# ANEXOS

# ÍNDICE ANEXOS

# ANEXOS I

# CONFIGURACIONES REALIZADAS

# 1. COFIGURACIÓN DE REPOSITORIOS

## 1.1. REPOSITORIO LOCAL

a) Copiamos los archivos requeridos de la unidad donde se ubican los archivos de instalación del sistema.

```
# mount /media/RHEL_6.5\ x86_64\ Disc\
# mkdir -p /var/repo/rhel65
# cp -r /media/RHEL_6.5\ x86_64\ Disc\ 1/* /var/repo/rhel65/
```

b) Para la creación del repositorio local requeriremos la instalación previa de la siguiente paquetería para ello desde el directorio /var/repo/rhel62/Packages/ ejecutamos los siguientes comandos

```
#rpm -ivh deltarpm-3.5-0.5.20090913git.el6.x86_64.rpm
#rpm -ivh python-deltarpm-3.5-0.5.20090913git.el6.x86_64.rpm
#rpm -ivh createrepo-0.9.8-4.el6.noarch.rpm
```

c) Una vez completada la instalación ejecutamos el siguiente comando:

```
#createrepo rhel65/
```

d) Luego vamos al directorio /etc/yum.repos.d/ y crearemos el archivo rhel-local.repo y lo editaremos de la siguiente manera:

```
[rhel-local]
gpgcheck=1
name=Red Hat linux $releaseserver - $basearch - DVD
baseurl=file:///var/repo/rhel65
```

e) Instalamos el certificado GPG del repositorio para ello vamos al directorio /etc/pki/rpm-gpg/ y ejecutamos los siguientes comandos:

```
#rpm --import RPM-GPG-KEY-redhat-beta
#rpm --import RPM-GPG-KEY-redhat-release
```

f) Por último podemos comprobar que yum utiliza el nuevo repositorio ejecutando el siguiente comando:

```
#yum clean all
#yum repolist
```

## 1.2. REPOSITORIO EPEL CENTOS AND RED HAT LINUX 6.X

a) Para la instalación del repositorio EPEL ejecutamos los siguientes comandos:

```
#wget http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-8.noarch.rpm
#sudo rpm -Uvh epel-release-6*.rpm
```

b) Si durante el proceso de instalación apareciera el mensaje de error: "Cannot retrieve metalink for repository: epel. Please verify its path and try again" ejecutamos la siguiente línea de comandos:

```
#sudo sed -i "s/mirrorlist=https/mirrorlist=http/" /etc/yum.repos.d/epel.repo
#yum -y update
```

c) Instalamos el certificado GPG del repositorio para ello vamos al directorio /etc/pki/rpm-gpg/ y ejecutamos los siguientes comandos:

```
#rpm --import RPM-GPG-KEY-EPEL-6
```

# 2. INSTALACIÓN DE PAQUETES

## 2.1. OPENSSL

a) Al no contar con la versión requerida en ninguno de los repositorios instalados descargamos los paquetes de cualquiera de los dos links adjuntos:

```
#wget ftp://ftp.muug.mb.ca/mirror/centos/6.5/updates/x86_64/Packages/openssl-1.0.1e-16.el6_5.14.x86_64.rpm
#wget ftp://mirror.switch.ch/pool/4/mirror/scientificlinux/6.4/x86_64/updates/security/openssl-1.0.1e-16.el6_5.14.x86_64.rpm
```

b) Instalamos los paquetes ejecutando el siguiente comando.

```
#rpm -Uvh openssl-1.0.1e-16.el6_5.14.x86_64.rpm
```

## 2.2. APACHE SERVER

a) Instalamos los paquetes de los repositorios instalados ejecutando el siguiente comando.

```
#yum install httpd httpd-devel
```

## 2.3. MYSQL DATABASE

a) Instalamos los paquetes de los repositorios instalados ejecutando el siguiente comando.

```
#yum install mysql mysql-server
```

b) Seguimos los siguientes pasos para crear la base de datos a utilizar por Cacti.

```
# mysql -u root -p -e cacti
# mysql -u root –p
mysql> GRANT ALL ON cacti.* TO cacti@localhost IDENTIFIED BY 'password';
mysql> FLUSH privileges;
mysql> exit;
```

## 2.4. PHP 5.3

a) Instalamos los paquetes de los repositorios instalados ejecutando el siguiente comando.

```
#yum install php-mysql php-pear php-common php-gd php-cli php-mysql
```

b) Para los paquetes como php-mbstring y php-devel que no los encontramos en los repositorios los instalamos de la siguiente manera:

```
#wet http://linuxsoft.cern.ch/cern/slc61/x86_64/yum/updates/php-mbstring-5.3.3-3.el6_1.3.x86_64.rpm
#rm -ivh php-mbstring-5.3.3-3.el6_1.3.x86_64.rpm
```

```
#wget http://linuxsoft.cern.ch/cern/slc61/x86_64/yum/updates/php-devel-5.3.3-3.el6_1.3.x86_64.rpm
```

```
#rpm -ivh php-devel-5.3.3-3.el6_1.3.x86_64.rpm
```

### 2.5. PHP-SNMP 5.3

a) Al no contar con el paquete de instalación en los repositorios configurados ejecutamos los siguientes comandos.

```
#wget http://linuxsoft.cern.ch/cern/slc61/x86_64/yum/updates/php-snmp-5.3.3-3.el6_1.3.x86_64.rpm
#rpm -Uvh php-snmp-5.3.3-3.el6_1.3.x86_64.rpm
```

### 2.6. PHP 5.5

a) En caso que se desee usar una versión más actual al no encontrar los paquetes requeridos en los repositorios utilizaremos el siguiente repositorio e instalaremos los paquetes requeridos tal como sigue:

```
#rpm -Uvh https://mirror.webtatic.com/yum/el6/latest.rpm
```

```
#yum install php55w php55w-pdo php55w-mysql php55w-opcache php55w-pear php55w-snmp php55w-devel php55w-mbstring php55w-gd
```

### 2.7. NET-SNMP

a) Instalamos los paquetes de los repositorios instalados ejecutando el siguiente comando.

```
#yum install net-snmp-utils net-snmp-libs php-pear-Net-SMTP
```

### 2.8. RRDTool

a) Instalamos los paquetes de los repositorios instalados ejecutando el siguiente comando.

```
#yum install rrdtool
```

# 3. INSTALACIÓN DE CACTI

a) Instalamos Cacti haciendo uso de los archivos de instalación de la siguiente manera:

```
#cd /var/www/html/
#wget http://www.cacti.net/downloads/cacti-0.8.8c.tar.gz
#tar -xzvf cacti-0.8.8c.tar.gz
#ln -s cacti-0.8.8c cacti
```

b) De la base de datos mysql que creamos anteriormente la vinculamos a Cacti con el siguiente comando.

```
#mysql -p cacti < /var/www/html/cacti/cacti.sql
```

c) Vamos al directorio de Cacti y modificamos el archivo config.php tal como se indica.

```
#cd /var/www/html/cacti/include
#vim config.php
```

```
$database_type = "mysql";
$database_default = "cacti";
$database_hostname = "localhost";
$database_username = "cacti";
$database_password = "password";
$database_port = "3306";
```

```
$config['url_path'] = '/'
$config['url_path'] = '/cacti/'
```

d) Para permitir el acceso al servidor Apache desde el segmento LAN de la red configuramos el segmento IP en el archivo indicado de la siguiente manera.

```
# vim /etc/httpd/conf.d/cacti.conf
```

```
#
# Cacti: An rrd based graphing tool
#
Alias /cacti    /usr/share/cacti

<Directory /usr/share/cacti/>
        Order Deny,Allow
        Deny from all
        Allow from 10.0.0.0/8
</Directory>
```

e) Finalmente configuramos el Crontab del sistema con el poller de Cacti.

```
#vim /etc/cron.d/cacti

*/5 * * * * cactiuser /usr/bin/php /var/www/html/cacti/poller.php > /dev/null 2>&1
```

f) Para iniciar los servicios requeridos por Cacti de manera automática cada vez que el equipo se encienda ejecutamos los siguientes comandos.

```
#/sbin/chkconfig --levels 345 httpd on
#/sbin/chkconfig --levels 345 mysqld on
#/sbin/chkconfig --levels 345 snmpd on
#/sbin/chkconfig --levels 345 iptables off
```

### 3.1. POOLER SPINE

a) Para la instalación del poller Spine realizaremos la instalación previa de los siguientes paquetes.

```
#yum install gcc mysql-devel net-snmp-devel autoconf automake dos2unix libtool
```

b) Descargamos los paquetes de instalación tal como se indica y seguimos los siguientes pasos.

```
#cd /tmp/
#wget http://www.cacti.net/downloads/spine/cacti-spine-0.8.8c.tar.gz
#tar -xzvf cacti-spine-0.8.8c.tar.gz
#cd cacti-spine-0.8.8c
#./bootstrap
#./configure
#make
#make install
#cp /usr/local/spine/etc/spine.conf.dist /etc/spine.conf
```

c) Una vez instalado el poller modificamos el archivo spine.conf con los valores previamente configurados para la base de datos de cacti.

```
#vim /etc/spine.conf
```

```
DB_Host         localhost
DB_Database     cacti
DB_User         cactiuser
DB_Pass         password
DB_Port         3306
DB_PreG         0
```

d) Finalmente validamos los logs de Cacti de la siguiente manera donde validaremos el correcto funcionamiento del poller

```
#ln -s /usr/local/spine/bin/spine /sbin/spine
#path: /usr/local/spine/bin/spine
#tail -f /var/www/cacti/log/cacti.log
```

### 3.2. SENDMAIL

a) Instalamos los paquetes requeridos haciendo uso de los repositorios configurados de la siguiente manera:

```
#yum install sendmail
```

b) Modificamos el archivo sendmail.cf para definir la IP o el hostname del relay SMTP del que haremos uso para poder enviar correos.

```
#vim /etc/mail/sendmail.cf
```

```
#"Smart" relay host (may be null)
DS"Hostame or IP"
```

c) Para iniciar el servicio y validar el correcto funcionamiento del servicio hacemos uso de los siguientes comandos.

```
#service sendmail start
#tail -f /var/log/maillog
```

d) Si dentro de las pruebas de envio de correos aparece el error: sendmail[12006]: NOQUEUE: SYSERR(apache): can not hdir(/var/spool/clientmqueue/): Permission denied. Ejecutamos el siguiente comando.

```
#setsebool -P httpd_can_sendmail on
```

### 3.3. SYSLOG

a) Para el funcionamiento del Syslog instalamos el servicio rsyslog de la siguiente manera.

```
#yum install rsyslog
```

b) Para validar el estado del servicio hacemos uso de los siguientes comandos:

```
#service rsyslog status
#chkconfig rsyslog on
```

c) Los mensajes deberán ser almacenados en una base de datos por separado para lo cual creamos una base de datos adicional tal como sigue a continuación:

```
#mysql -u root -p
mysql> CREATE DATABASE syslog;
mysql> GRANT ALL ON syslog.* TO sysloguser@localhost IDENTIFIED BY
'passwd';
mysql> flush privileges;
mysql> exit;
```

d) Para definir los puertos en uso y los segmentos de red involucrados en la comunicación se modificara el archivo rsyslog.conf de la siguiente manera.

```
#vim /etc/rsyslog.conf
```

```
#### MODULES ####
$ModLoad imuxsock # provides support for local system logging (e.g. via logger command)
$ModLoad imklog   # provides kernel logging support (previously done by rklogd)
#$ModLoad immark  # provides --MARK-- message capability
# Provides UDP syslog reception

## uncomment ##
$ModLoad imudp
$UDPServerRun 514

# Provides TCP syslog reception
## Uncomment ##
$ModLoad imtcp
$InputTCPServerRun 514

## Add the following lines ##
$ModLoad ommysql
$ModLoad ommysql
*.* :ommysql:127.0.0.1,syslog,sysloguser,password
$AllowedSender UDP, 127.0.0.1, 192.168.1.0/24
$AllowedSender TCP, 127.0.0.1, 192.168.1.0/24
```

e) Finalmente dentro del pluging de Cacti para el funcionamiento del Syslog se modificara el archivo config.php tal como se indica a continuación.

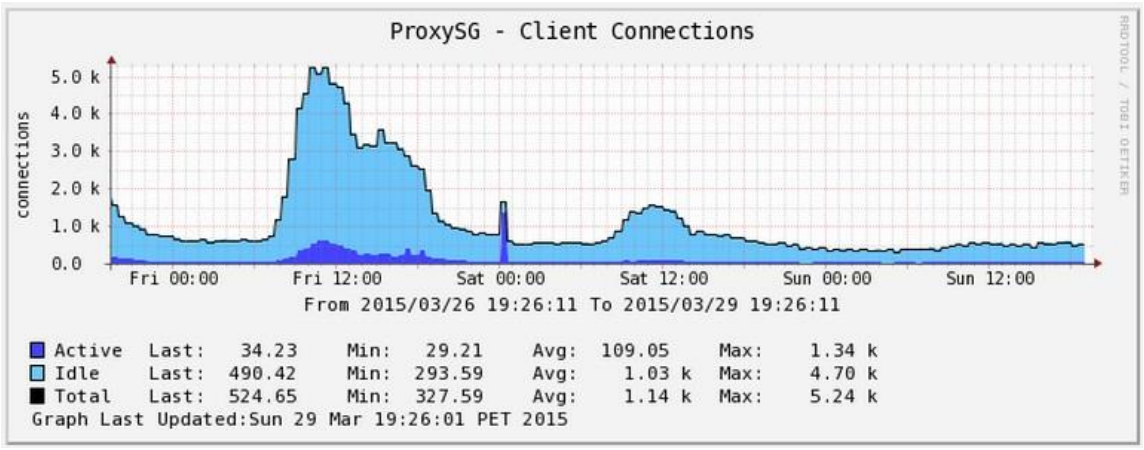
```
#cd /var/www/html/cacti/plugins/syslog
#vim config.php
```

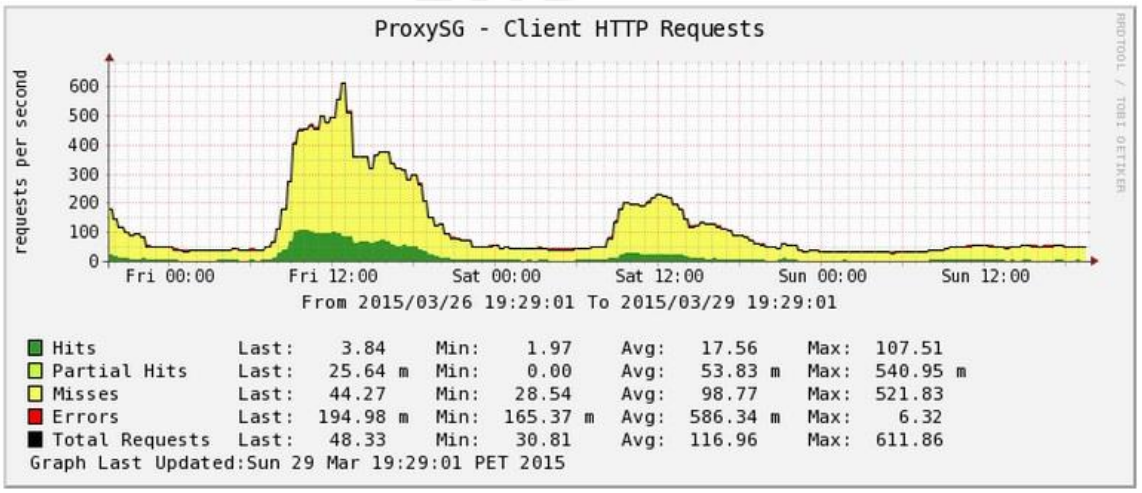
```
$syslogdb_type     = 'mysql';
$syslogdb_default  = 'syslog';
$syslogdb_hostname = 'localhost';
$syslogdb_username = 'sysloguser';
$syslogdb_password = 'password';
$syslogdb_port     = 3306;
```

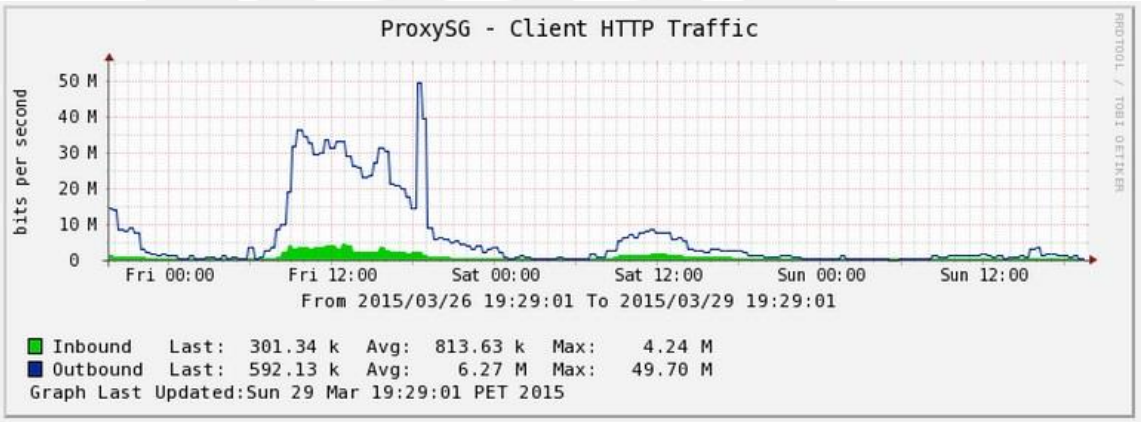
# ANEXOS II

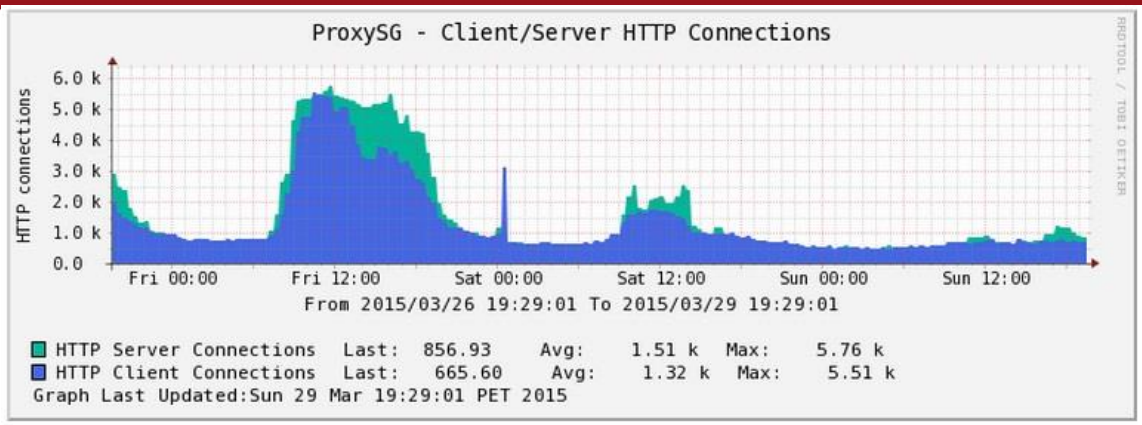
# GRÁFICAS OBTENIDAS

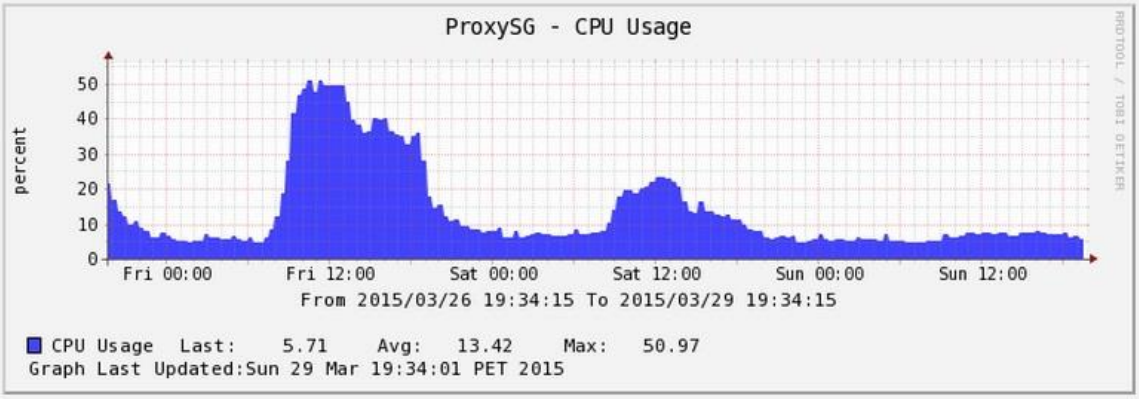
# 1. PROXY WEB BLUECOAT SG900-30

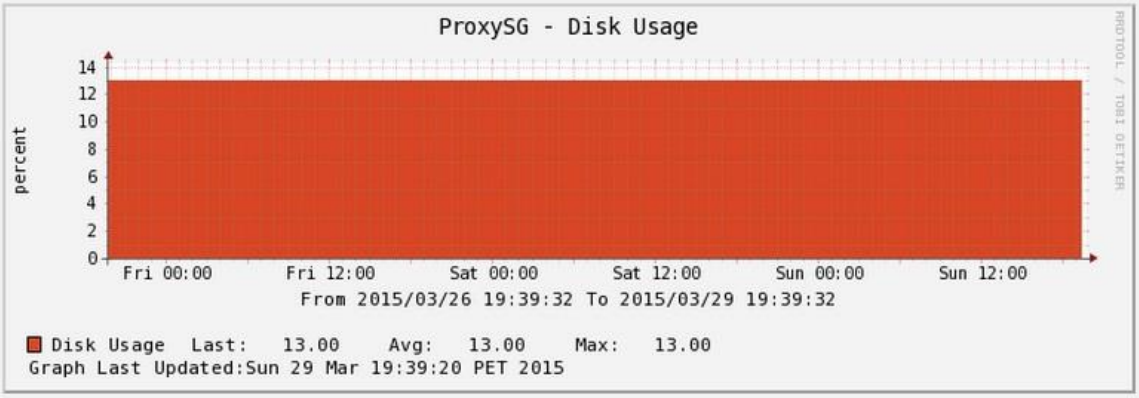


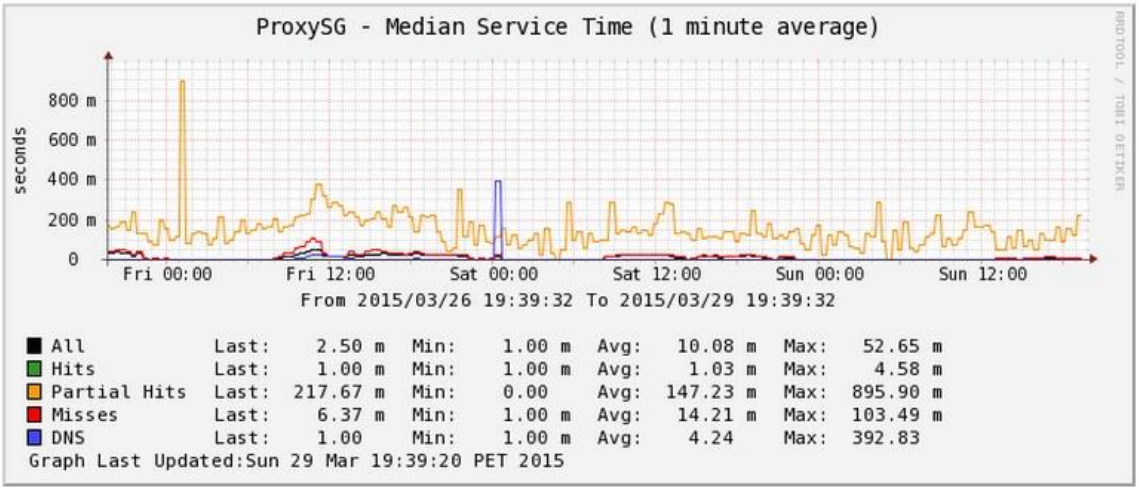



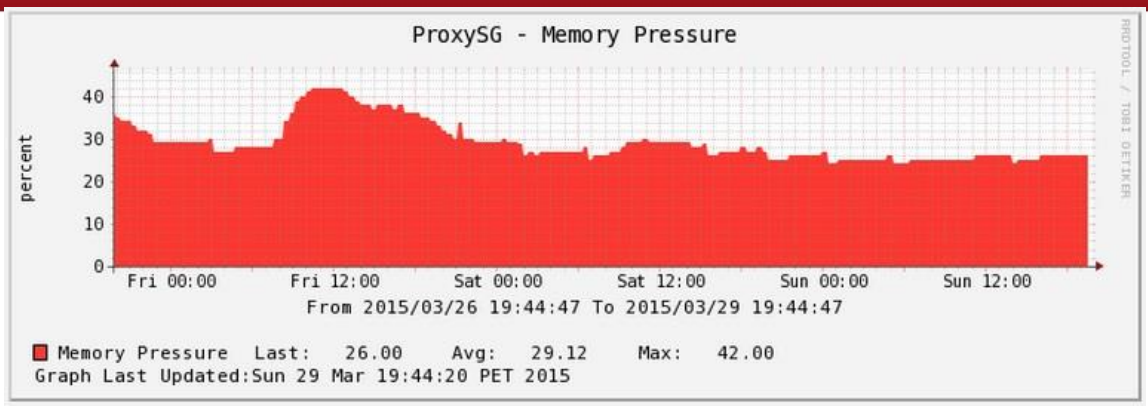



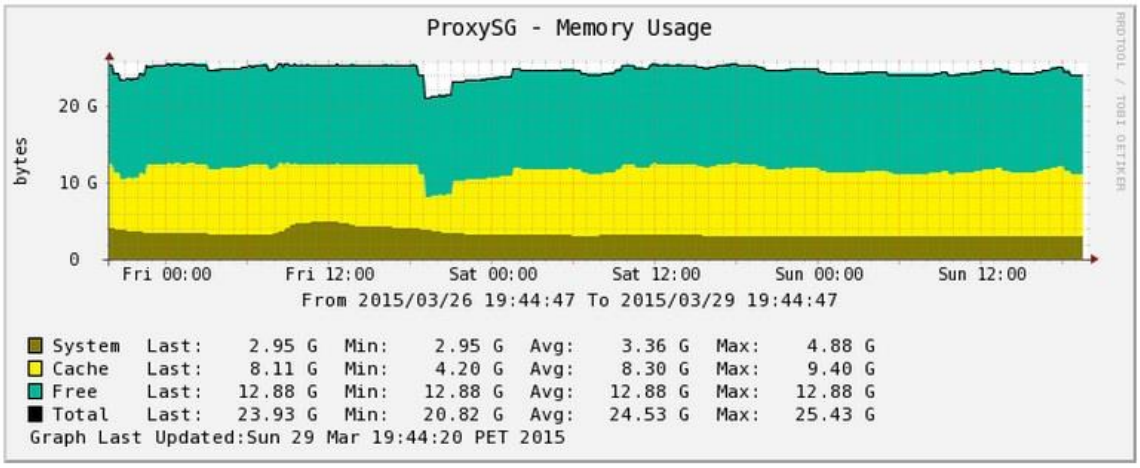


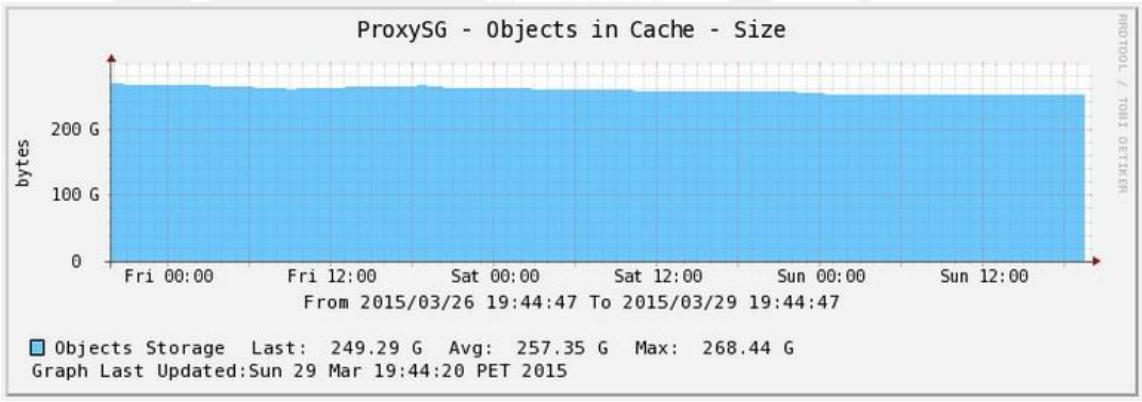



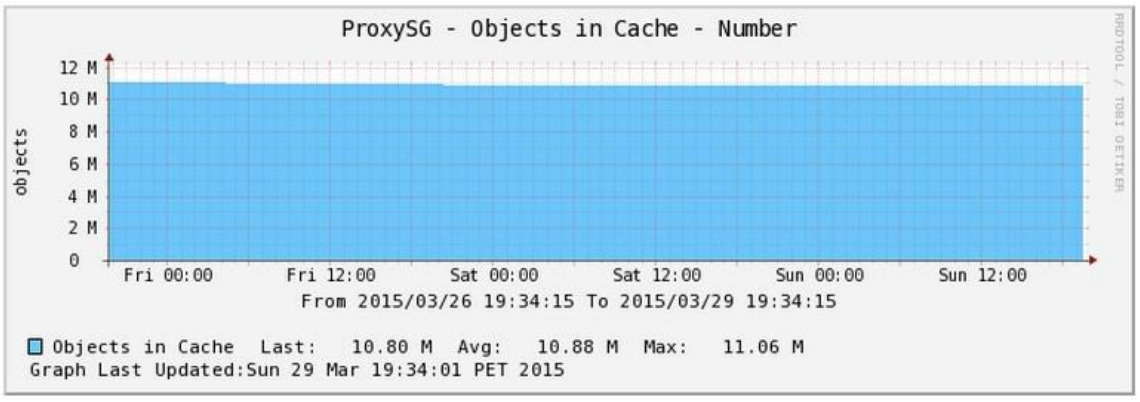



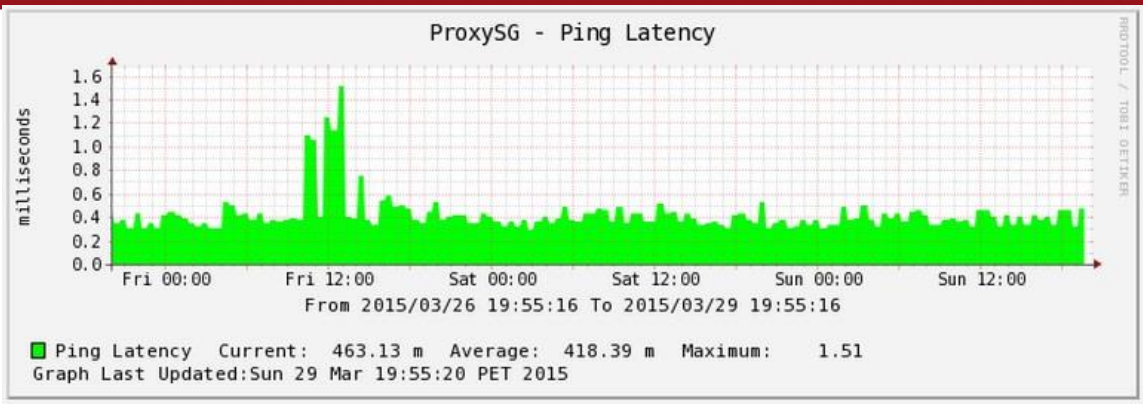



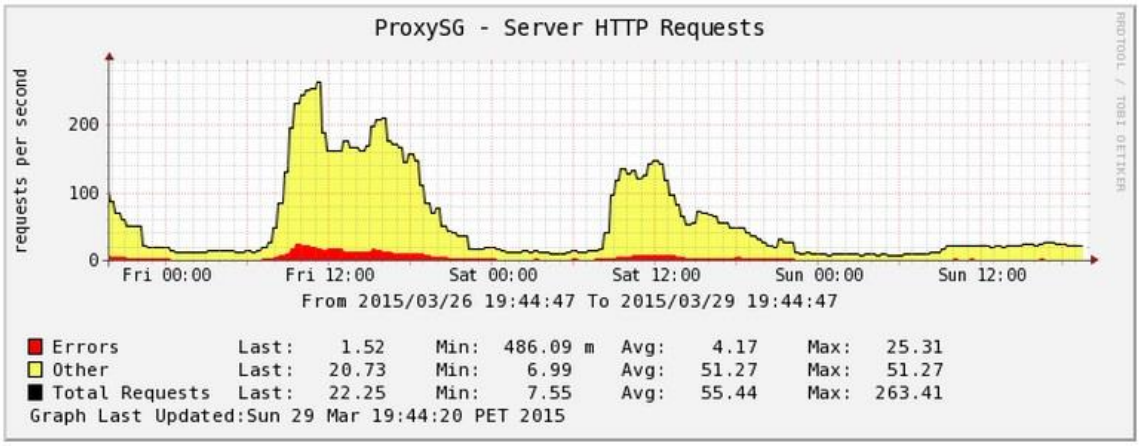


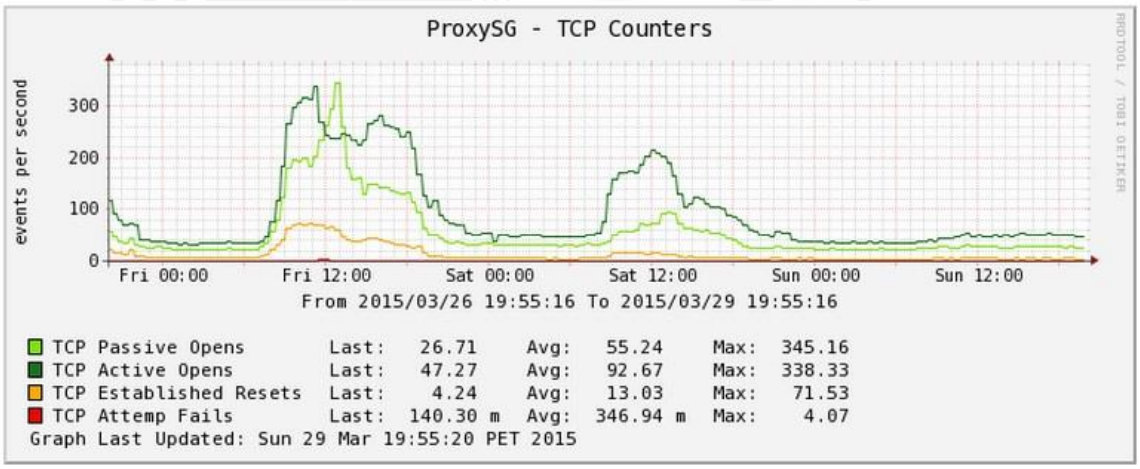



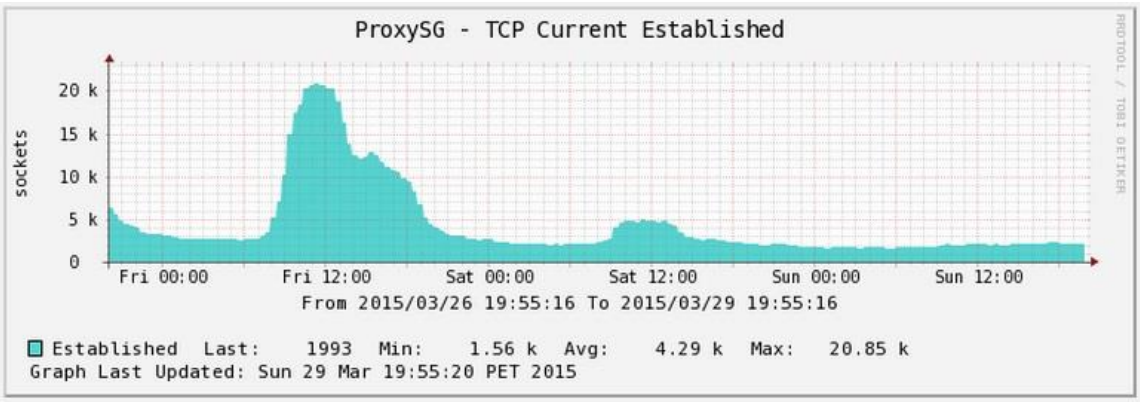



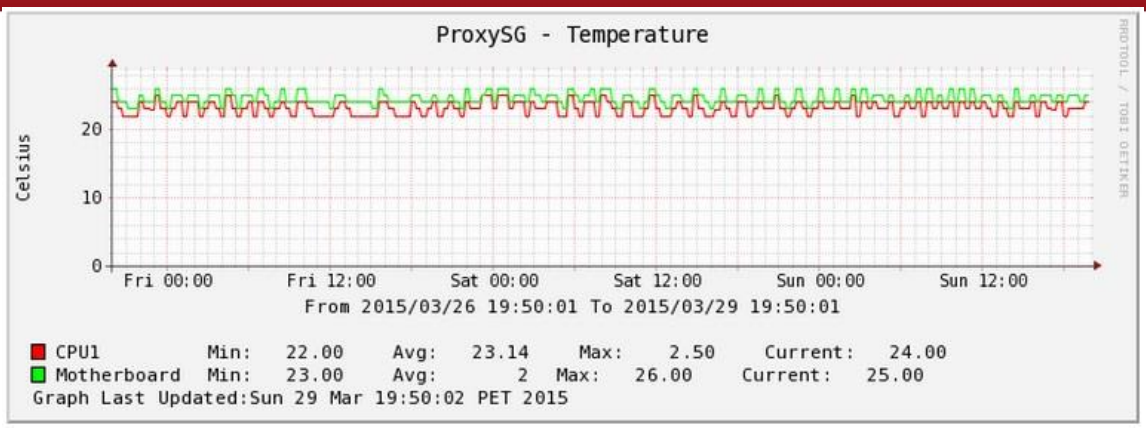



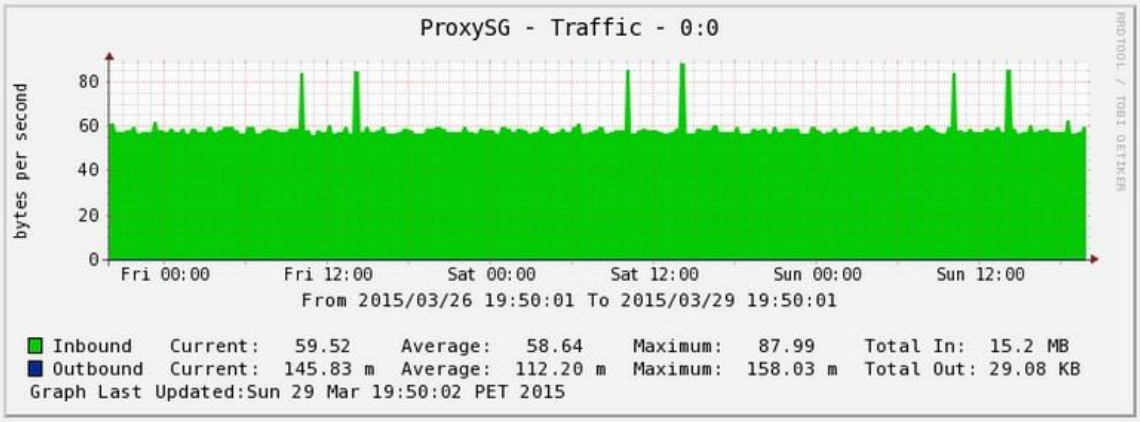


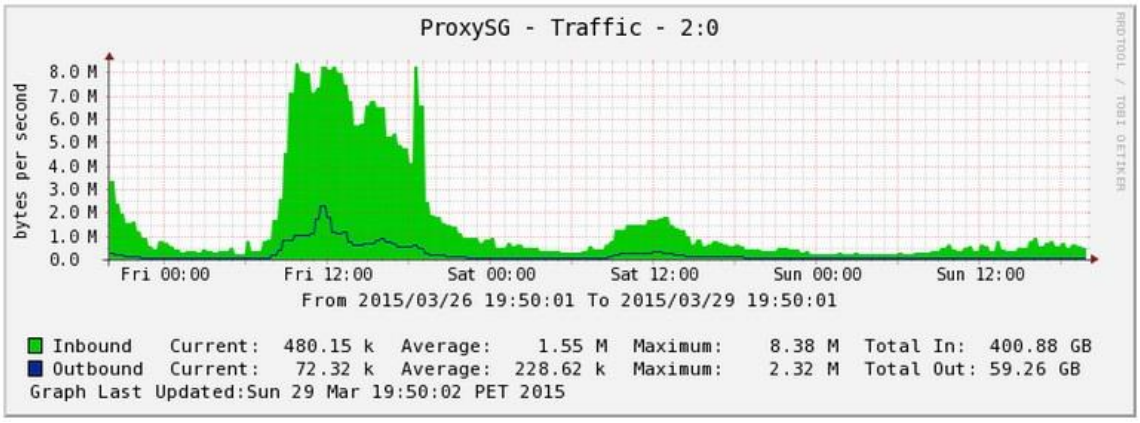



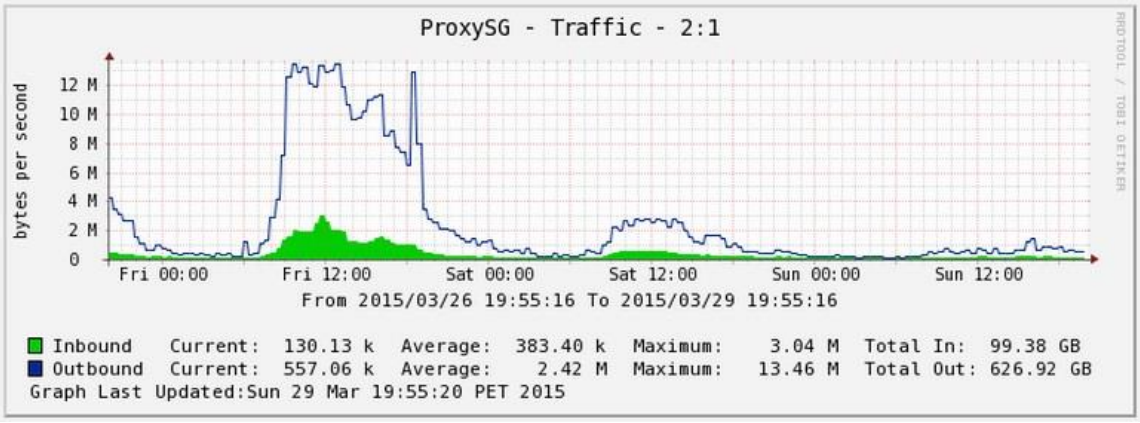



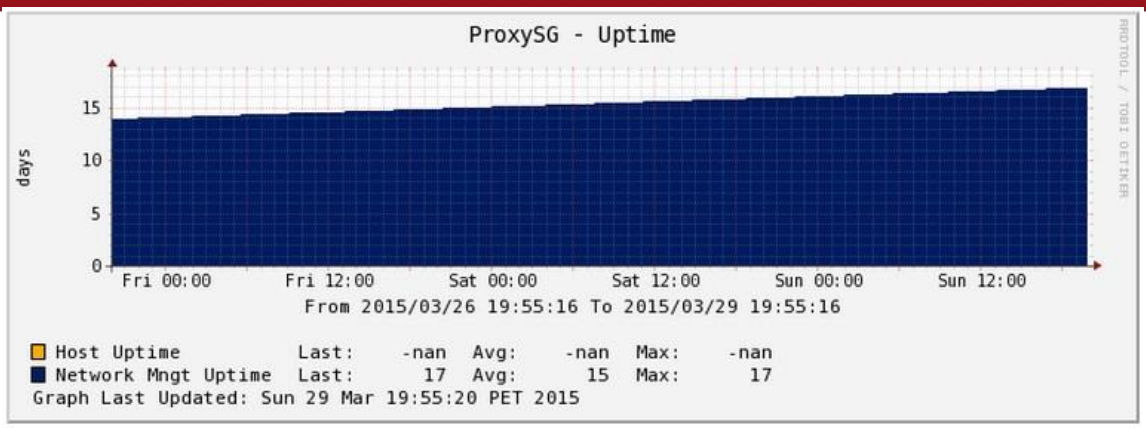





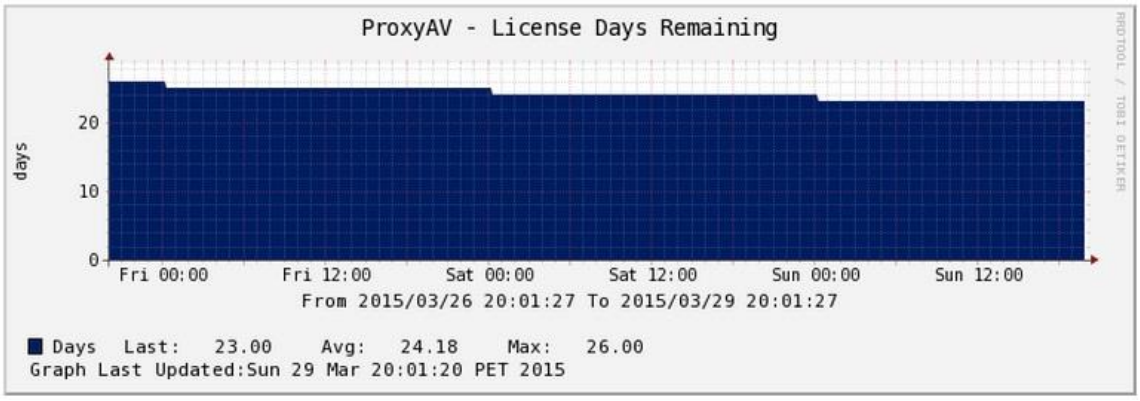



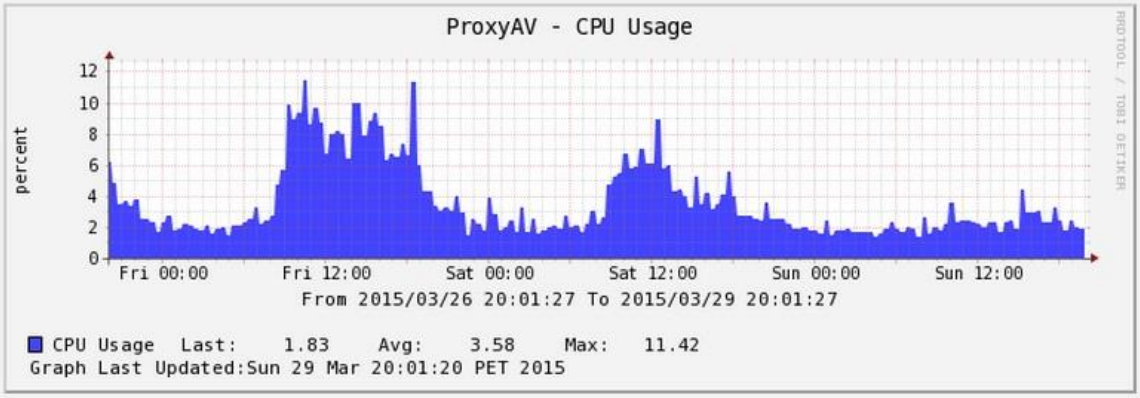



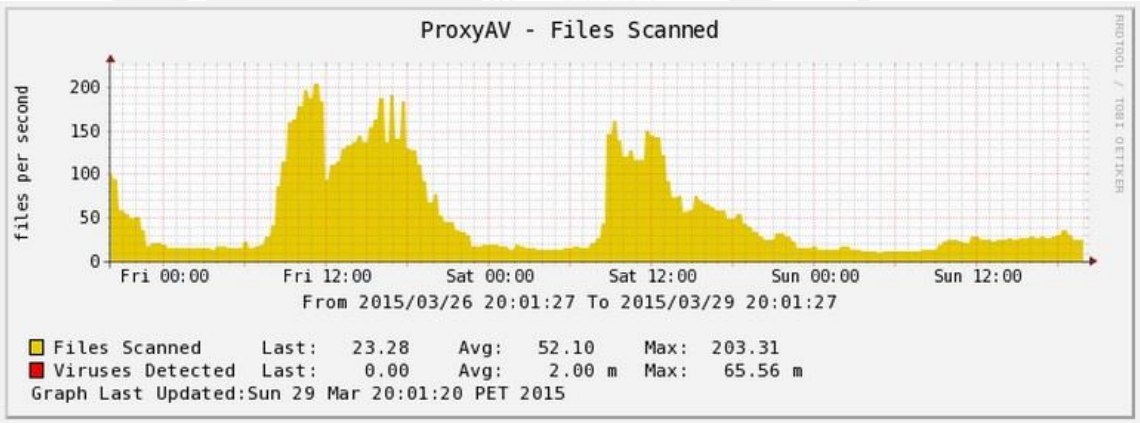



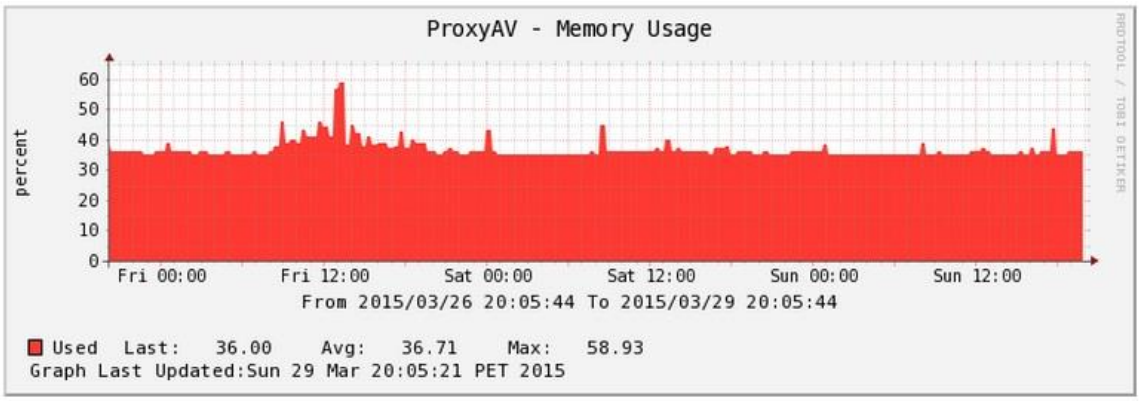


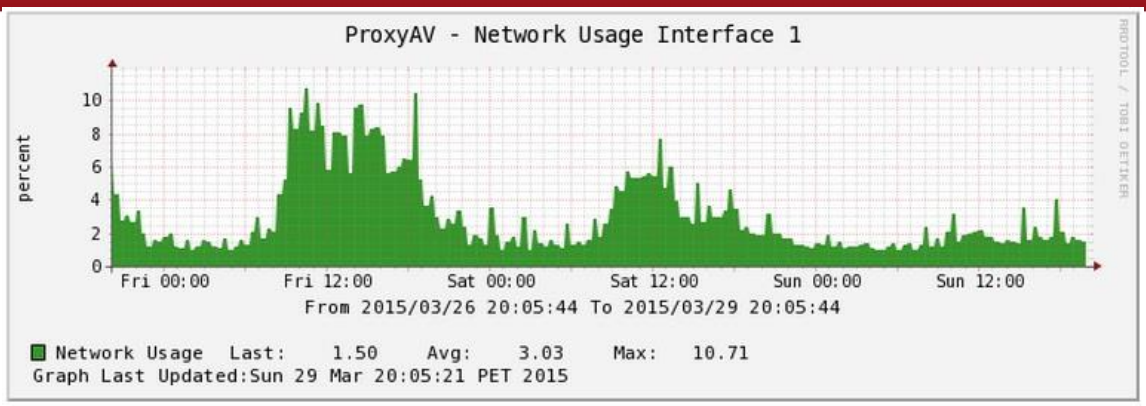
## 2. PROXY ANTIVIRUS BLUECOAT AV1400

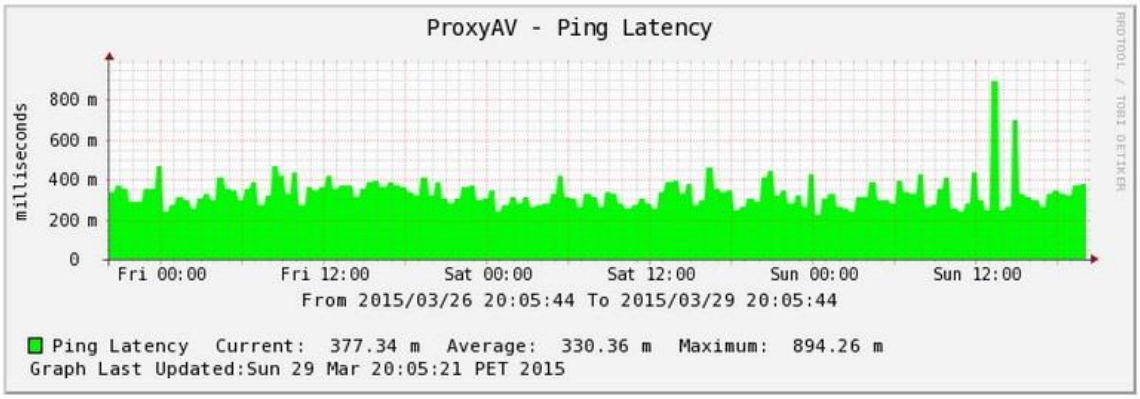


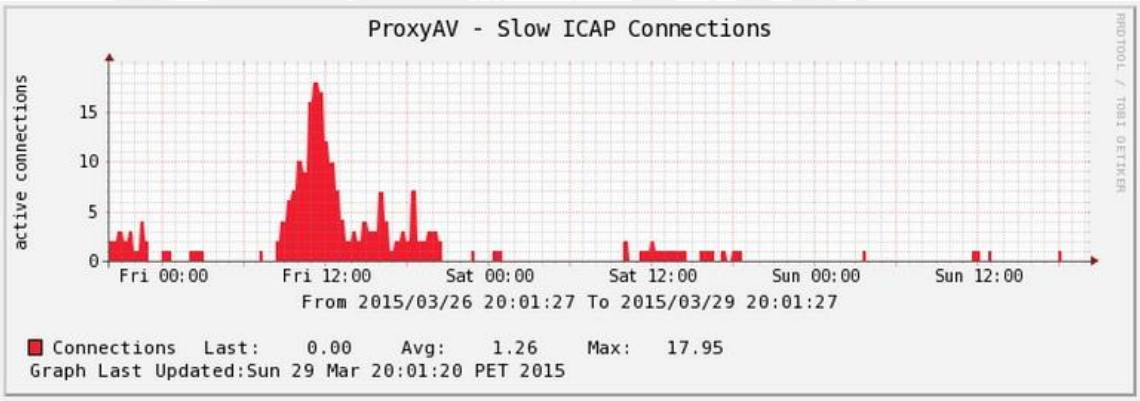



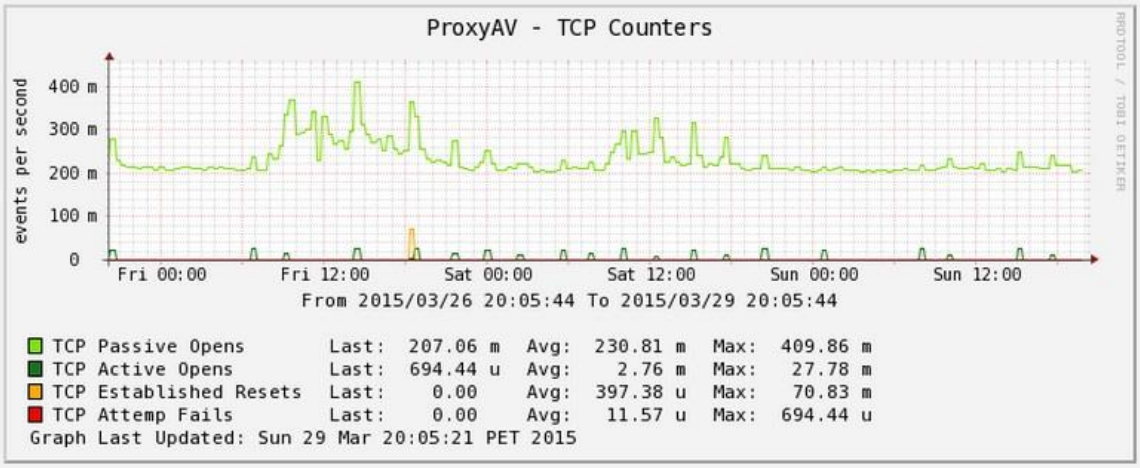



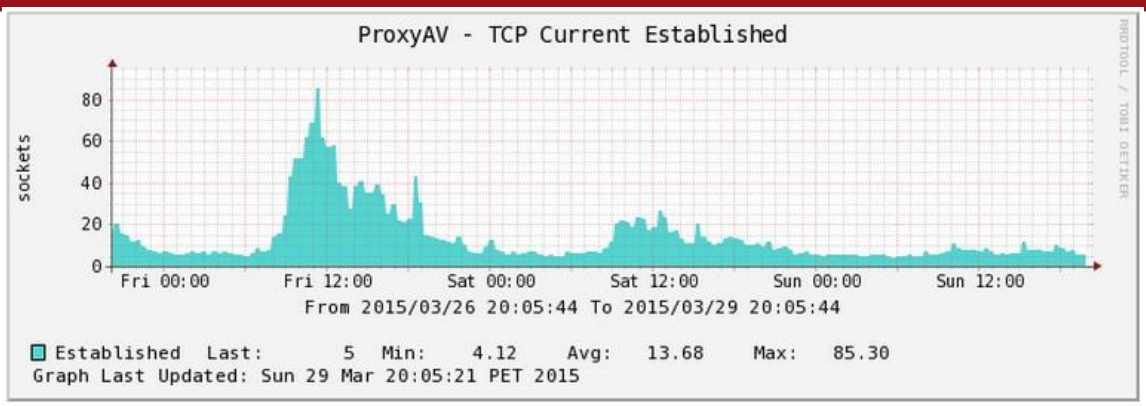



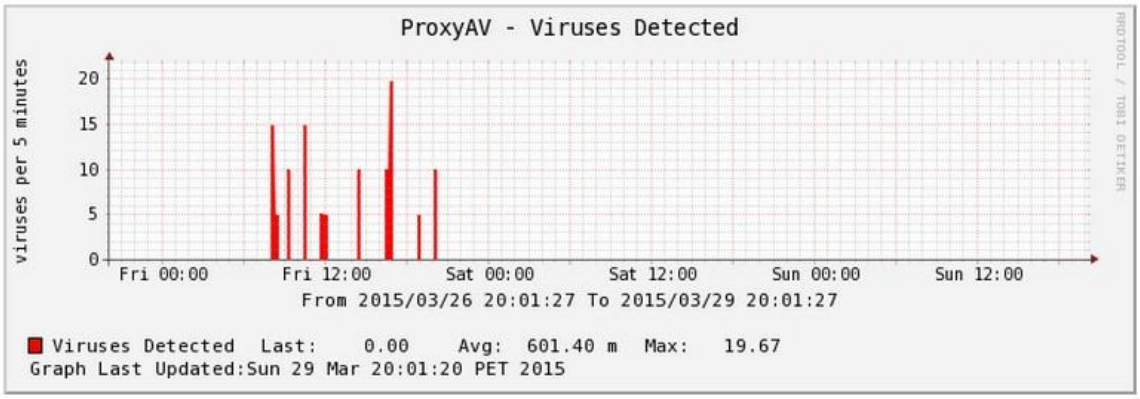


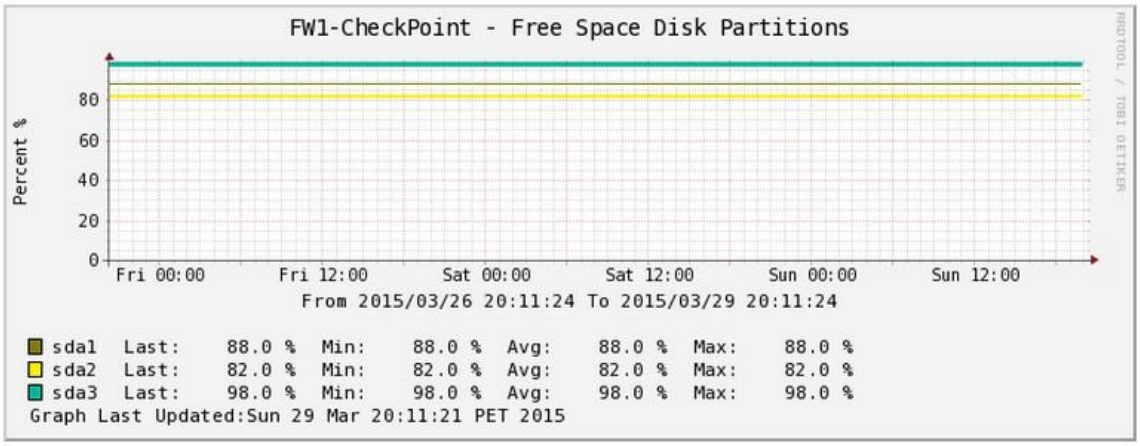



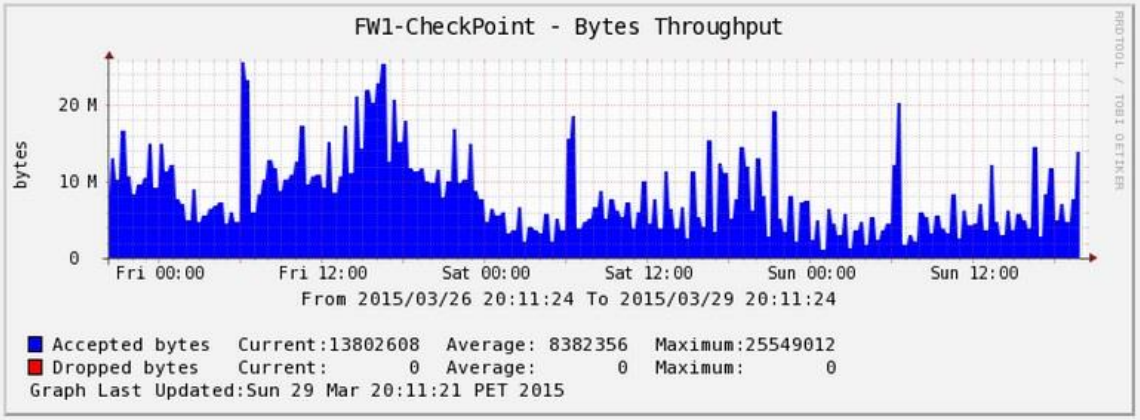



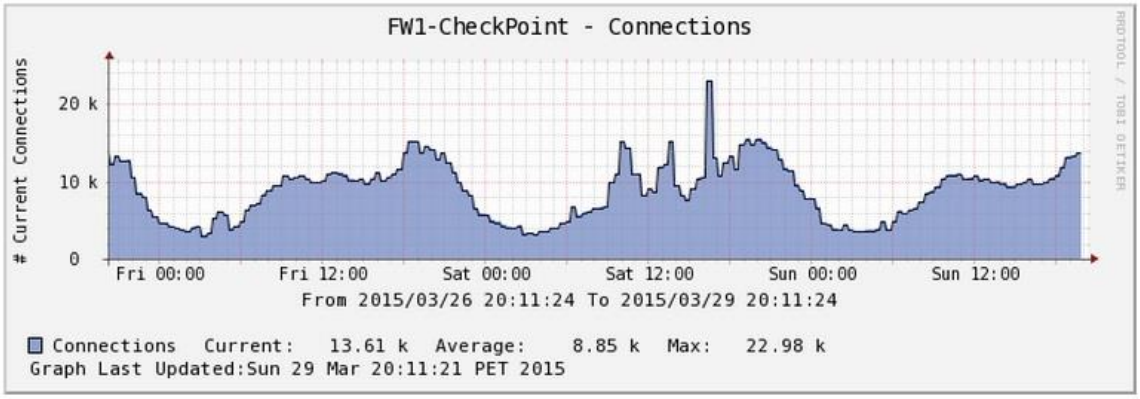


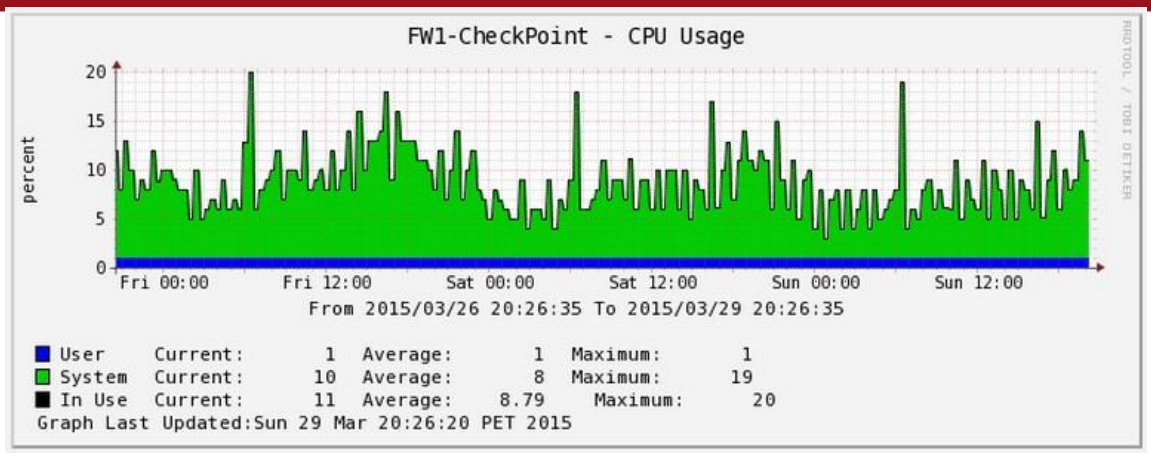



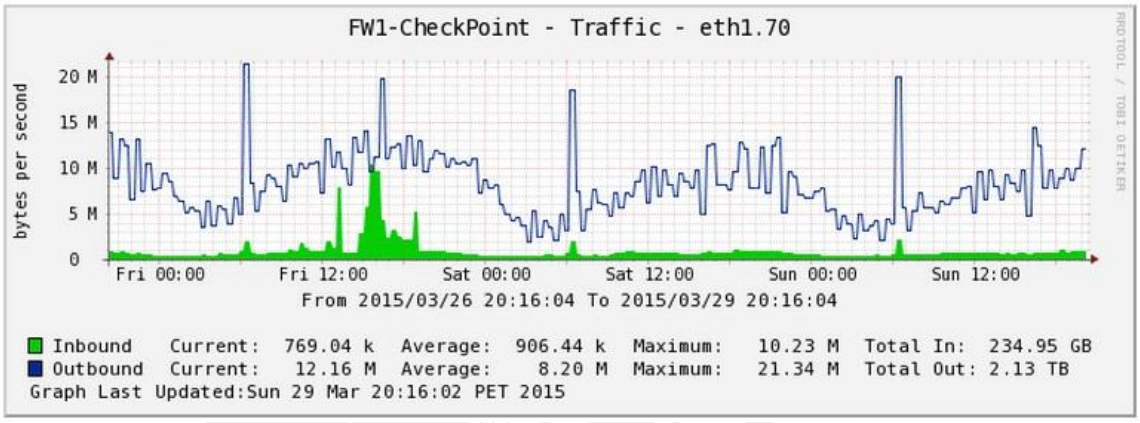
# 3. FIREWALL CHECK POINT 4800

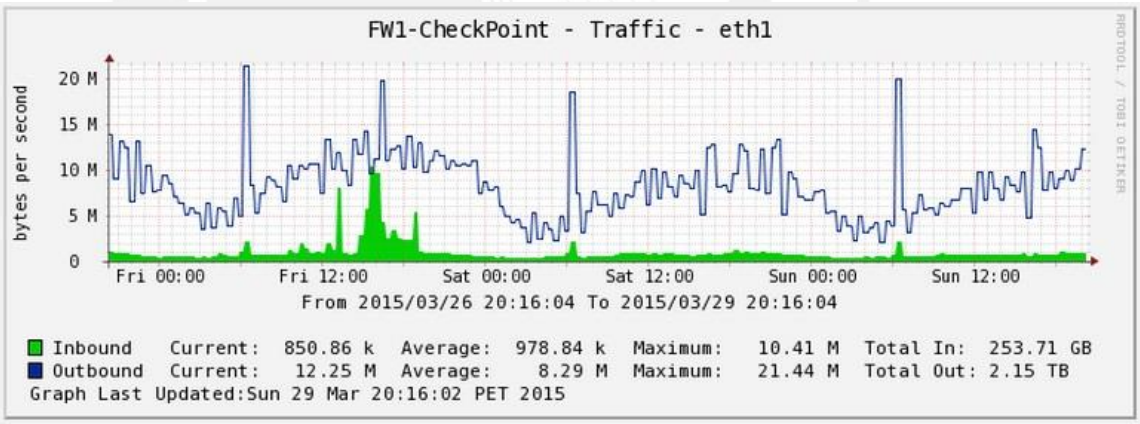


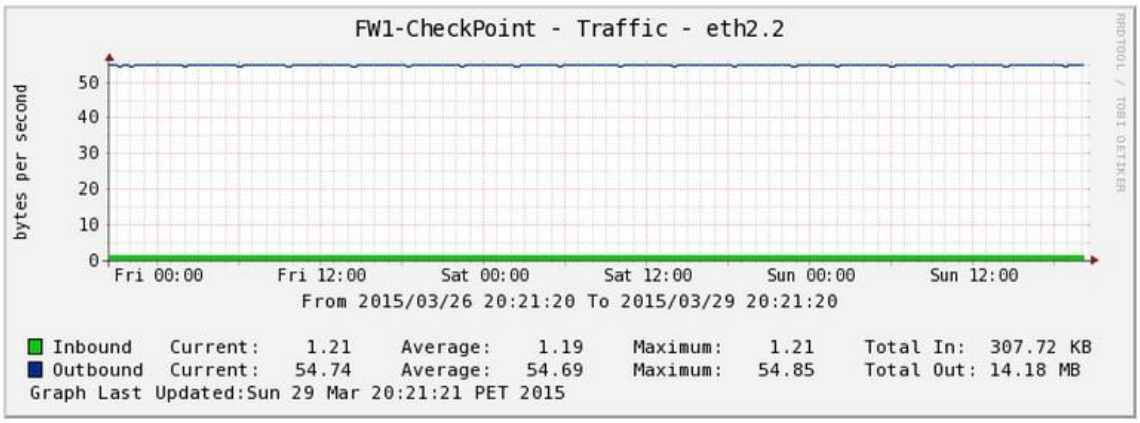



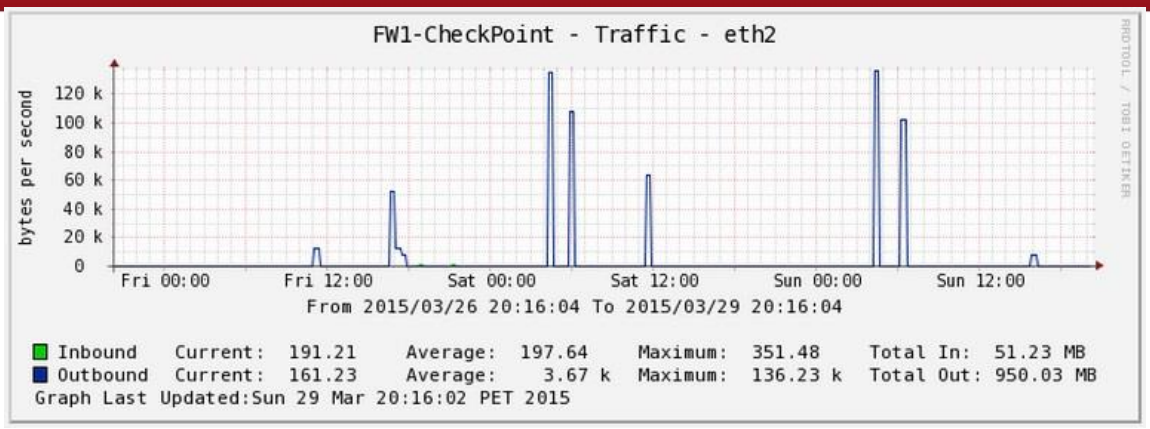



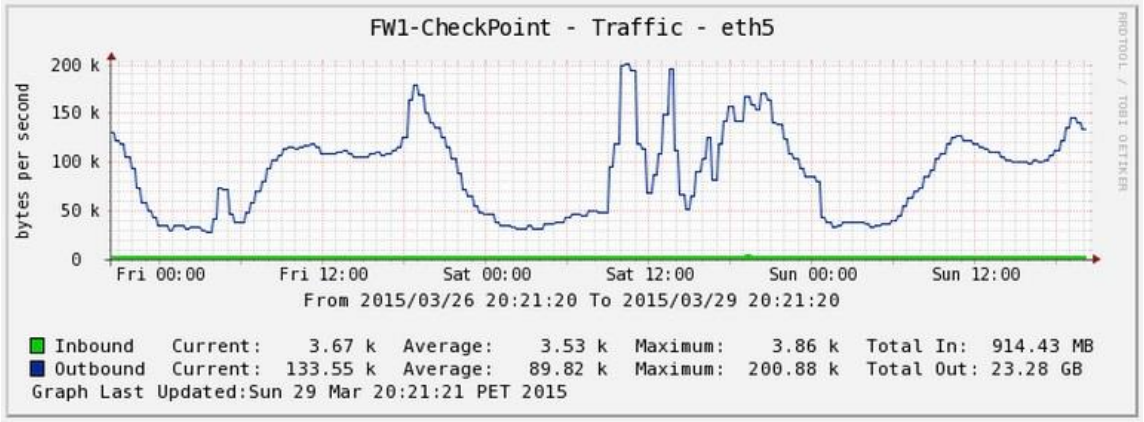


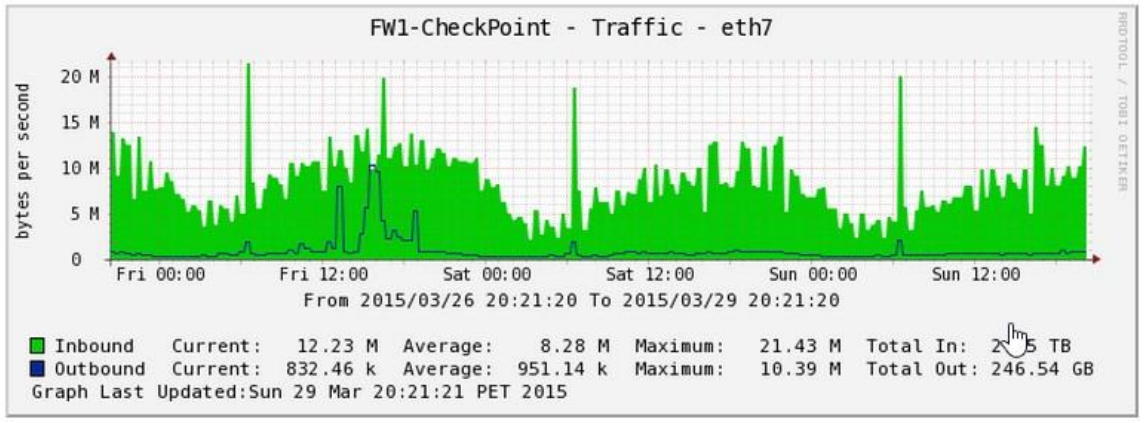



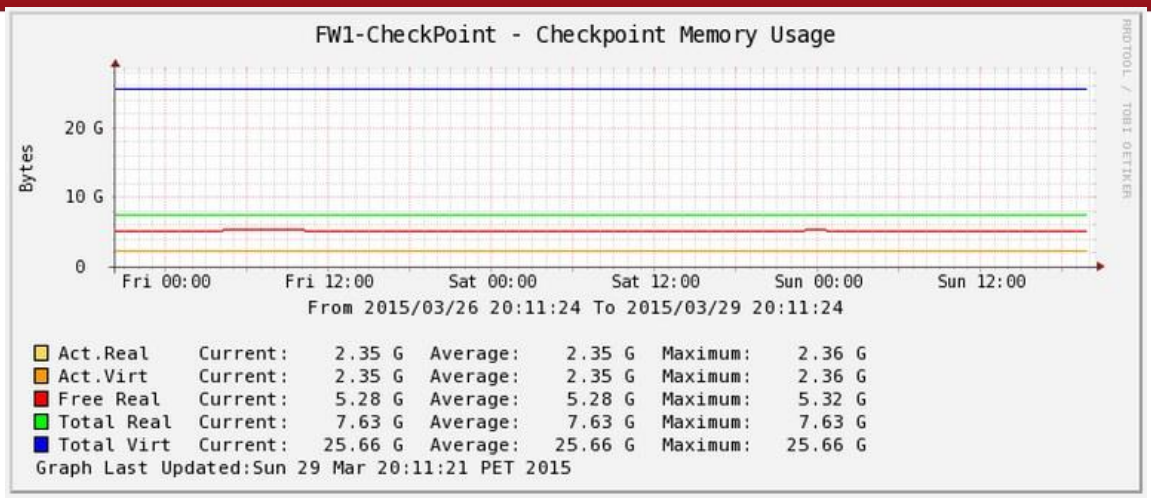



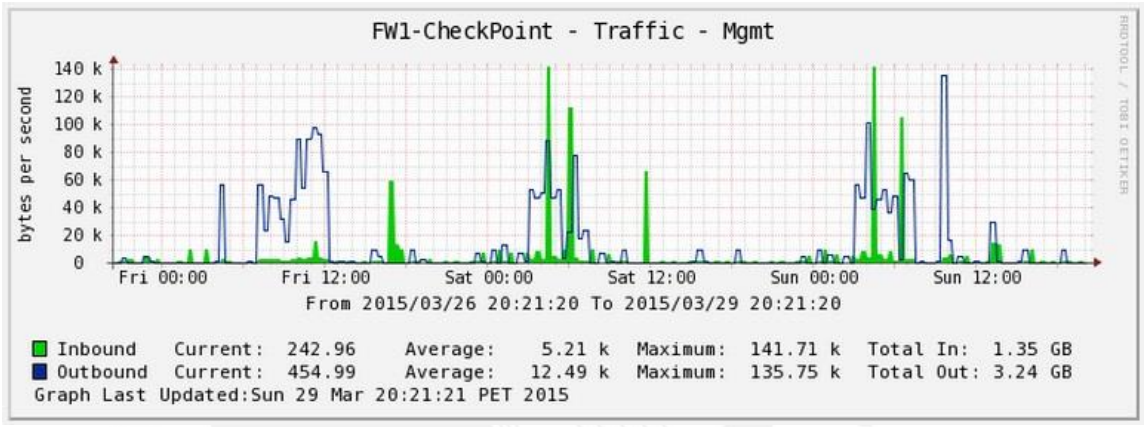



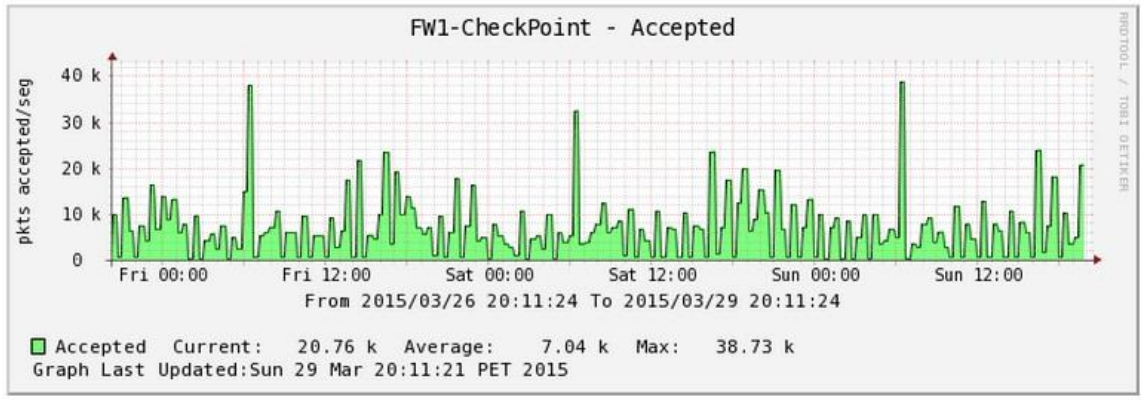


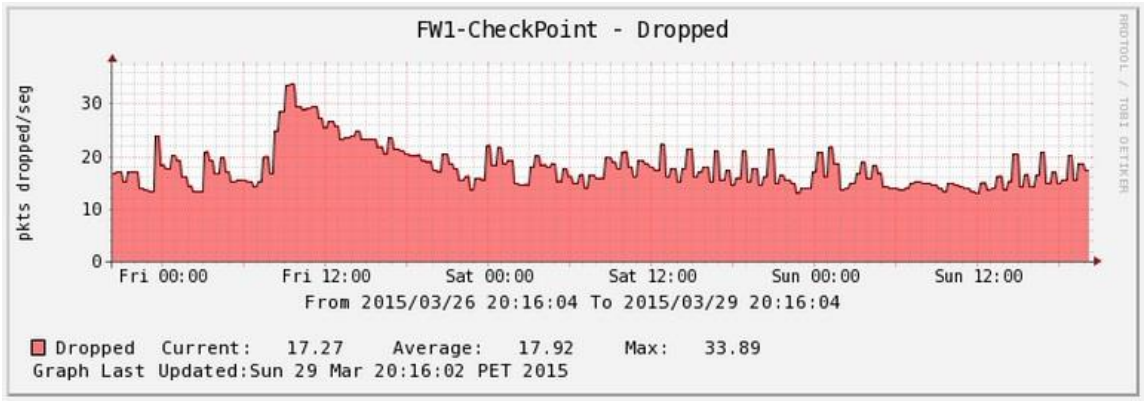



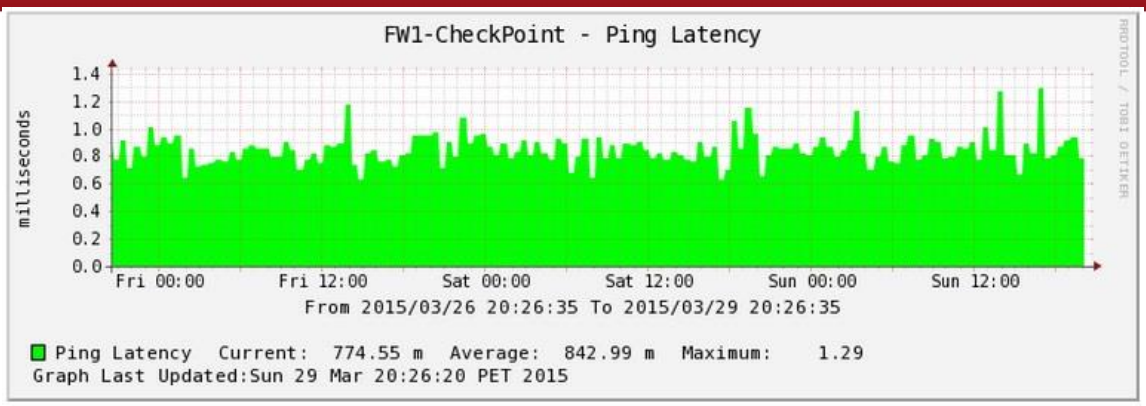



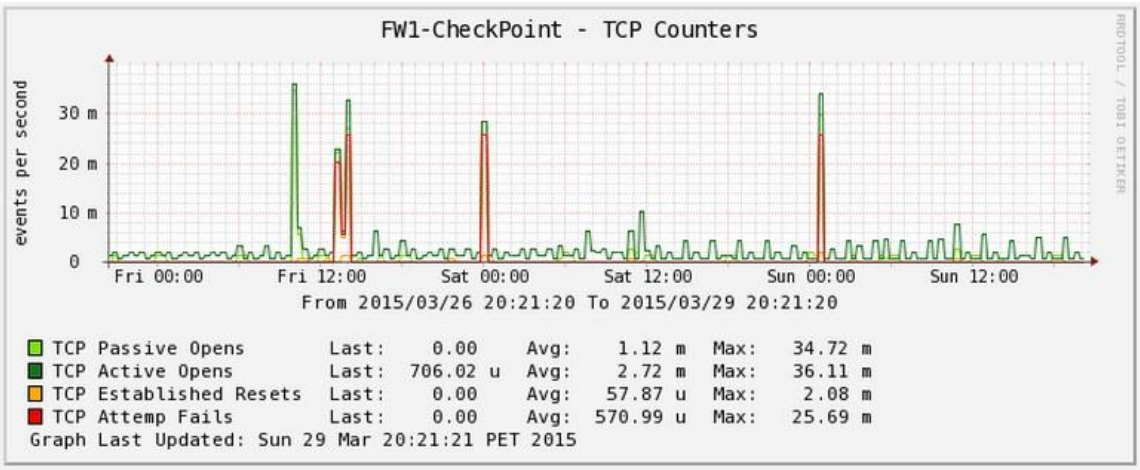


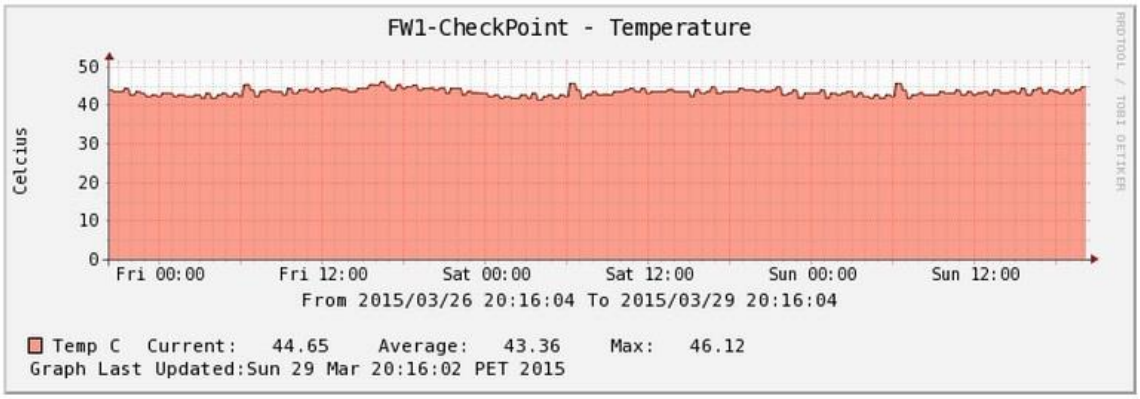



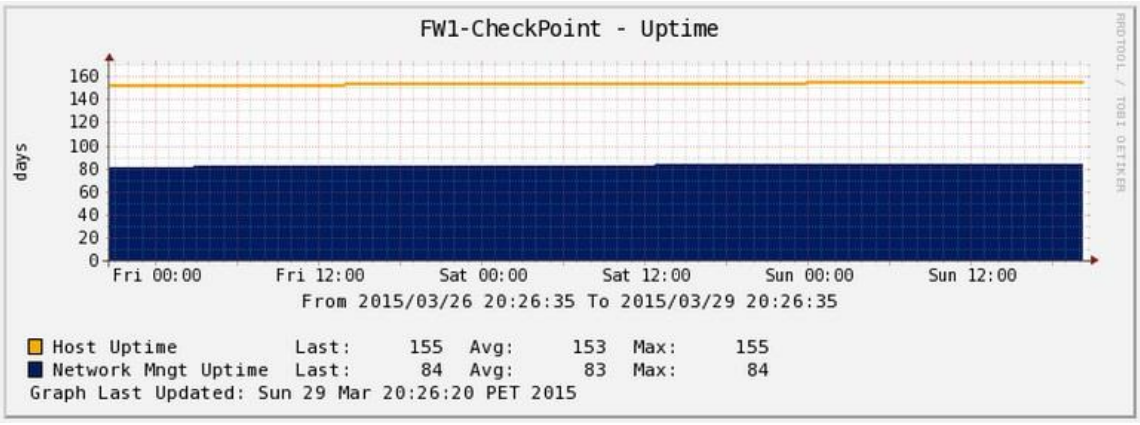



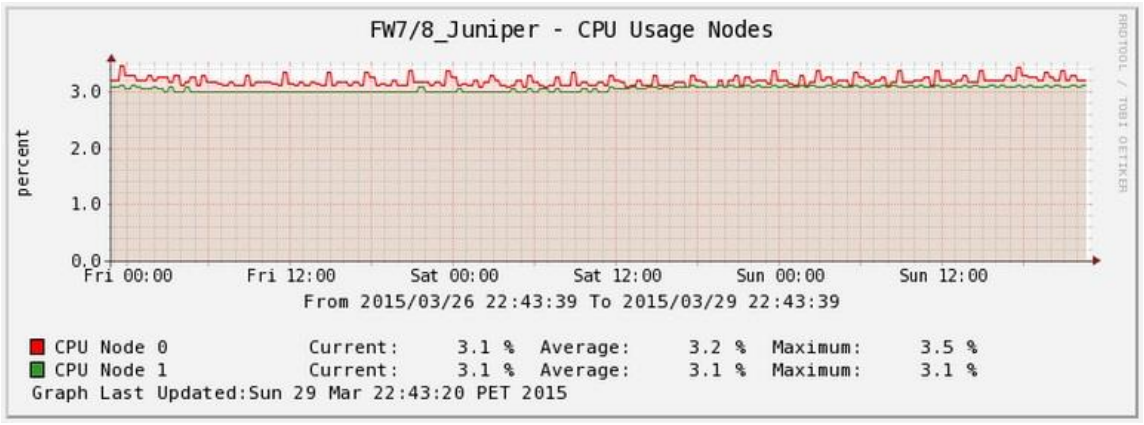


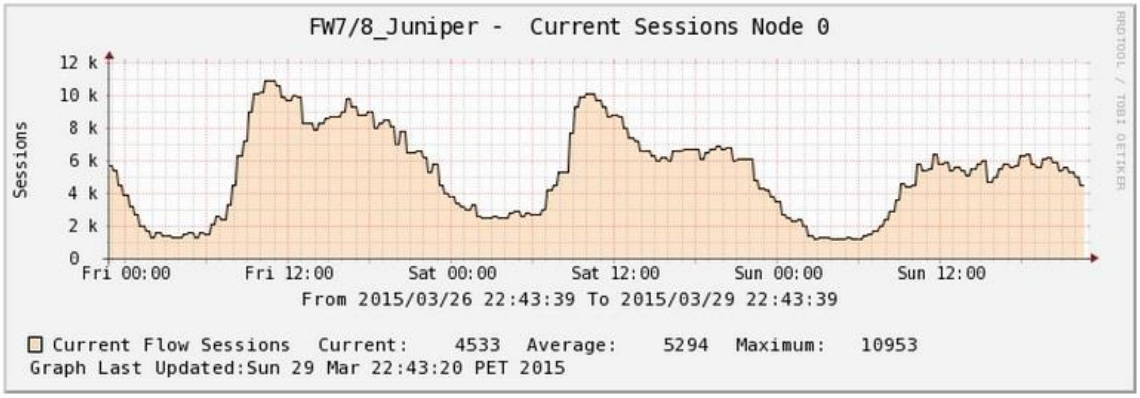



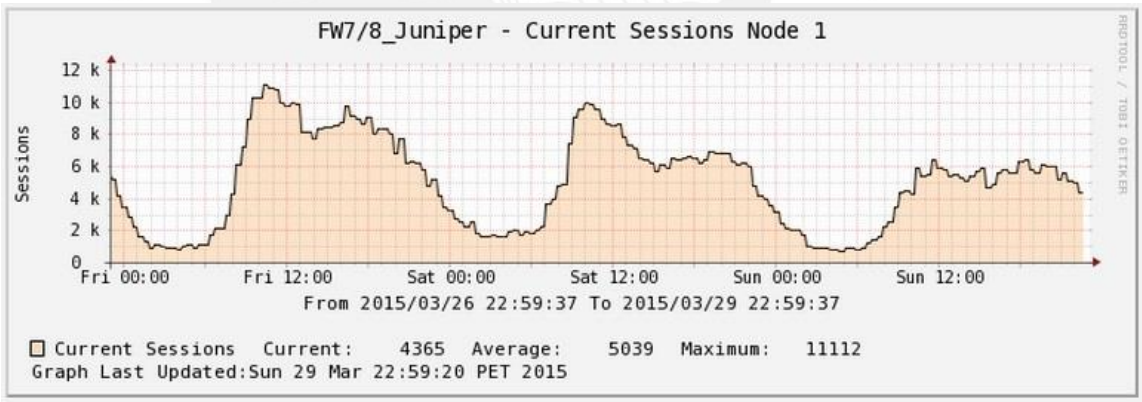



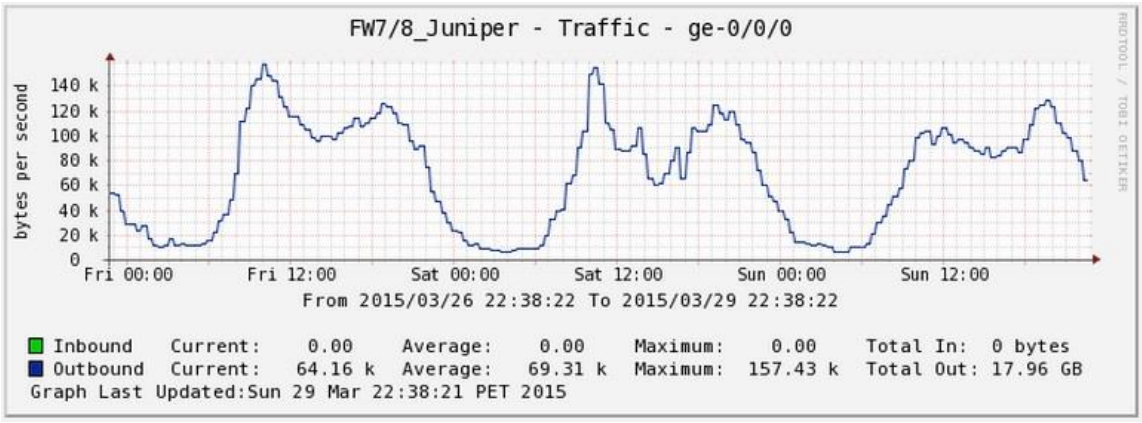



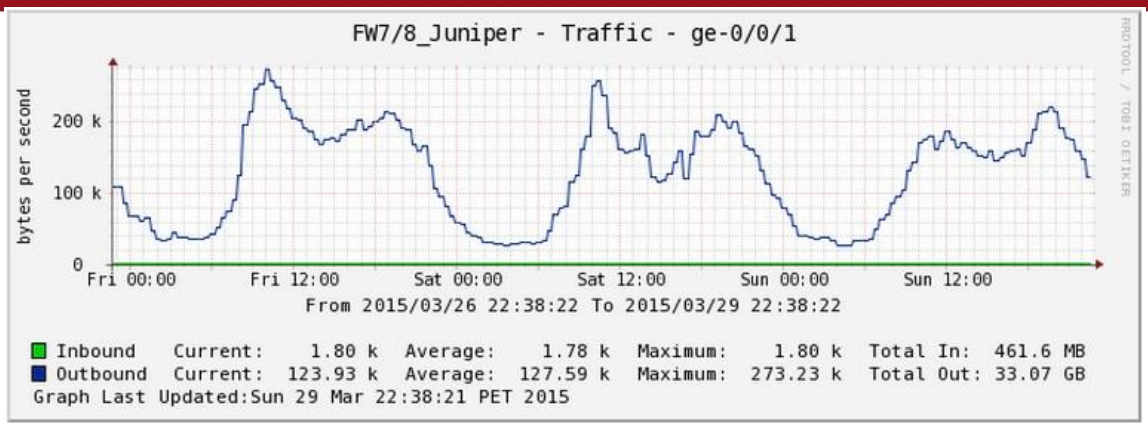
# 4. FIREWALL JUNIPER SRX 3400

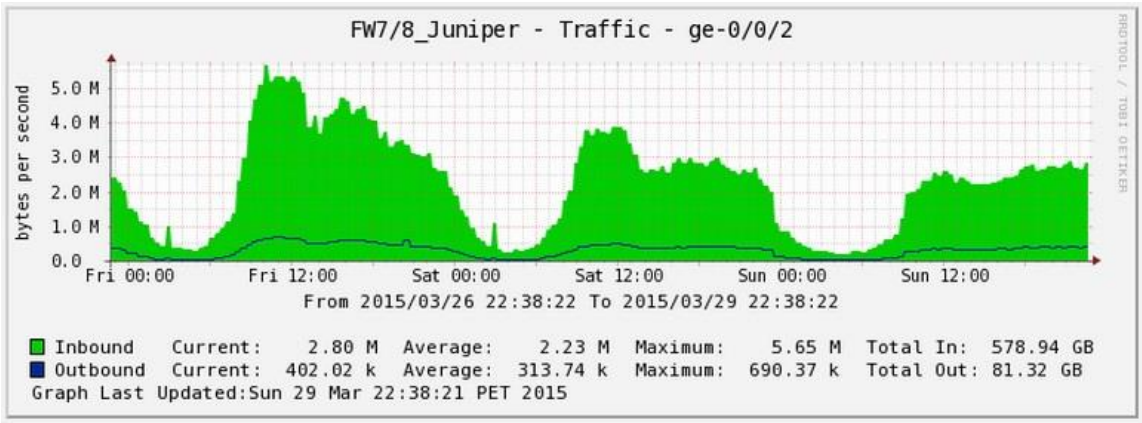


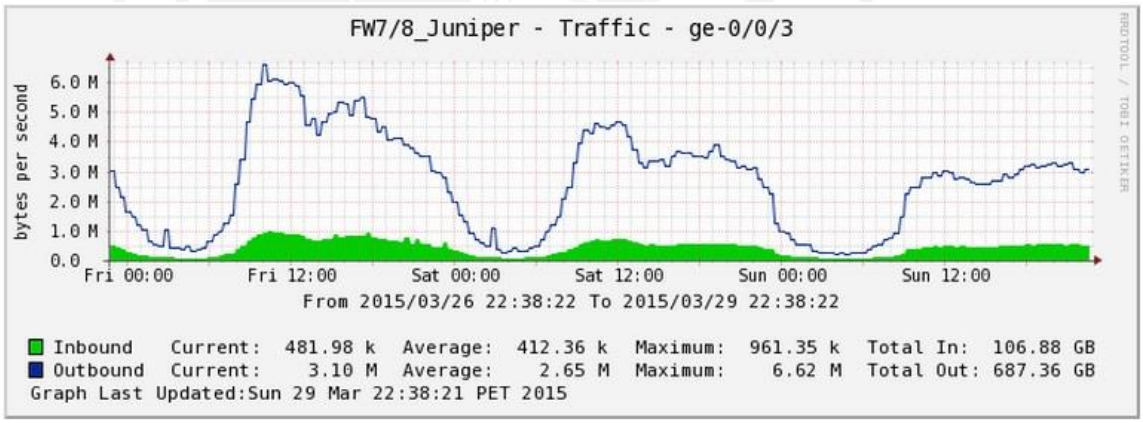



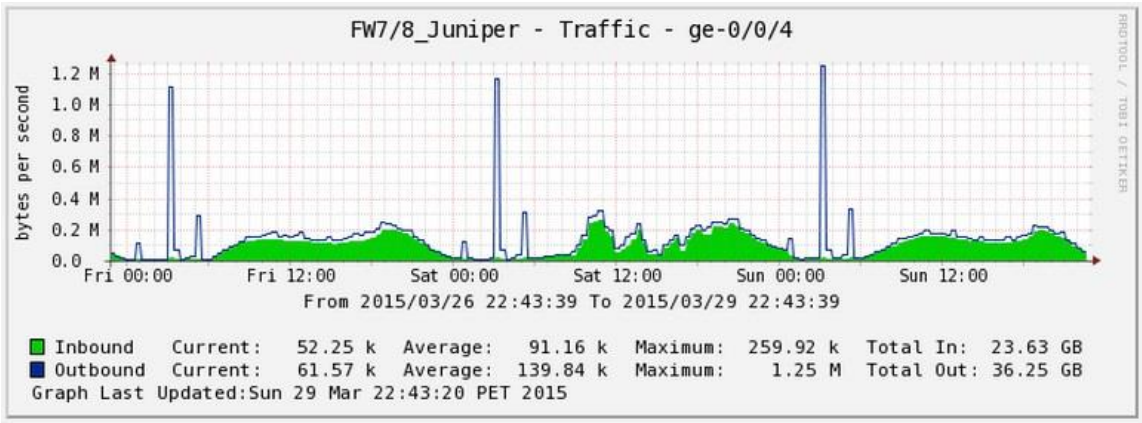



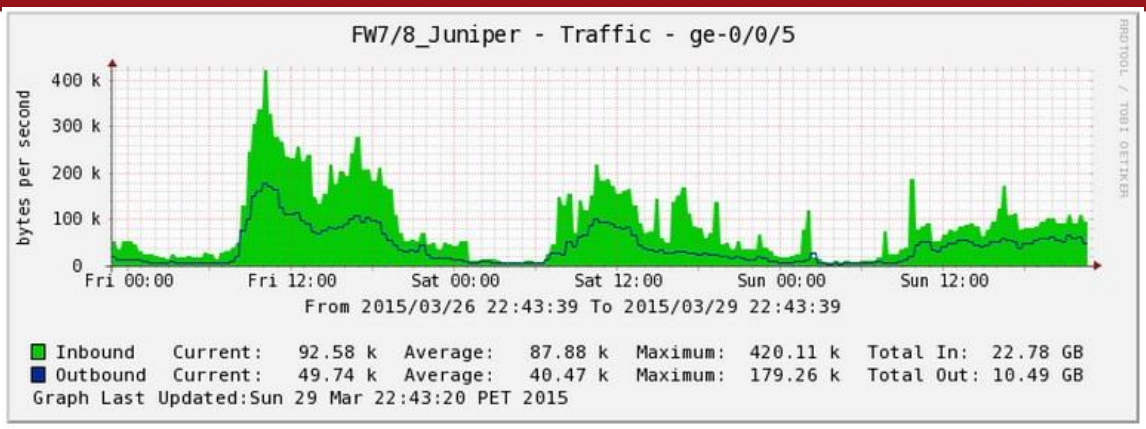



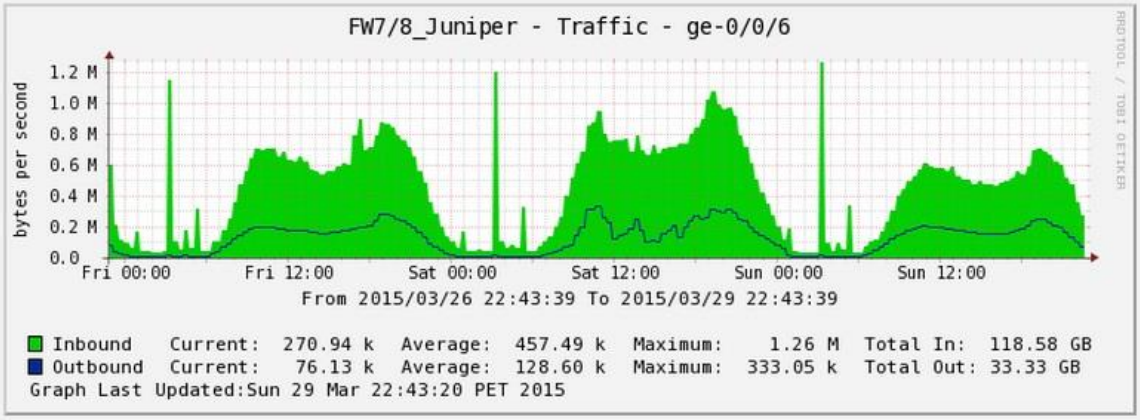


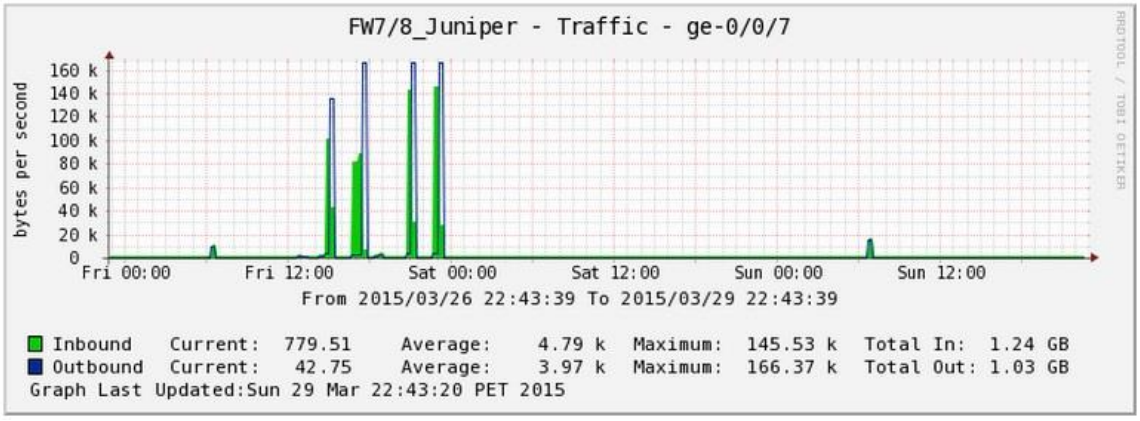



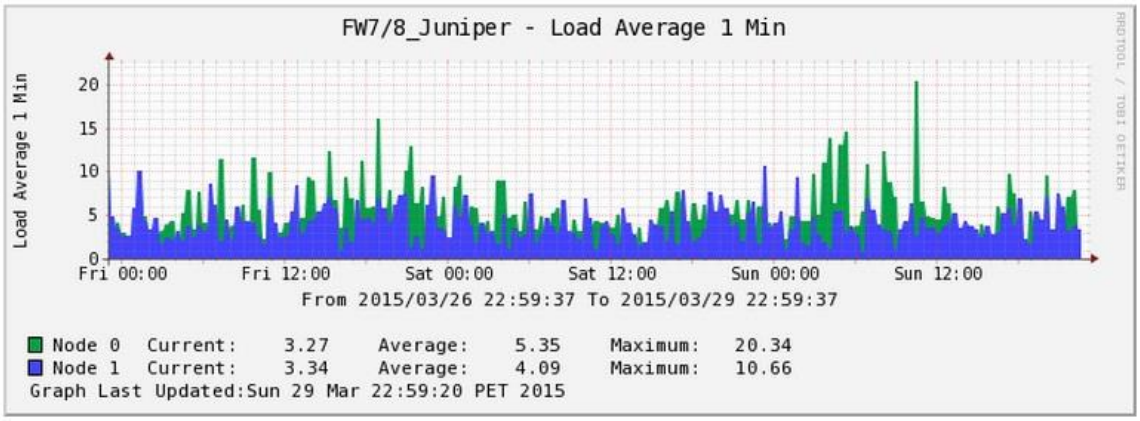



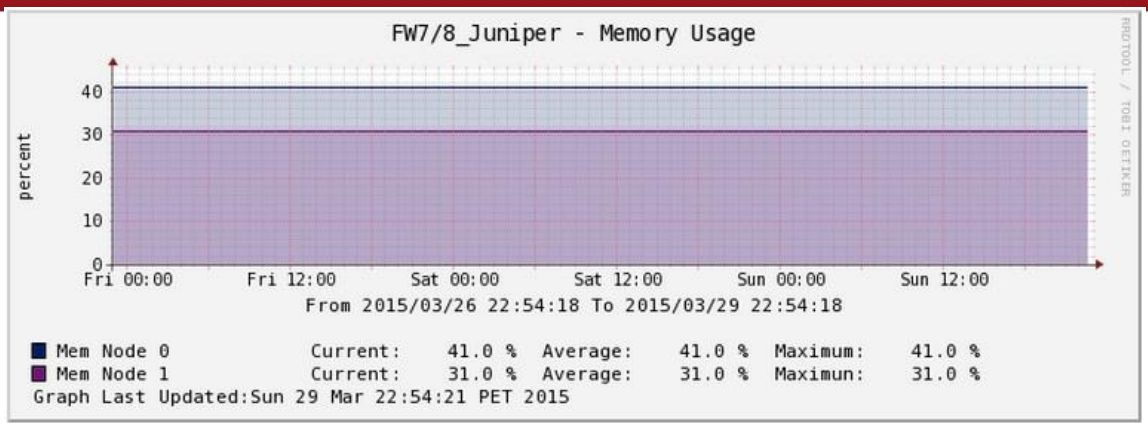



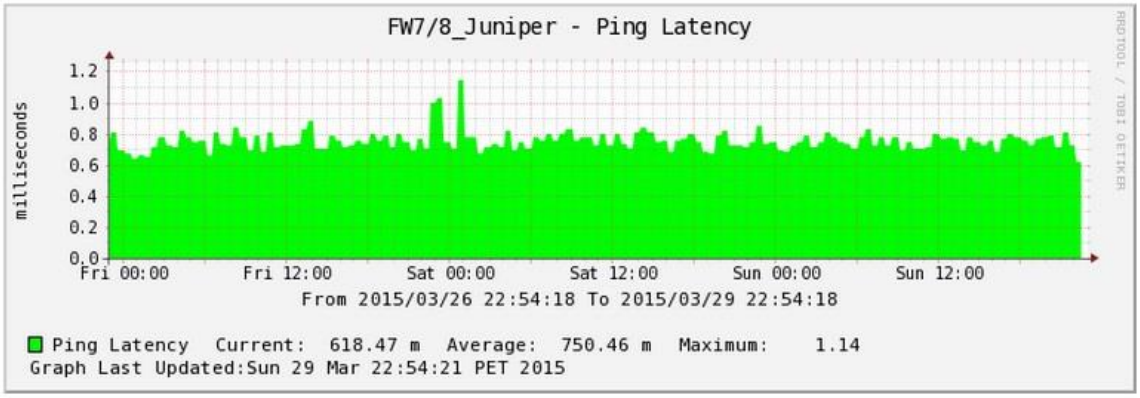


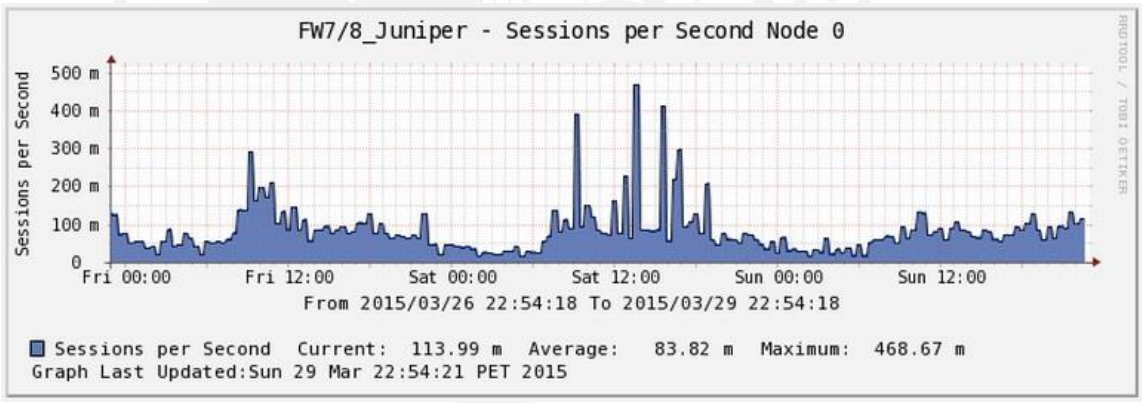



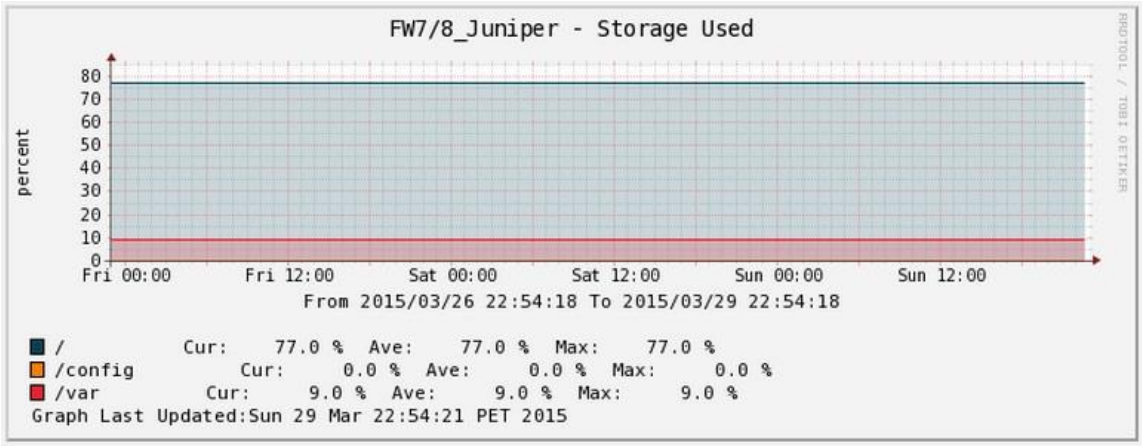



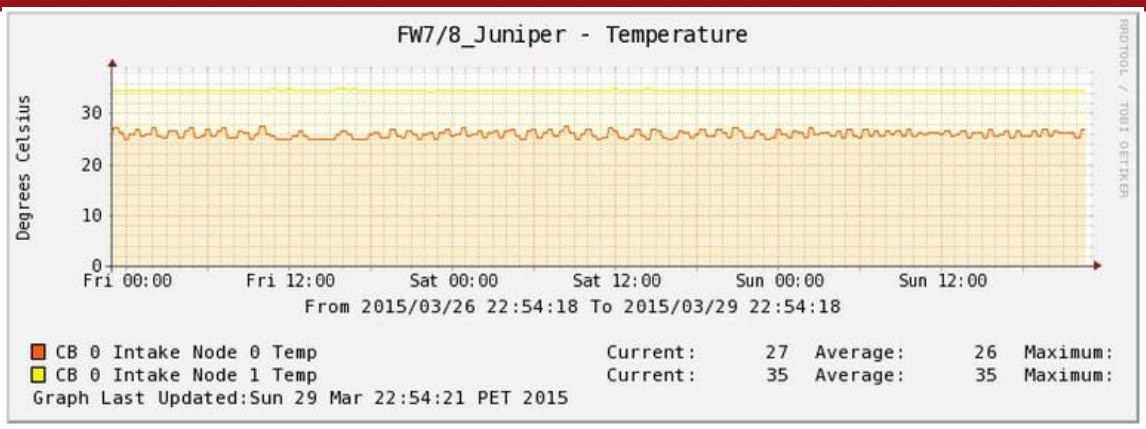



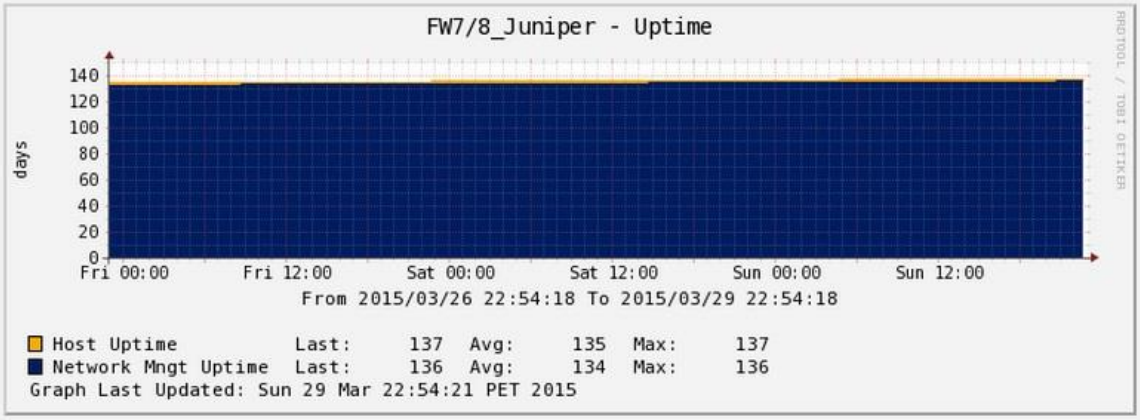




# 5. SENSOR IPS MCAFEE M6050