para Ethernet 10Mbit o 100Mbit, para el arcnet 2Mbit.
3. <peso>: es el parametro de prioridad que debe ser proporcional al ancho de banda. Por lo general: <peso> = <ancho de banda>/10.

Cuando usted tiene más de una clase en un interfaz, basta con especificar ancho de banda y peso solamente una vez, por lo tanto en otros archivos usted necesita solamente especificar DEVICE=ifname.

Parametros de Clases

Parametro:

RATE=<speed>

Ejemplo:

RATE=5Mbit

Descripcion:

Ancho de banda asignado a la clase. El tráfico que pasa a través de la clase se regula para adecuarse a la medida asignada. Usted puede utilizar Kbit, Mbit o los BPS, Kbps y Mbps son sufijos. Si usted no especifica ninguna unidad, se utilizan bits/sec. También observe que los "BPS" significan "octetos por segundo", no los bits.

Parámetro:

WEIGHT=<speed>

Ejemplo:

WEIGHT=500Kbit

Descripción:

Parámetro de tuning que debe ser proporcional a RATE. En general, WEIGHT \approx RATE / 10.

Parámetro:

PRIO=<1-8>

Ejemplo:

PRIO=5

Descripción:

Prioridad del tráfico de la clase. Cuanto más alto es el número, menor es la prioridad. La prioridad de 5 el valor por defecto.

Parámetro:

PARENT=<clsid>

Ejemplo:

PARENT=1280

Descripción:

Especifica la identificación de la clase con el padre a la cual usted desea que esta clase este vinculada. puede ser que usted desee utilizar LEAF=none para la clase del padre según lo mencionado abajo. Usando este parámetro cuidadosamente en los archivos de configuración, es posible crear las estructuras jerárquicas simples de las clases de CBQ. El orden es importante debe construir clases padre antes de sus hijos.

Parámetro:

LEAF=none|tbf|sfq

Ejemplo:

LEAF=none

Descripción:

Dice al script que relacione la disciplina de encolamiento que específico a la clase de CBQ. Por el defecto, es utilizado TBF. Observe que adjuntando TBF a la clase de CBQ controla el tráfico para conformar los parámetros de TBF y evita que la clase pida prestada ancho de banda de su padre incluso si usted HA LIMITADO el sistema a "no". Para permitir que la clase pida prestado el bandwidth, usted a debe fijar LEAF "none" o "sfq".

Si usted desea asegurar compartir del recurso de ancho de banda entre varios anfitriones en la misma clase, puede ser que desee especificar LEAF=sfq para adjuntar SFQ como disciplina que de

encolamiento de esa clase.

Parámetro:

BOUNDED=yes|no

Ejemplo:

BOUNDED=yes

Descripción:

Si se setea a "yes", la clase no se permite pedir prestado ancho de banda de la clase del padre en la situación de sobrepasar el límite asignado. Si se setea a "no", la clase permite pedir prestado ancho de banda de su padre.

Nota: No se olvide de fijar la LEAF a "none" o a "sfq", si no la clase tendrá TBF unida a sí mismo y no podrá pedir prestado el ancho de banda si usar de su padre.

Parámetro:

ISOLATED=yes|no

Ejemplo:

ISOLATED=yes

Descripción:

Si se setea a "sí", la clase no presta ancho de banda no usada a sus hijos.

Parámetros de TBF qdisc

Parámetro:

BUFFER=<bytes>[/<bytes>]

Ejemplo:

BUFFER=10Kb/8

Descripción:

Este parámetro controla la profundidad del token bucket. En otras palabras representa el tamaño máximo de burst que la clase puede enviar. La parte opcional del parámetro se utiliza para determinar la longitud de los intervalos en el tamaño de los paquetes, para los cuales se guardan los tiempos de la transmisión.

Parámetro:

LIMIT=<bytes>

Ejemplo:

LIMIT= 15Kb

Descripción:

Este parámetro determina la longitud máxima de la reserva. Si la cola contiene más datos que los especificados por LIMIT, se descartan los paquetes nuevos que llegan. La longitud de la reserva determina la latencia de la cola en caso de congestión.

Parámetro:

PEAK=<speed>

Ejemplo:

Descripción:

Tarifa máxima para el tráfico a *censurado* plazo de la explosión. Esto permite que usted controle la tarifa máxima absoluta que la clase puede enviar, porque solo TBF que permite 256Kbit/s por supuesto permitiría el índice de 512Kbit para la mitad de un segundo o un 1Mbit para un cuarto de segundo.

Parámetro:

MTU=<bytes>

Ejemplo:

MTU=1500

Descripción:

Número máximo de octetos que se pueden enviar inmediatamente sobre el medio físico. Se requiere este parámetro cuando usted especifica parámetro PEAK. Por lo general se omite el MTU de Ethernet, pero, para otros tipos de medios puede ser que desee cambiarlo.

Nota: Fijar TBF como qdisc de la hoja evitará con eficacia que la clase pida prestado ancho de banda de la clase padre, porque incluso si la clase permite que más tráfico pase a través de ella, este se adapta para conformarse con TBF.

Parámetros de SFQ qdisc

Parámetro:

QUANTUM=<bytes>

Ejemplo:

Descripción:

Este parámetro no se debe fijar más bajo que el MTU, para Ethernet es 1500b, o (con la cabecera MAC) 1514b que es el valor usado en los ejemplos de Alexey Kuznetsov.

Parámetro:

PERTURB=<seconds>

Ejemplo:

Descripción:

Período de la perturbación de la función hash. Si no se setea, la reconfiguración del hash nunca ocurrirá, que es lo que usted probablemente no desea. El valor prefijado de 10 segundos es uno bueno.

Parámetros de filtrado

Parámetro:

RULE=[[saddr[/prefix]][:port[/mask]],][daddr[/prefix]][:port[/mask]]

Ejemplo:

RULE=10.1.1.0/24:80

Descripción:

Estos parámetros levantan las reglas de filtrado "u32" que deriva el tráfico por cada una de las clases. Usted puede utilizar campos múltiples de RULE por cada configuración. La máscara de puertos opcional se debe ser utilizada solamente por usuarios experimentados que entienden cómo funciona el filtro u32.

Parámetro:

REALM=[srealm,][drealm]

Ejemplo:

REALM=russia,internet

Descripción:

Estos parámetros levantan las reglas del filtro "route" que clasifican tráfico según reinos del paquete origen/destino. Para más información sobre reinos, vea el Alexey Kuznetsov's IP Command Reference. Este script no define ningún reino, él solo construye los comandos de "filtro de tc" para usted si necesita clasificar tráfico esta manera. El reino es un número decimal o una secuencia que se refiere a `a/etc/iproute2/rt_realms` de la entrada.

Parámetro:

MARK=<mark>

Ejemplo:

Descripción:

Estos parámetros levantan las reglas del filtro del "fw" para cada una de las clases de acuerdo a "mark" del cortafuegos. La marca del paquete es un número decimal que etiqueta cada paquete que

cumple una regla si una regla del firewall así lo dicta. Usted puede utilizar campos múltiples de la MARK por cada configuración.
Nota: Las reglas para diversos tipos del filtro pueden ser combinadas. Se debe prestar atención
Last Updated lunes, octubre 07 2002 @ 07:23 ART