Transferencias masivas de información ™

Natalia Costas Lago
[natalia at cesga dot es]
Centro de Supercomputación de Galicia

Objetivos del escenario TM

Necesidad: En escenarios de elevado ancho de banda la capacidad obtenida es notablemente inferior a la capacidad nominal de los enlaces. Esto es debido al bajo rendimiento/eficiencia de:

- *Aplicaciones*
- *Sistemas (PCs/servidores, incluyendo S.O. y config TCP)*
- *La red.*

R.Carlson, The Performance Bottleneck Application, Computer, or Network, Internet2 2007

Objetivo: El escenario previsto pretende mostrar la existencia de un problema de *rendimiento bajo en la transmisión de grandes cantidades de datos* y buscar soluciones fácilmente aplicables en los entornos de la red académica.

Resultados: Como resultado se pretende proporcionar un documento que refleje la problemática, así como las soluciones encontradas, y aquellas herramientas que se identifiquen como útiles para la depuración de estos problemas.

Participantes: CESGA, CESCA, I2BASK, UVIGO

1. Teoría TCP
2. Transferencias 10G
3. Metodologías de medida
4. Pilotos de test
5. TO-DO

Introducción

CESGA

TCP: Limites teóricos de Th.

PRODUCTO BDP = VENTANA DE TRANSMISION IDEAL

- Cantidad de datos "en transito" en la red.
- BDP = bottleneck link capacity (BW) * RTT
- Ej:

BDP(1Gb,20ms) = 2.5 MByte
BDP(10Gbps,20ms) = 25 MByte

THROUGHPUT TEORICO

Rate < (MSS/RTT)*(1/sqrt(pkt_loss)) [C=1] ; C=1

(basado en la formula Mathis et.al.)
Calculador: http://www.switch.ch/network/tools/tcp_throughput/

THROUGHPUT MAXIMO

throughput ≤ TCP buffer size / RTT

TCP: Búferes y autotunning

Autotuning de los búferes DE RECEPCIÓN

/proc/sys/net/ipv4/tcp_moderate_rcvbuf = 1

Limites memoria de TCP POR CONEXIÓN (limites autotuning)

- tcp_wmem (min default max): memoria reservada búf. emisión
- tcp_rmem (min default max): memoria reservada búf. recepción
- Páginas de memoria reservadas a TCP: tcp_mem (min default max)

Limites memoria solicitada por la aplicación POR CONEXIÓN

- rmem_max,
- wmem_max

NO modificar tcp_mem (sin conocimiento). Su valor está en páginas y determina como el sistema balancea el total del espacio del búfer de red sobre otra utilización de la memoria LOWMEM.

Window scaling

/proc/sys/net/ipv4/tcp_window_scaling = 1

Asentimientos selectivos

/proc/sys/net/ipv4/tcp_sack = 1

Sellos temporales

/proc/sys/net/ipv4/tcp_timestamps = 1

Control de congestión explícito

/proc/sys/net/ipv4/tcp_ecn = 1

Mecanismos TCP

CESGA

- **Ideal: Convivencia de distintas MTU**
- **Path MTU Discovery. RFC1191, RFC 1981**

  net.ipv4.ip_no_pmtu_disc = 0 (habilitado “by default”)
- **Packetization Layer Path MTU Discovery** (=Robust Path MTU Discovery). Marzo 2007.

  RFC 4821. linux >=2.6.17

  net.ipv4.tcp_mtu_probing = 2 (off by default)

  0 Deshabilitado
  1 Deshabilitado by def., habilitado cuando se detecta Blackhole
  2 Siempre habilitado, utilizar como MSS inicial tcp_base_mss.
- **Jumbo frames … de 150KBytes o mas?.**

Jumboframes

CESGA

Seleccionar la pila TCP

Por defecto:

net.ipv4.tcp_congestion_control = bic

Configurar HTCP

modprobe tcp_hpc

Sysctl net.ipv4.tcp_congestion_control=tcp_hpc

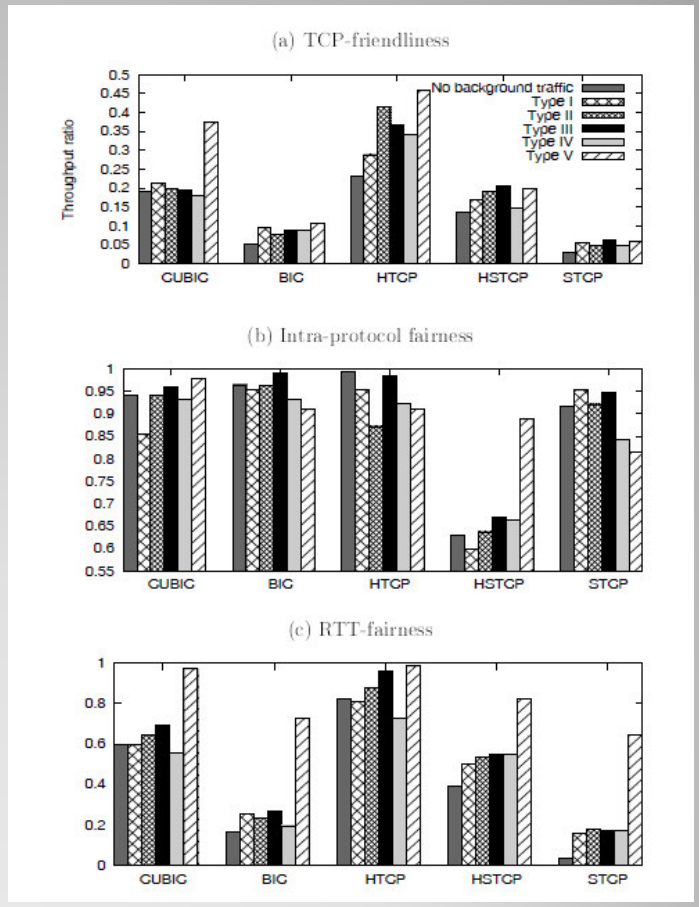
Disponibles en kernel 2.6.23:

Cubic: CUBIC-TCP

Bic: BIC-TCP

Htcp: Hamilton TCP

Westwood: optimizado para redes con perdidas



Pilas TCP alternativas

CESGA

Optimización de búferes

1. Incrementar el tamaño de búfer maximo TCP que puede fijarse con setsockopt()
   Paths 10G: 16MB (recomendado, varios flujos)
   Paths 10G o 40G muy largos: 32MB

   net.core.rmem_max = 16777216
   net.core.wmem_max = 16777216

2. Incrementar los límites para los búferes TCP en autotuning :

   net.ipv4.tcp_rmem = 4096 87380 16777216
   net.ipv4.tcp_wmem = 4096 65536 16777216

Deshabilitar soporte timestamps

net.ipv4.tcp_timestamps = 0

Desactivar soporte SACK

net.ipv4.tcp.sack = 0

Optimizaciones para 10G

Optimizar el rendimiento de las NICs

Optimizar el número de peticiones SYN en cola (kernel 2.6)

ifconfig eth0 txqueuelen 5000

(Se recomienda un valor de 5000-10000 para paths de más de 50 ms de RTT.)

Optimizar la cola del interfaz en recepción

net.core.netdev_max_backlog = 30000 (default 1000)

Habilitar TCP segmentation offload (Otros: Large receive offload):

ethtool –k eth0

Habilitar Interrupt Coalescing (Mejoras: NAPI)

ethtool –C eth0

Asignar CPU Binding/Affinity (si se necesita): Por puerto, PID, interrupción…
Tx checksum, Rx checksum, etc.

Optimizaciones para 10G

No cachea el estado de conexiones previas

net.ipv4.tcp_no_metrics_save = 1

Rendimiento del algoritmo Nagle con Delayed ACKs

- Recomendado: default on.
- Excepción: Aplicaciones muy interactivas
- Posibles problemas de rendimiento: En interacciones con Delayed ACKS

Recomendaciones para 10G

IPV4 specific settings
#turns TCP timestamp support off, default 1, reduces CPU use net.ipv4.
#tcp_timestamps = 0
 # turn SACK support off, default on systems with a VERY fast bus ->
 # memory interface this is the big gainer
net.ipv4.tcp_sack = 0
 # sets min/default/max TCP read buffer, default 4096 87380 174760
net.ipv4.tcp_rmem = 4096 87380 16777216
 # sets min/pressure/max TCP write buffer, default 4096 16384 131072
net.ipv4.tcp_wmem = 4096 87380 16777216
 # sets min/pressure/max TCP buffer space, default 31744 32256 32768
net.ipv4.tcp_mem = 4096 87380 16777216
CORE settings (mostly for socket and UDP effect)
 # maximum receive socket buffer size, default 131071
net.core.rmem_max = 524287
 # maximum send socket buffer size, default 131071
net.core.wmem_max = 524287
 # default receive socket buffer size, default 65535
net.core.rmem_default = 524287
 # default send socket buffer size, default 65535
net.core.wmem_default = 524287
 # maximum amount of option memory buffers, default 10240
net.core.optmem_max = 524287
 # number of unprocessed input packets before kernel starts dropping
 # them, default 300
net.core.netdev_max_backlog = 300000

Ejemplo de configuración TCP 10G

Posibles problemas

- Sobresuscripción
- Colas/búferes en interfaces entrada/salida
- Capacidad hardware/software
- Soporte MTU (y soporte MTU en interfaces enrutados)
- Límites para soporte de flujos de gran tamaño (diseñados para agregación)

PD: ¿Son fiables las hojas de características técnicas…..?

Límites en los elementos de red

- En servidores atención a:
  - CPU
  - Transferencia I/O (HD, Disco)
  - Buses (10G=> PCI Express)
  - Otro hardware: NICs
  - Protocolos utilizados

- En aplicaciones:
  - Conocer el funcionamiento (Caso openssh)

Límites en aplicaciones/servidores

Condición:

- Equilibrio TCP
- Integridad de red L2/L3 [RFC2544]
- Definido por el IPPM del IETF

Pasos:

- Obtención MTU: cota alta al MSS.
- Obtención RTT y BW: Tamaño de ventana
- Tests de throughput de la conexión TCP

1. Simples
2. Multiples

- Test de gestión del tráfico (opcional)

    Traffic shaping, priority queue, … con flujos múltiples

Tener en cuenta:

- Implementación y opciones TCP
- Capacidad de equipos emisor/receptor

Framework for TCP Throughput Testing

- http://tools.ietf.org/html/draft-ietf-ippm-tcp-throughput-tm-07#section-3

Framework for TCP Th Testing

Objetivo:

Definición de un conjunto pequeño de métricas robusto, fáciles de comprender, ortogonales, relevantes y fáciles de computar que describan el estado de la red al usuario final.

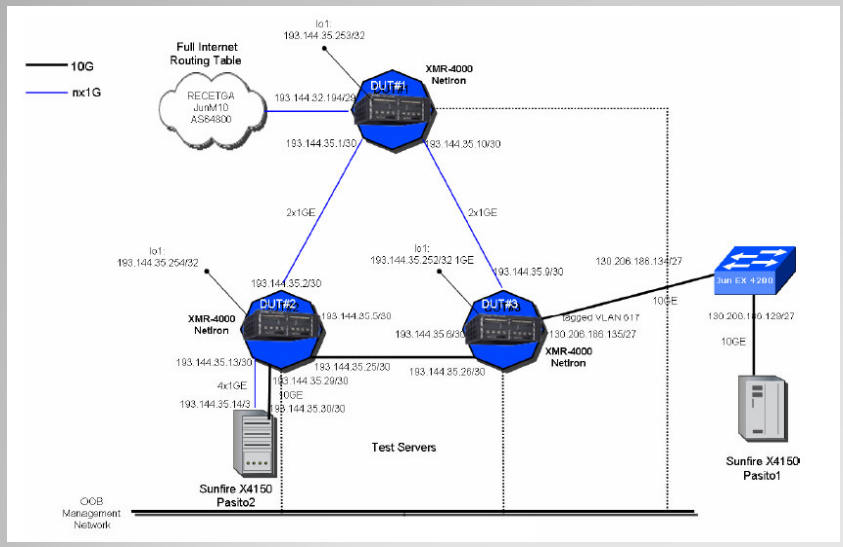
Definido por el IPPM (IETF).

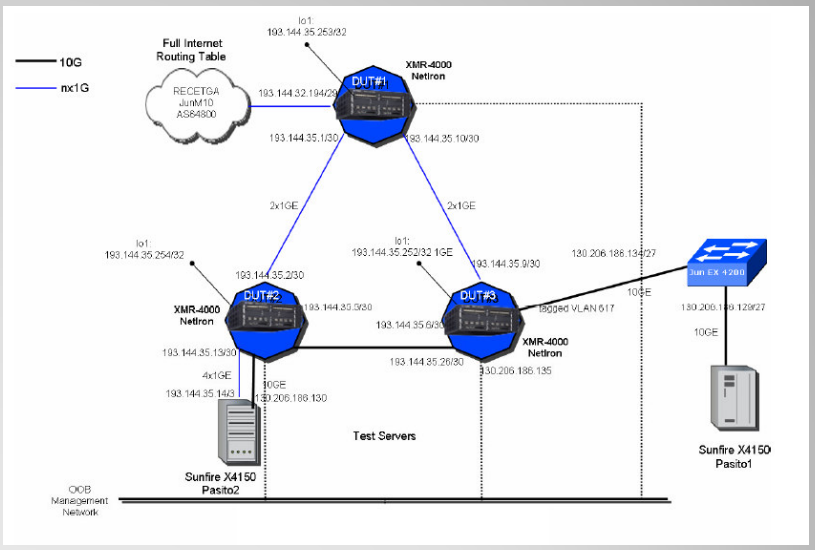
1. Retardo
2. Pérdida de pkts
3. Jitter
4. Duplicados
5. Reordenación

Reporting IP Performance Metrics to Users

http://tools.ietf.org/html/draft-shalunov-ippm-reporting-03#page-4

Metricas de usuario final





Caso 1: Piloto LAN 10G

CESGA

Pruebas preliminares co las estaciones Sunfire

- **Características de los servidores**
  - CPU: 2 intel Xeon Quad-core 2.33GHz
  - Disco Duro: 146GB SAS
  - RAM: 2G PC2-5300 DDR2 Fully Buffered
- **Transmisión punto a punto**
  - Transmisión 1Gbps y 10Gbps configuración parámetros TCP básicos: máx 3.9Gbps
  - Transmisión 10Gbps con mejora MTU: máx. 6.3Gbps. Limitación: CPU
  - **Transmisión de varios flujos simultáneos: máx. 9.92Gbps**
  - Transmisión de disco a memoria: 3.2Gbps

Transmisión a través de red enrutada y VPN L2

- Transferencias a de red enrutada: máx. ~9.83 Gbps, 9.5Gbps bidireccional

Problemas encontrados:

- Capacidad conmutación entre puertos 10G de uplink
- Límite CPU en servidores
- Límite velocidad transferencia HD servidores

Caso 1: Resultados

CESGA

THROUGHPUT TEORICO

Rate < (MSS/RTT)*(1/sqrt(pkt_loss)) [C=1] ; C=1

(basado en la formula Mathis et.al.)

Calculador: http://www.switch.ch/network/tools/tcp_throughput/

VENTANA DE TRANSMISION IDEAL

Ventana ideal = bottleneck link capacity (BW) * RTT

= 1Gbps*53ms = **6.75MB**

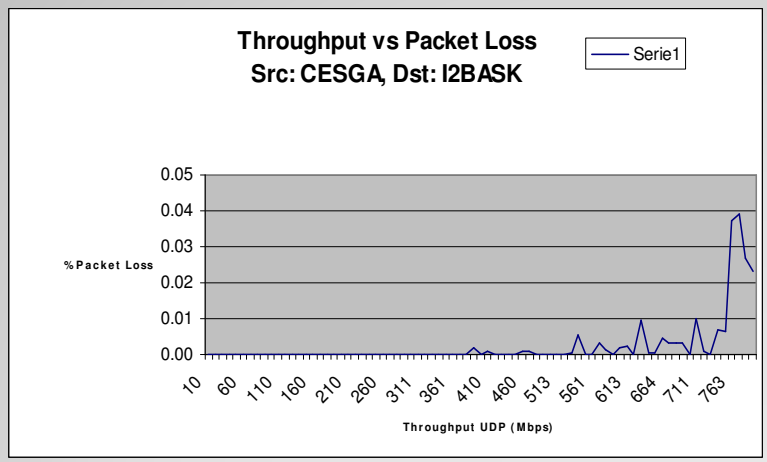
THROUGHPUT MAXIMO

throughput <= TCP buffer size / RTT = 16MB/53ms = **2473Mbps**

Caso 2: Piloto WAN

CESGA

- **Condición:**
  - Integridad de red L2/L3
  - Aparentemente pérdida de pkts en sentido origen I2BASK destino CESGA.



Parametros de red:

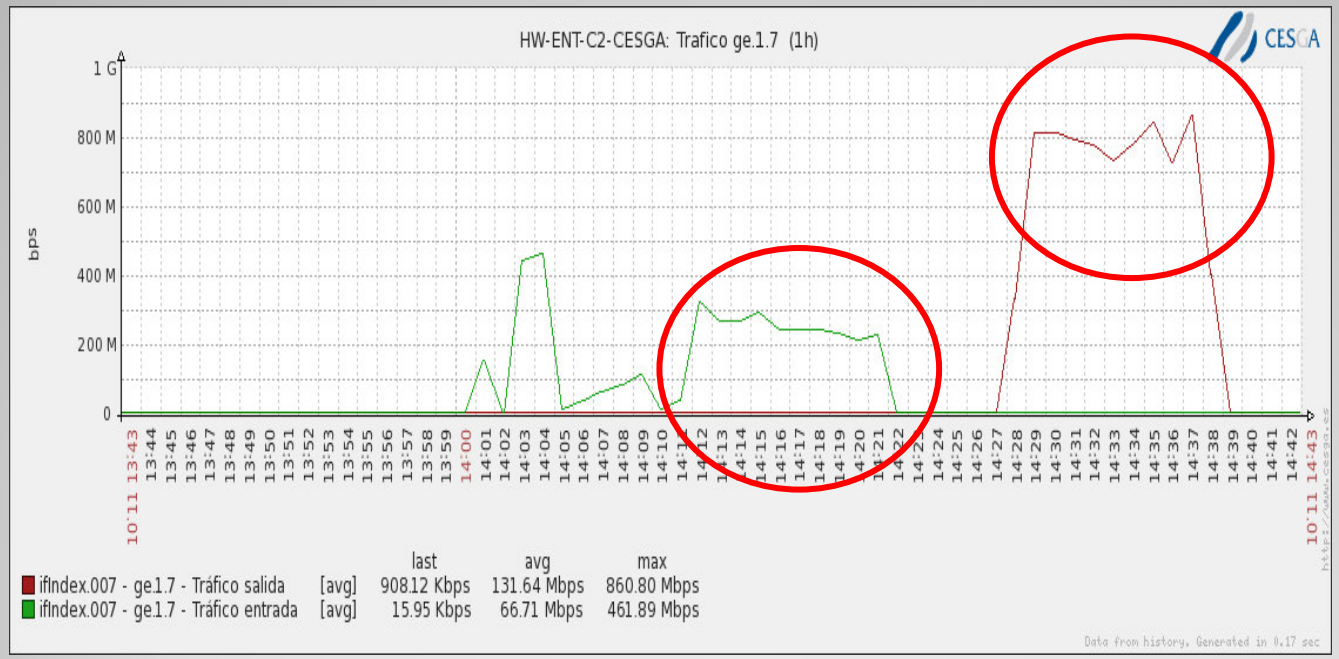
MTU: 9000

RTT: **53 ms**

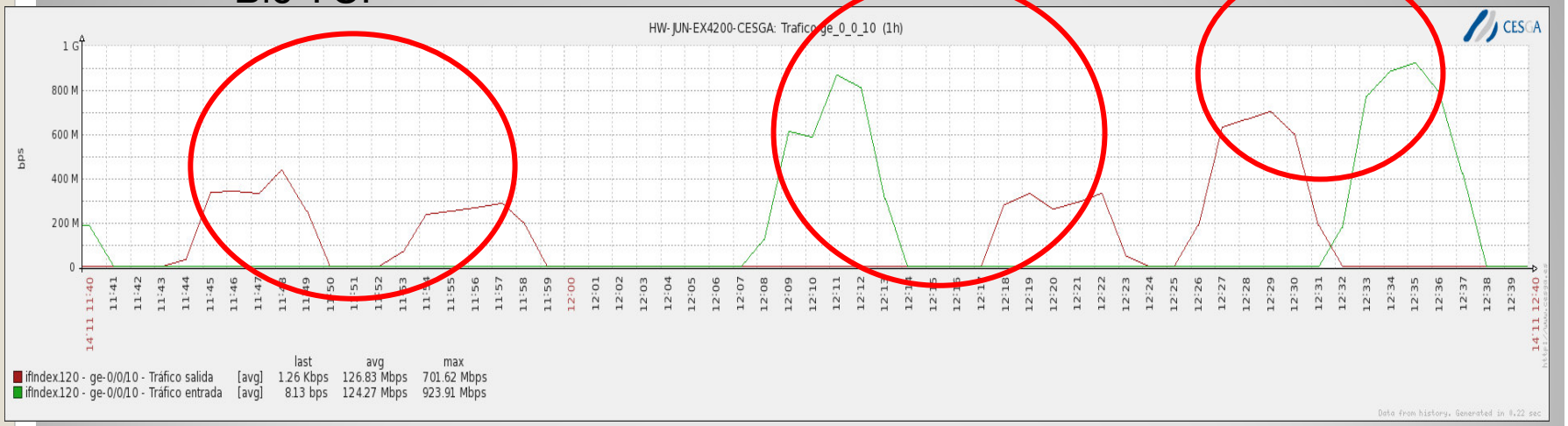
Bottleneck BW: 1Gbps

Caso 2: Verificar estado conexión

- **Transferencia** TCP mediante **IPERF**
  - Origen CESGA, Destino I2BASK: **780 Mbps**
  - Origen I2BASK, Destino CESGA: **256 Mbps** (CPU 7-10%)



Caso1: Th. TCP 1 flujo (1 way)



Caso1: Th. HTCP 1 flujo (2 way)

- **Problemas encontrados:**

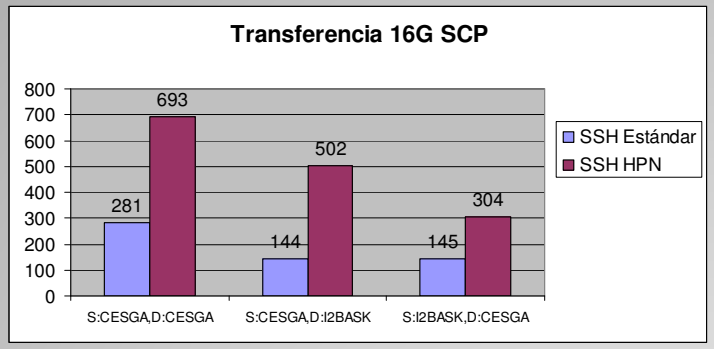
- Límite MTU en switch
- Pérdida de paquetes en línea
- Software xen afecta MTU
- Software vmware afecta rendimiento
- Bug en IPERF para medidas UDP

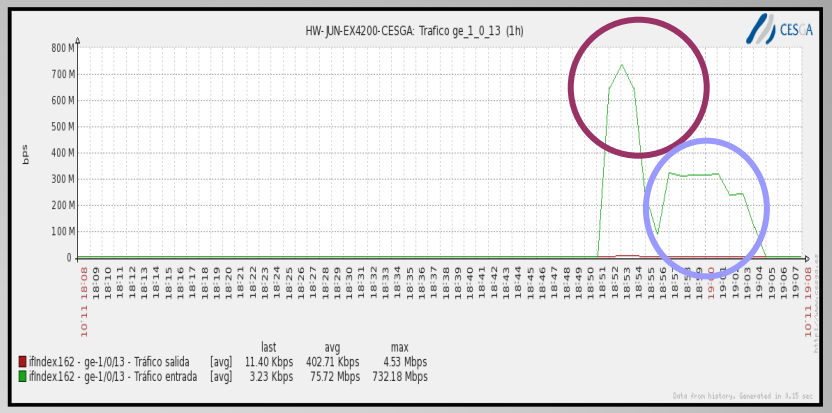
Otros:
En pruebas preliminares CESGA-UVIGO se detectó limite MTU en equipos CWDM

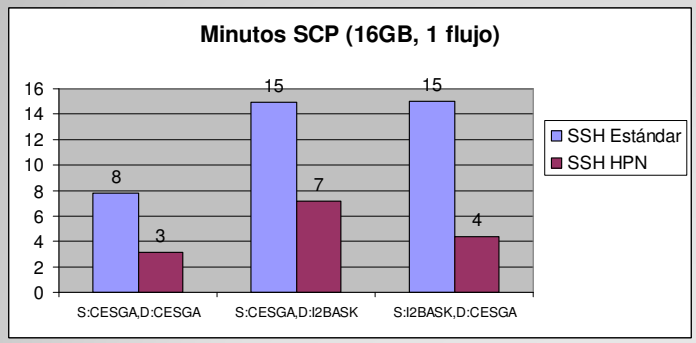
Problemas encontrados

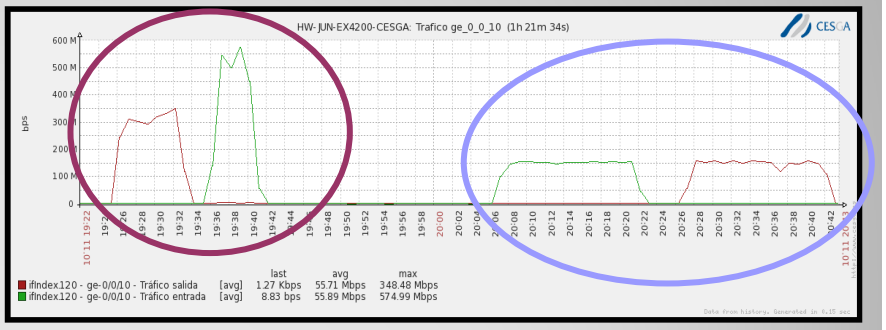
CESGA

A) SCP local en CESGA
B) SCP CESGA - I2BASK









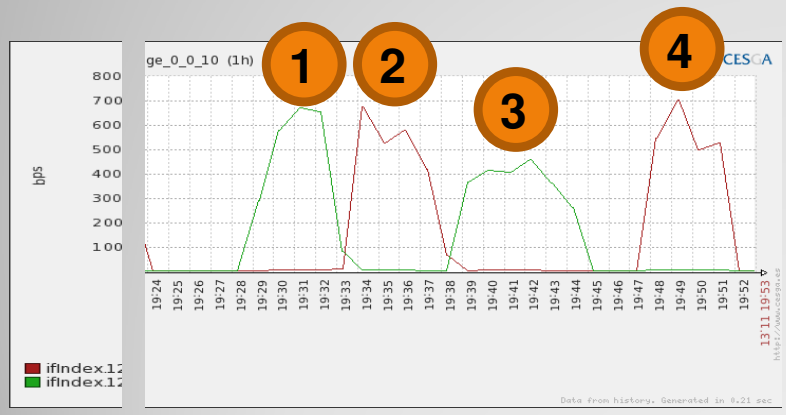
Ref: High Performance Enabled SCP/SSH
http://www.psc.edu/networking/projects/hpn-ssh/ /

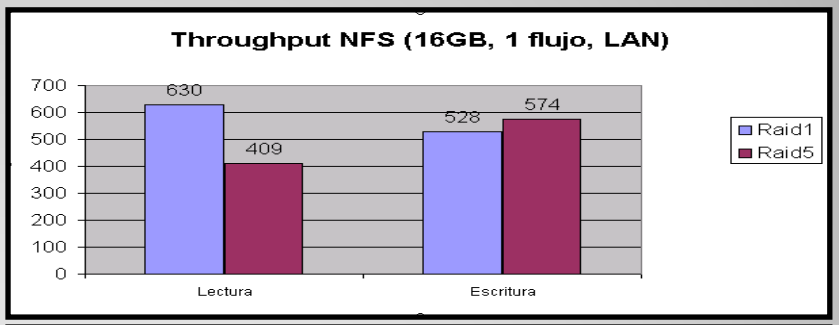
Caso 2: 16Gbps via SCP

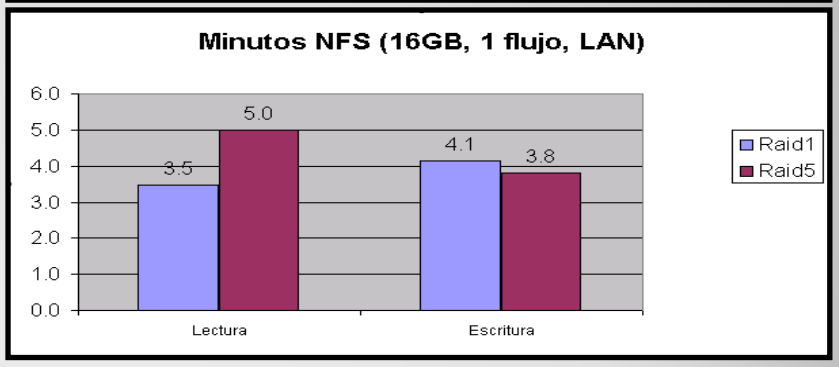
CESGA

- NFS v2: UDP por defecto
- NFS v3: TCP pror defecto
- Verificar el estado de la red
- Verificar capacidad del sistema: HD, CPU
- Según versión de NFS es necesario verificar tamaños de bloque lectura/escritura (rsize,wsize).
  - En v3 estos parametros ya se gestionan automáticamente.

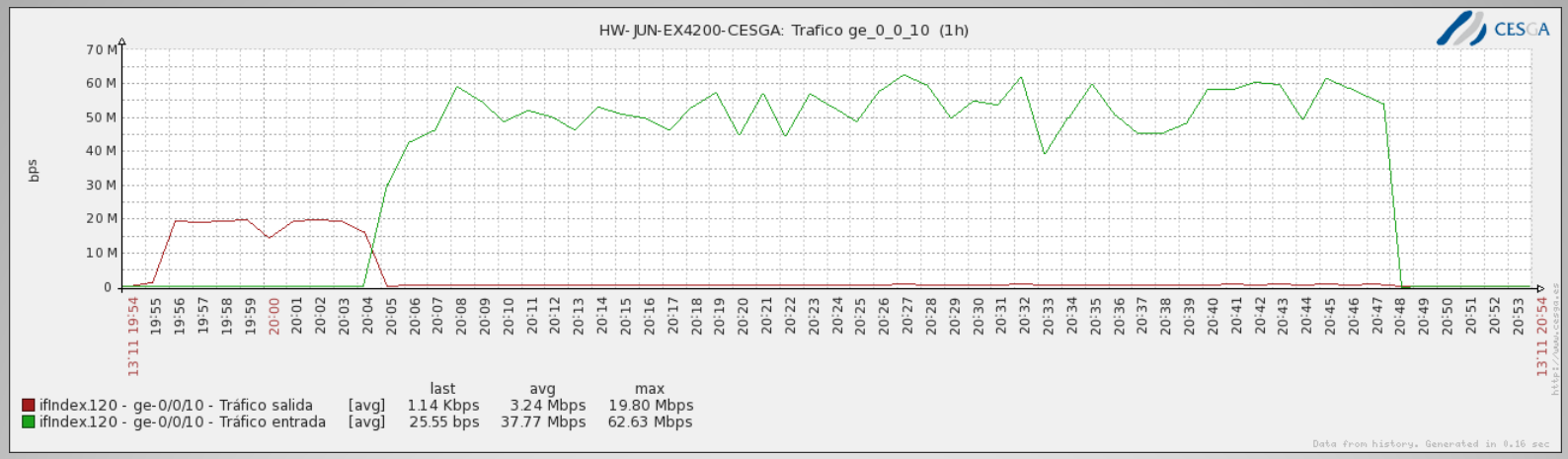
1.- Raid5, Lectura
2.- Raid5, Escritura
3.- Raid1, Lectura
4.- Raid1, Escritura







Transferencias NFS. En LAN



Pendiente de analizar. A Priori el resultado

Transferencia NFS. CESGA-I2BASK

Medición en enlaces

1. Verificar medidas CESGA-I2BASK a 1G
2. Verificar algoritmo FERFAST (CESGA-CESCA)
3. Prueba de transferencia en RedIris Nova
4. Identificar disciplina TCP más ajustada a entorno LAN y WAN

Medición en sistemas virtualizados

Medición en aplicaciones

1. NFS (en curso)
2. Sistema de archivos distribuido (SFS/IBRIX)
   - Cliente NFS
   - Cliente nativo IBRIX

TO-DO

CESGA

¡Gracias!

Natalia costas Lago
natalia at cesga dot es