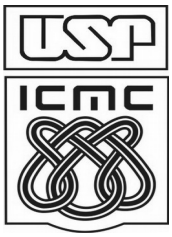


Universidade de São Paulo
Instituto de Ciências Matemáticas e de Computação
www.icmc.usp.br

SISTEMAS PARA GERENCIAMENTO DE REDES

Trabalho Prático desenvolvido para a disciplina de
Administração e Gerenciamento de Redes

pelo alunos:

Ubiratan Soares (*5634292*) – ubiratan.f.soares@gmail.com
Paulo Ricardo Chagas(*5634816*) - paulo.portacopo@gmail.com
Ulisses Soares (*5377365*) – apblanc@gmail.com

São Carlos, 17 de dezembro de 2009

INTRODUÇÃO AO MRTG

O Multi Router Traffic Grapher, ou simplesmente MRTG, é uma popular ferramenta para o monitoramento de redes, em especial por ser software livre[1]. Foi desenvolvida por Tobias Oetiker e Dave Rand especificamente para o monitoramento de redes, mas com o tempo acabou se tornando também ferramenta para gerar gráficos e estatísticas para outros propósitos. É escrito em linguagem Perl, e está disponível em versões para diversos sistemas operacionais, como Linux, Solaris, Windows, Mac OSX, dentre outros.

Protocolos Utilizados

O MRTG utiliza o SNMP[2] (*Simple Network Management Protocol*) para enviar dois identificadores de objeto para um dispositivo sob monitoramento. Esse disposito deve estar habilitado ao protocolo SNMP, de maneira que os objetos sob requisição serão inspecionados na MIB[3](*Management Information Base*) do mesmo, e então encapsulados e enviados novamente ao agente MRTG, que criará logs sobre os dados e executará as demais funcionalidades, conforme a configuração desejada. O suporte ao SNMP do MRTG é complemante *built-in* na ferramenta, não sendo necessário nenhum pacote externo para adicionar funcionalidades relacionadas ao protocolo.

Adicionalmente, o MRTG possui suporte[4] ao SNMP v2c (*Community-Based*), que é considerada *de facto* o padrão para a segunda versão desse protocolo[2], bem como ao SNMPv3.

Principais Funcionalidades

Algumas das principais funcionalidades[3] oferecidas pelo MRTG são:

- Identificação confiável de interfaces, uma vez que interfaces de roteadores podem ser identificadas por endereço IP, descrição e MAC Adress, em adição ao número da interface;
- Arquivos de log de tamanho constante, que não aumentam de tamanho linearmente com o tempo devido a um bem desenvolvido algoritmo para consolidação de dados;
- Configuração simples, em especial graças ao um conjunto de acessórios nativos que são obtidos junto com a ferramenta;
- Gráficos gerados diretamente no formato PNG, sem as complicações inerentes ao formato GIF;
- Customização das páginas HTML de relatório extremamente simples;
- Suporte a Ipv6;
- Suporte nativo ao RRDtool[5], software para armazenamento e recuperação de séries temporais também desenvolvido por Tomas Oetiker e que apresenta vantagens significativas em desempenho para armazenamento de dados como largura de banda, carga em CPU, temperatura, dentre outros;
- Envio de emails em condições de alarme, dentre outras características.

Scripts e Programação

O MRTG pode ser configurado para rodar adequadamente através de scripts e comandos, e analisar a saída dessa execução para gerar relatórios e gráficos. O site oficial do MRTG (http://oss.oetiker.ch/mrtg) contém dezenas de scripts já validados, com funcionalidades que abragem o monitoramento de servidores de bancos de dados SQL, regras de firewall, velocidade de rotação de ventoinhas em gabinetes, dentre diversas outras. Esses scripts são em geral escritos em Perl, como o exemplo a seguir, que testa a perda de pacotes via ping:

```
#!/usr/bin/perl -w
#round-trip min/avg/max/mdev = 10.614/25.990/73.134/23.708 ms#
use strict;
my ($min, $max, $values);
open (PING, "/bin/ping -qc5 213.51.100.1 |")
or die "can't find ping: $!\n";
while (<PING>) {
      next unless m/^round/o;
      ($values) = (split(' ',$_))[3];
      ($min, $max) = (split("/", $values))[0,2];
      ($min) = int($min);
      ($max) = int($max);
      #print "$min, $max\n";
}
#open MRTG, "> /var/mrtg/data/ping.data"
#or die "can't open ping/data: $!\n";

print "$min\n$max\n0\nping test";
#print MRTG "$min\n$max\n0\nping test";
#close MRTG;
```

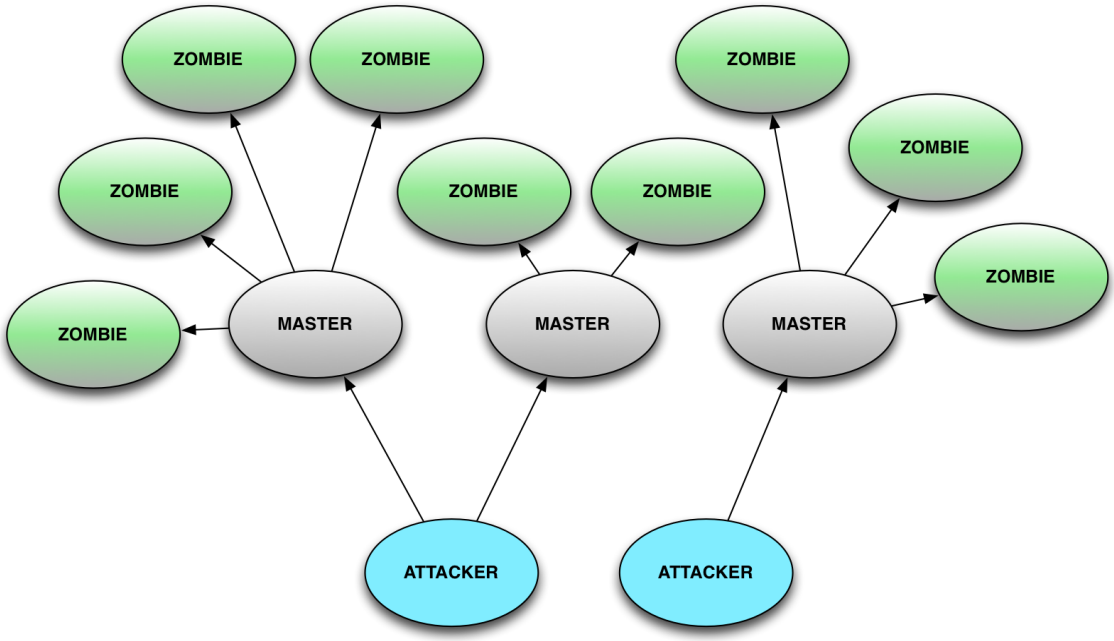
ESTUDO DE CASO - DDOS

A Ferramenta Trinoo

Trinoo é uma ferramenta para executar um ataque por negação de serviço distribuído (*Distributed Denial of Service Attack* – DDOS [6]), sendo comumente caracterizado como a primeira dessas ferramentas conhecida.

O software Trinoo[7] foi encontrado originalmente em seu formato binário em sistemas Solaris 2.x. Uma investigação da rede de computadores infectados levou ao descobrimento do código fonte da ferramenta, por meio de inspeção de caches de disco e arquivos de log[8]. O Trinoo foi utilizado, em seu primeiro registro conhecido, para indisponibilizar a rede da Universidade de Minnesota (EUA) por quase dois dias no ano de 1999.

As máquinas infectadas com o Trinoo se auto organizam segundo uma hierarquia de "mestres" (masters) e "zumbis" (zombies ou deamons).



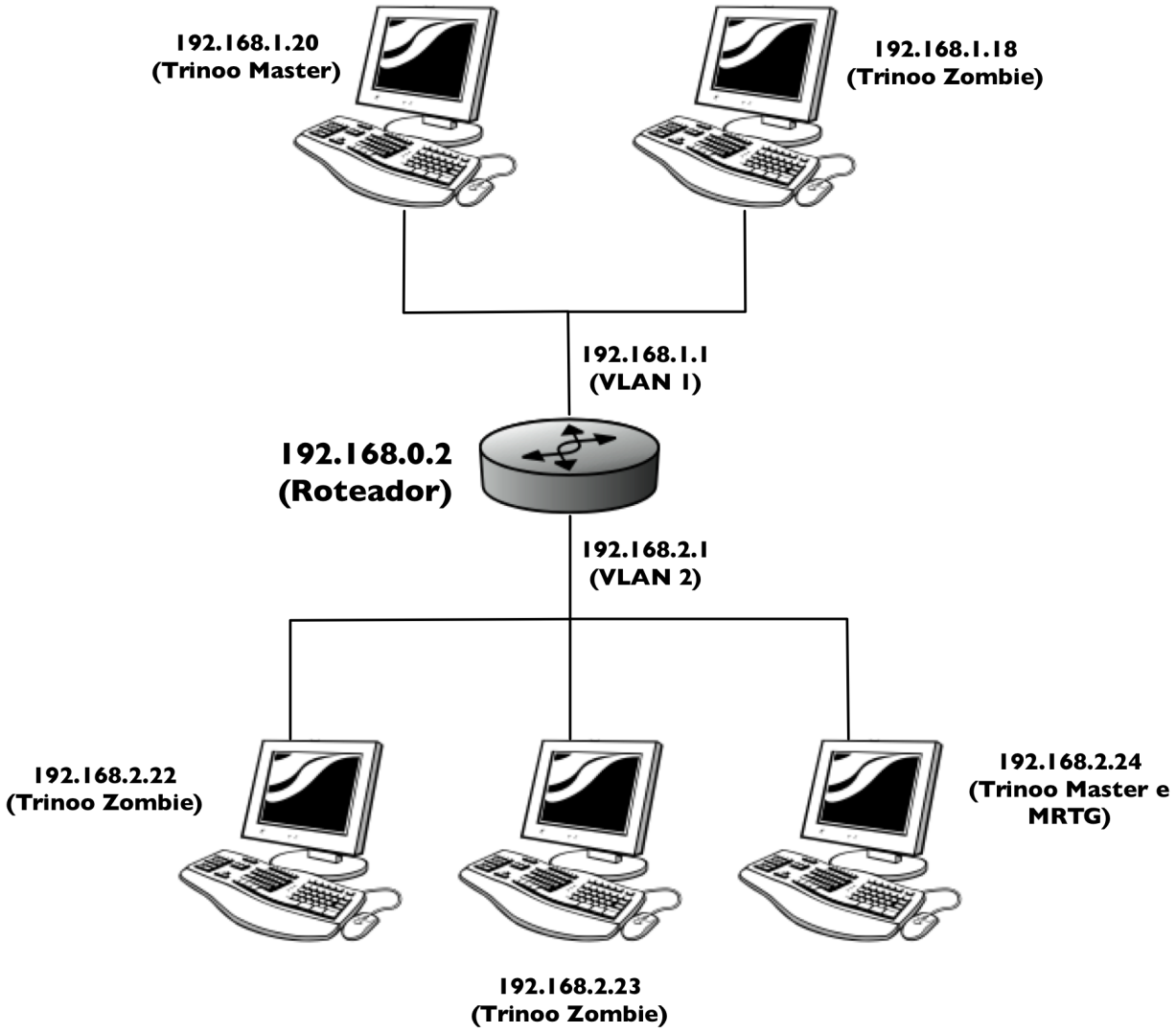
O atacante acessa o programa mestre, que irá comandar instruir as máquinas zumbis sob sua tutela a executar um determinado tipo de ataque, segundo as funcionalidades oferecidade pelo Trinoo. A comunicação entre atacantes, mestres e zumbis se dá pela seguintes portas/protocolos:

Atacante para Mestre	**27665 / TCP**
Mestre para Zumbi	**27444 / UDP**
Zumbi para Mestre	**31335 / UDP**

O acesso ao controle de processos mestres é protegido por senha, bem como a passagem de comandos entre mestres e zumbis. O acesso do atacante a uma máquina mestre é feito por **telnet**. É possível, então, que o atacante conheça a lista de máquinas controladas pelo mestre (*Bcast hosts*) e execute facilmente um ataque DDOS coordenado mediante um script relativamente simples, executado sobre "rede" Trinoo diretamente da sua máquina.

Cenário de Teste

Para a execução do ataque DDOS em ambiente controlado, montamos a seguinte estrutura no laboratório didático:



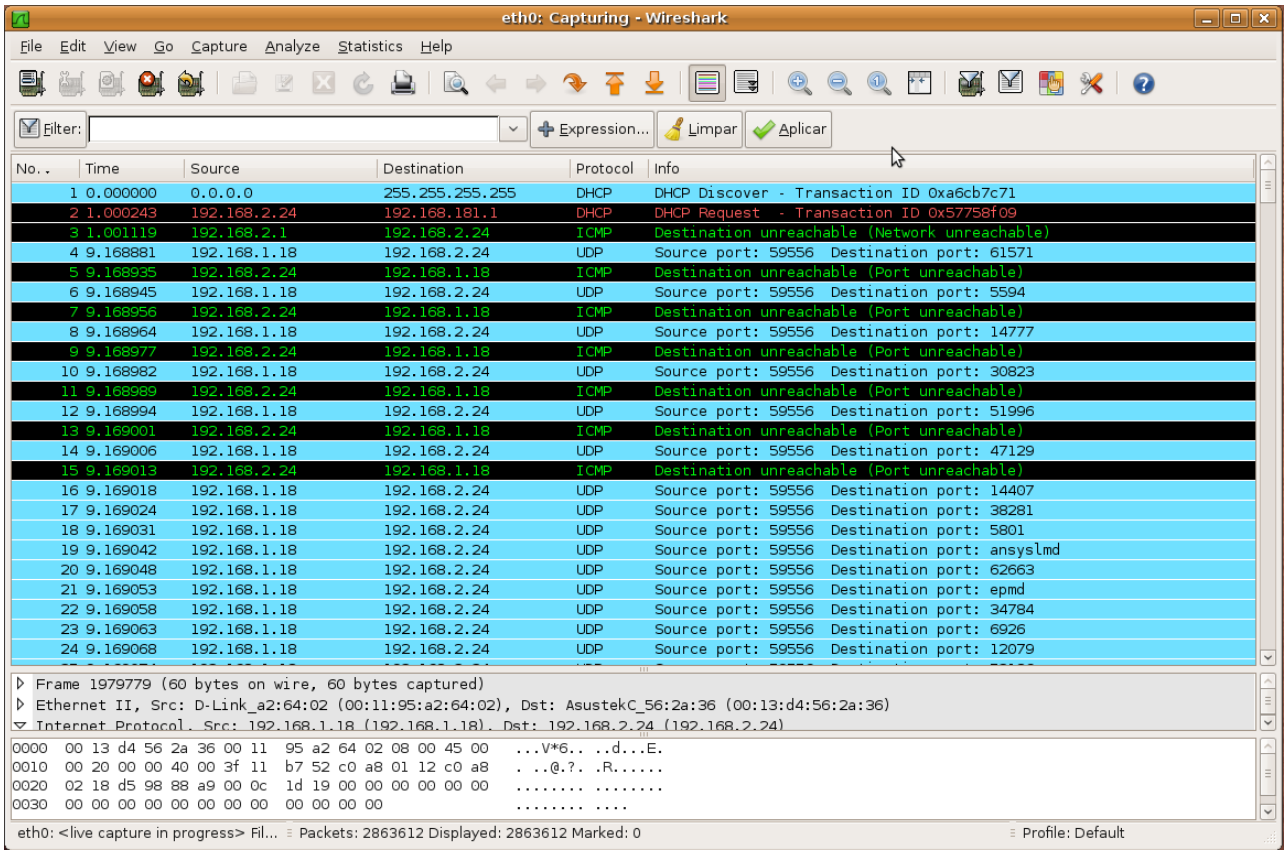
Nesse cenário, ambas as VLANS abrigam as máquinas zumbis, que executarão o ataque a uma máquina na rede adversária segundo ordem recebida da máquina mestre presente na mesma VLAN. As VLANS estão interconectadas por bridge, e ao mesmo tempo, o monitoramento de tráfego com o MRTG será realizado em cima do **roteador**, no qual espera-se constatar um aumento anormal do tráfego.

Uma dificuldade encontrada foi em relação ao software daemon do Trinoo (**ns.c**). Inicialmente, a rudimentar concepção do código obriga que a lista de possíveis mestres do daemon seja declarada explicitamente dentro do código-fonte, que então deve ser compilado e não permite nenhuma modificação de Ips dentro da rede para execução correta do ataque, algo que definitavamente não esperávamos. Em segundo lugar, uma das variáveis auxiliares (*arg1*, linha 57) estava declarada com 4 bytes, quando eram necessários 1024 bytes para sua inicialização na função *bzero* (linha 90).

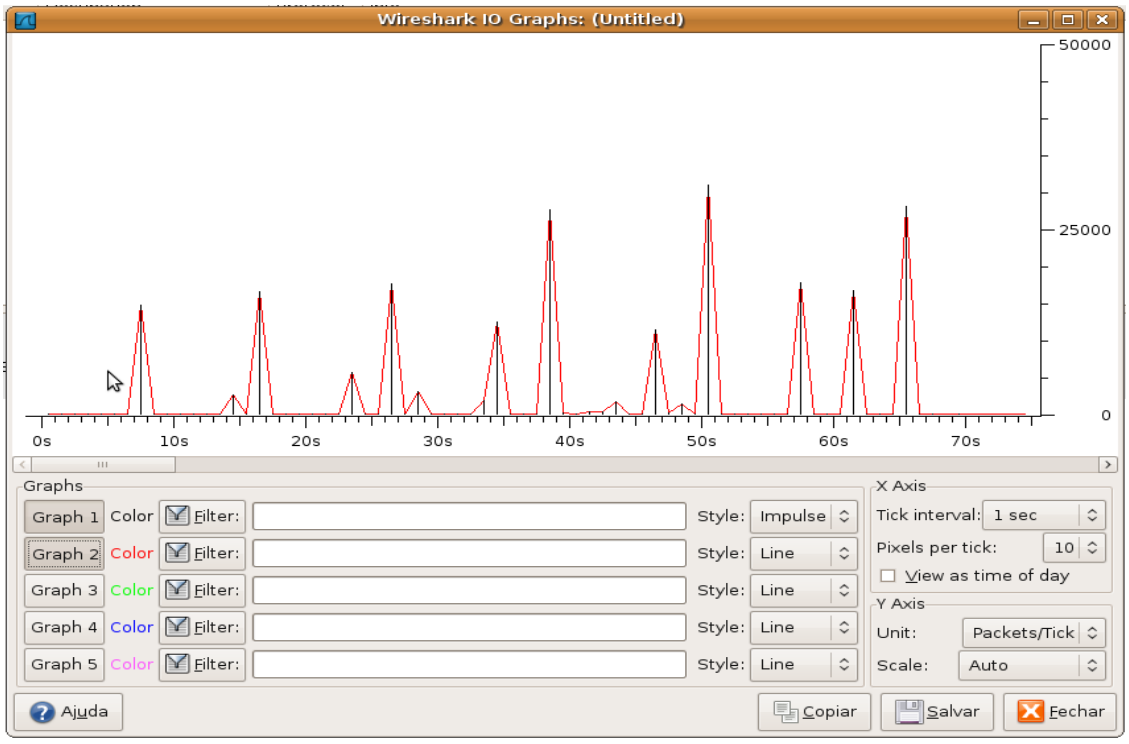
Os ataques foram executados de uma VLAN para outra, de maneira a verificar experimentalmente a ferramenta de maneira bilateral. Alguns resultados são colocados a seguir.

Resultados Obtidos

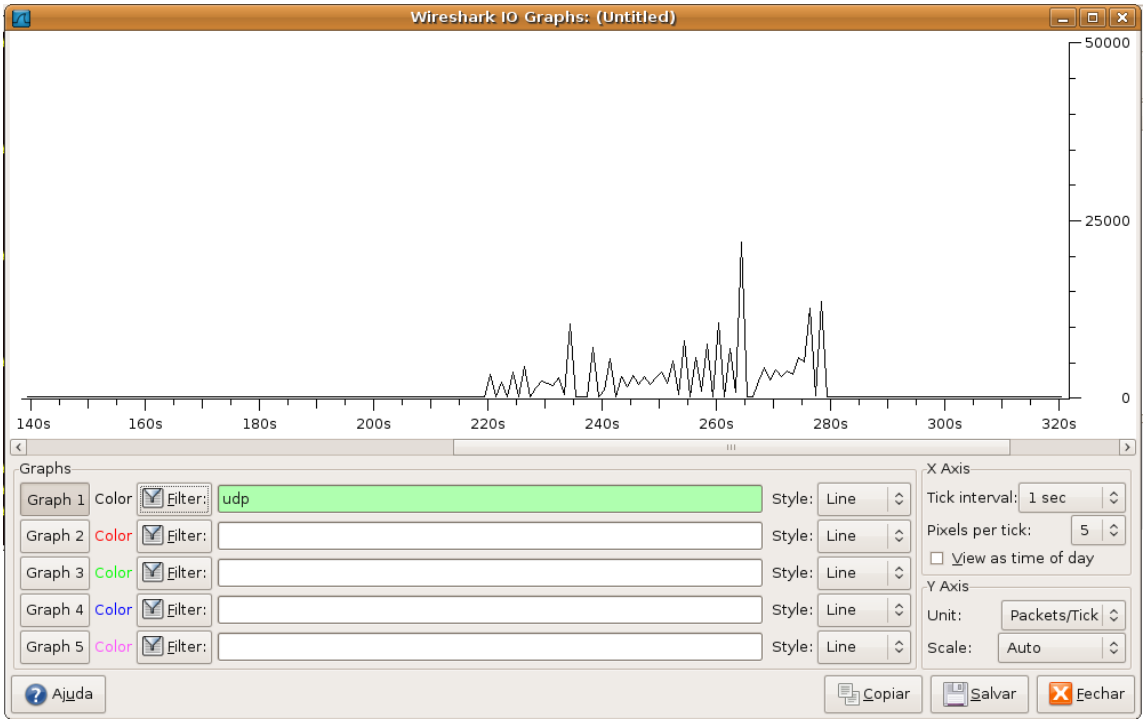
A execução do Trinoo efetivamente acarreta em um aumento abrupto de pacotes UDP enviados pela máquina zumbi, conforme pode ser visualizado a seguir:



Executando o Trinoo por apenas um minuto em um primeiro ataque, registrou-se graficamente o aumento no tráfego de pacotes deixando a máquina zumbi:



Adicionalmente, também foi anotado graficamente o aumento dos pacotes orinigados por comunicação UDP, em um segundo ataque de um minuto :



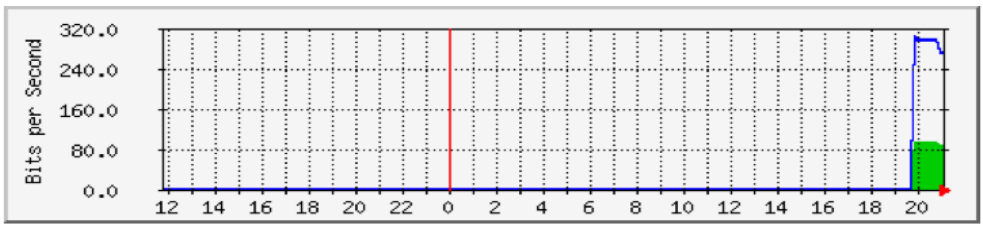
O aumento de pacotes fluindo entre as VLANS pode ser notado pelos gráficos gerados pelo MRTG, após um terceiro ataque, dessa vez entre as duas máquinas de VLAN 2 para uma máquina da VLAN 1, que foi executado continuamente durante toda a noite, entre quarta(16/12) e quinta-feira (17/12). Um gráfico da verificação por tempo mínimo de amostragem(5 minutos) do MRTG mostra o início do ataque:

Traffic Analysis for 1/5 --

System: in
Maintainer:
Description: RMON-Port--5-on-Unit-1
ifType: Gigabit Ethernet (117)
ifName: 1/5
Max Speed: 1000.0 Mbits/s

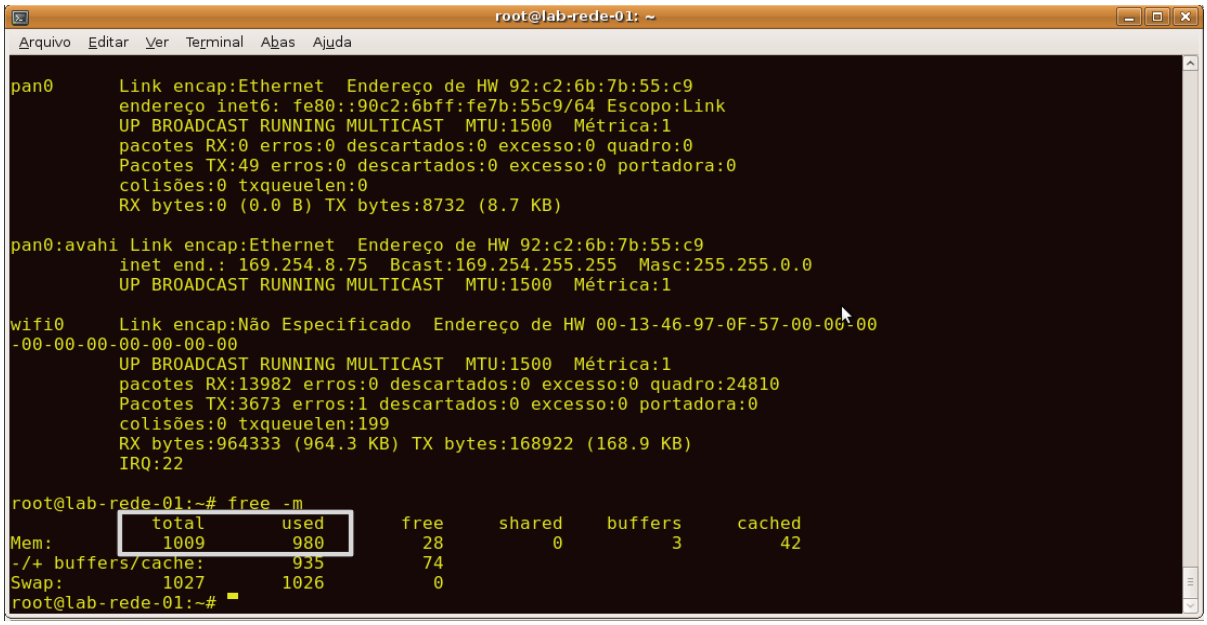
The statistics were last updated **Wednesday, 16 December 2009 at 21:06**, at which time the device had been up for **3:04:44**.

`Daily' Graph (5 Minute Average)

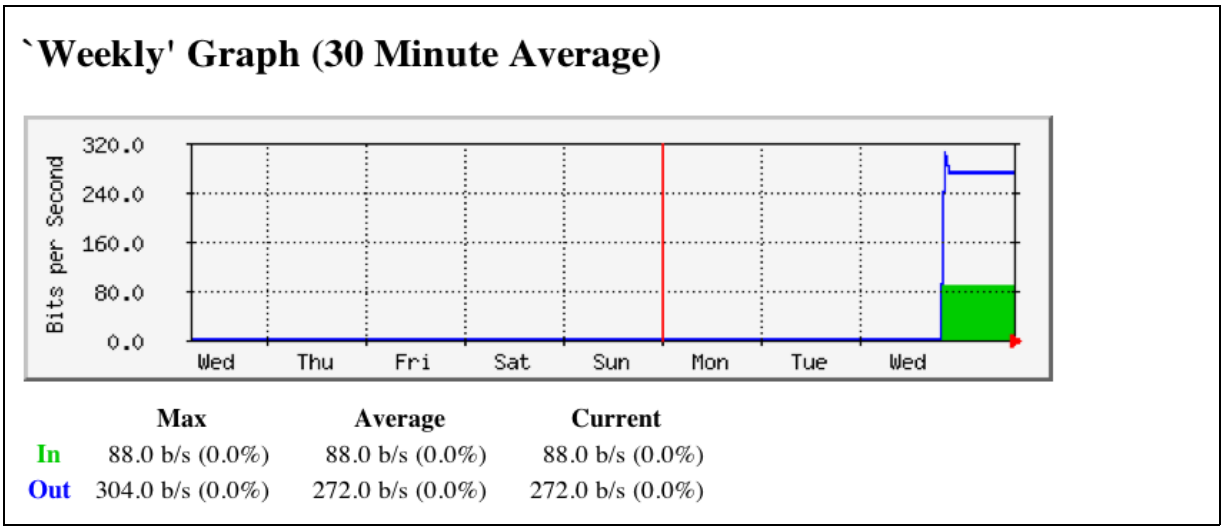


	Max	Average	Current
In	96.0 b/s (0.0%)	88.0 b/s (0.0%)	88.0 b/s (0.0%)
Out	304.0 b/s (0.0%)	288.0 b/s (0.0%)	272.0 b/s (0.0%)

Apenas duas máquinas zumbis já foram capazes, em alguns minutos, de provocar o uso de memória *swap* no alvo, originando significativa queda de desempenho do computador para atividades simples, como gerenciar arquivos, uso do browser e mesmo obter um *screenshot* de tela:



O resultado desse ataque noturno também foi registrado, ficando disponível no diretório **var/www/mtg** da máquina, bem como os arquivos de log, segundo configuração executada pelo utilitário **cfgmaker**.



Conclusões

Esse trabalho mostra que em certos ambientes, desprovidos de segurança e sem o menor comprometimento administrativo para com a infra-estrutura da rede - como eram aqueles no final da década de 90 quando surgiu o Trinoo - uma ferramenta simples de DDOS operada por usuários danosos era capaz de comprometer facilmente uma organização inteira, como o ocorrido com a Universidade de Minnesota em 1999.

O Trinoo foi o primeiro software para DDOS conhecido e, por ser pioneiro, apresenta diversos pontos fracos facilmente exploráveis para detectar ataques provenientes de seu uso. O principal deles talvez seja o uso exclusivo do protocolo UDP. Dittrich[8] cita diversas fraquezas da ferramenta derivadas desse fato, que originaram as primeiras técnicas para evitar ataques DDOS via Trinoo. Existem ainda fraquezas adicionais, incluindo até mesmo recuperação das senhas para o controle dos masters através de análise do arquivo binário do programa. Um segundo aspecto do Trinoo que inviabiliza seu uso nos dias atuais é a falta de um mecanismo dinâmico para que zumbis encontrem mestres na rede, estando os endereços IP dos mestres fixos por já virem definidos do código-fonte do programa. Ambos esses fatores foram o insumo para a evolução do Trinoo para outros softwares de DDOS nos anos seguintes, como TFN, TFN2K, Shaft, Mstream, dentre outros.

O MRTG se mostrou uma ferramenta confiável para monitoramento de redes. Em nosso trabalho, não tivemos oportunidade de explorar o monitoramento com base nas informações das MIBs de dispositivos, atentando somente para o tráfego gerado, e não monitorando nenhuma característica específica do roteador. Os utilitários para configuração do MRTG se mostram incrivelmente úteis, em especial **cfgmaker** e **indexmaker**. Uma exploração mais minuciosa desses utilitários, em conjunto com o uso amplo dos recursos previstos pelo protocolo SNMP e as MIBs de dispositivos, além da extensa lista de scripts funcionais e válidos existentes para todo o tipo de tarefa tornam o MRTG a ferramenta amplamente aceita e utilizada que é hoje, inclusive até fora do escopo do monitoramento de redes, devido à suas características para geração de gráficos peculiares.

Referências

1. *Multi Router Traffic Grapher.* Wikipedia, The Free Encyclopedia. Recuperado em 17/12/2009. http://en.wikipedia.org/wiki/Mrtg

2. *Simple Network Management Protocol.*Wikipedia, The Free Encycopedia. Recuperado em 17/12/2009. http://en.wikipedia.org/wiki/Simple_Network_Management_Protocol

3. *Management Information Base*. Wikipedia, The Free Encycopedia. Recuperado em 17/12/2009. http://en.wikipedia.org/wiki/Management_information_base

4. *What´s MRTG?.* Documentação Oficial Online. Recuperado em 17/12/2009. http://oss.oetiker.ch/mrtg/doc/mrtg.en.html

5. *RRDtool*. Wikipedia, The Free Encycopedia. Recuperado em 17/12/2009. http://en.wikipedia.org/wiki/RRDtool

6. *Denial of Service Attack*. The Free Encycopedia. Recuperado em 17/12/2009. http://en.wikipedia.org/wiki/Denial-of-service_attack#Distributed_attack

7. *Trinoo.* Wikipedia, The Free Encyclopedia. Recuperado em 17/12/2009. http://en.wikipedia.org/wiki/Trinoo

8. *Trinoo Analysis*. David Dittrich, Universidade de Washington, 1999. http://packetstormsecurity.org/distributed/trinoo.analysis.txt