

INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
RIO GRANDE DO NORTE

# Gerência de Redes de Computadores

## *MRTG*

Prof. Alex Furtunato

[alex.furtunato@ifrn.edu.br](mailto:alex.furtunato@ifrn.edu.br)

# Introdução

- Multi Router Traffic Grapher
- Script criado para monitorar o tráfego em um site, com link de 64kbps, em 1995
- Inicialmente desenvolvida para monitorar o tráfego em links
- Pode utilizar SNMP para ler os dados
- Desenvolvido em PERL
- Gera gráficos HTML
- <http://oss.oetiker.ch/mrtg>

# Instalação

- Debian
  - apt-get install apache2
  - apt-get install mrtg
- Criar pasta em `/var/www/html/mrtg`
  - Armazenamento dos arquivos HTML

# Configuração

- Alguns parâmetros globais :
  - Desabilitar o IPv6:
    - EnableIPv6: no
  - Diretório de gravação dos arquivos gerados:
    - WorkDir: /var/www/html/mrtg/
  - Unidade em bits e sentido de crescimento do gráfico:
    - Options[\_]: growright,bits
  - Linguagem:
    - Language: brazilian
- Arquivo pode ser gerado automaticamente com script abaixo:

```
# cfmaker --global "WorkDir:  
/var/www/html/mrtg/"  
--global "Options[_]: growright,bits" --global  
"Language: brazilian" public@localhost >  
/etc/mrtg.cfg
```

# Execução inicial

```
# env LANG=C /usr/bin/mrtg /etc/mrtg.cfg
```

- No início gera algumas erros, devido a algumas operações de deleção de arquivos que não existem ainda. É necessário executar algumas vezes para que as mensagens de erro deixem de aparecer.
- Os arquivos gerados serão gravados no WorkDir, ou seja, em `/var/www/html/mrtg`
- Obs: Serão gerados os arquivos HTML e as imagens dos gráficos, contudo, não é gerado um arquivo de index para um acesso central aos gráficos.

# Geração da página inicial

- Para gerar o arquivo de índice (index.html), deve-se usar o script a seguir:

```
# /usr/bin/indexmaker
    --output=/var/www/html/mrtg/index.html
    --title="Trafego na Interface"
    --sort=name --enumerate /etc/mrtg.cfg
```

- Gera um arquivo index.html com o resumo das variáveis monitoradas por /etc/mrtg.cfg

# Configuração do CRON

- Para que haja atualização dos arquivos, a intervalos de tempo predeterminados, configura-se o CRON para que a execução do mrtg aconteça sempre.
- No debian, por padrão, essa entrada do CRON já é adicionado
- Em `/etc/cron.d/mrtg`

```
*/5 * * * * root    if [ -x /usr/bin/mrtg ] && [
-r /etc/mrtg.cfg ]; then env LANG=C /usr/bin/mrtg
/etc/mrtg.cfg >> /var/log/mrtg/mrtg.log 2>&1; fi
```
- Com isso, o comando será executado a cada 5 minutos

# Configurando como Daemon

- Adiciona a opção no arquivo de configuração:

```
RunAsDaemon: Yes
```

```
Interval: 10
```

- O script MRTG tem que ser disparado uma primeira vez
- Cria um script de inicialização /etc/init.d/mrtg:

```
Veja exemplo no próximo slide
```

- Modifica atributos de execução do script

```
chmod +x /etc/init.d/mrtg
```

- Cria os links para os Runleves apropriados

```
update-rc.d mrtg defaults
```



# Script de inicialização

```
#!/bin/sh

# Zabbix daemon start/stop script.

# Copyright (C) 2001-2016 Zabbix SAI

NAME=mrtg

DAEMON=/usr/bin/${NAME}

PID=/tmp/${NAME}.pid

DAEMON_ARGS="/etc/mrtg.cfg --pid-file=$PID"

DESC="MRTG daemon"

test -f $DAEMON || exit 0

case "$1" in

  start)

    echo "Starting $DESC: $NAME"

    env LANG=C start-stop-daemon --start --oknodo --pidfile $PID --exec $DAEMON -- $DAEMON_ARGS

    ;;

  stop)

    echo "Stopping $DESC: $NAME"

    start-stop-daemon --stop --quiet --pidfile $PID --retry=TERM/10/KILL/5 && exit 0

    start-stop-daemon --stop --oknodo --exec $DAEMON --name $NAME --retry=TERM/10/KILL/5

    ;;

  restart|force-reload)

    $0 stop

    $0 start

    ;;

  *)

    N=/etc/init.d/${NAME}

    echo "Usage: $N {start|stop|restart|force-reload}" >&2

    exit 1

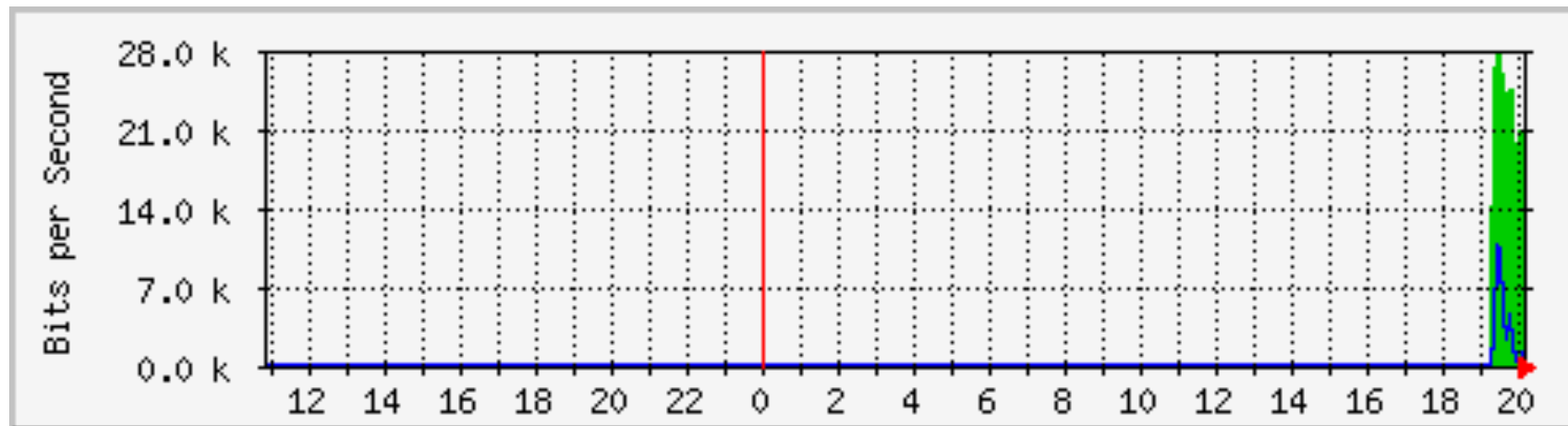
    ;;

esac

exit 0
```

# Visualização dos gráficos

- Acesso pelo navegador web no endereço:
  - <http://ipdamaquina/mrtg>



# Variáveis de monitoramento

- LoadMIBS: Carregar arquivo de MIBs
- Target: O que deve ser monitorado. Deve ser identificado por um nome único.
  - Ex:
    - Target[nome]: port:community@host
    - Target[nome]: OID1&OID2:community@host
  - Pode-se utilizar também um script que retorne até 04 linhas com o resultado a ser plotado no gráfico ou a string 'UNKNOWN':
    - Target[nome]: `script\_a\_ser\_executado`
    - As 04 linhas devem conter, respectivamente, valor da variável 1, valor da variável 2, string com tempo de uptime e string com nome
- MaxBytes: Número máximo que devem ser retornados pela variáveis
- Title: Título da página
- PageTop: Código HTML no topo da página

# Adicionar Variável por Shell Script

- Qualquer variável pode ser colhida através de Shell Script
- Por Exemplo: Uso de Disco

```
#!/bin/sh
DATA=`df -k /dev/sda1 | tail -1`
VAL1=`echo $DATA | cut -f3 -d" "`
VAL2=`echo $DATA | cut -f4 -d" "`
echo $VAL1
echo $VAL2
echo 0
echo 0
```

# Adicionar Variável por Shell Script

- Adiciona a nova variável no arquivo de configuração do MRTG

```
Title[disk_usage]: Disk usage (GB)
PageTop[disk_usage]: <H1>Disk usage (GB)</H1>
Options[disk_usage]: nobanner,gauge,noinfo,nopercent
Target[disk_usage]: `/opt/mrtg/disk.sh`
kMG[disk_usage]: G
MaxBytes[disk_usage]: 35410400
YLegend[disk_usage]: Disk total/used
Unscaled[disk_usage]: ymwd
LegendI[disk_usage]:
LegendO[disk_usage]:
Suppress[disk_usage]: y
```

# Exercício

- Adicione mais dois arquivos de configuração para monitorar:
  - Uso de CPU da máquina
  - Uso de memória da máquina