Análise do comportamento de servidores web sob ataques causados por BotNets

Leandro Ferreira Canhada[1], **Eduardo Maroñas Monks**[1]

[1]Faculdade de Tecnologia Senac Pelotas- Fatec Senac Pelotas
Rua Gonçalves Chaves, 602 - Centro - Pelotas - RS, 96015-560 (053) 3225-6918

lfcanhada,emmonks@gmail.com

Abstract. *This article aims to analyze the performance of web servers under denial of service attacks using botnets simulation tools. These attacks have caused substantial losses to companies who need to spend large sums to develop solutions and protection tools against these attacks.*

Resumo. *Este artigo tem o objetivo de analisar o desempenho de servidores web sob ataques de negação de serviço, com o uso de ferramentas de simulação de BotNets. Estes ataques tem causado grandes prejuízos à empresas, que necessitam gastar altas somas para desenvolver soluções e ferramentas de proteção contra esses ataques.*

1. Introdução

Atualmente, um servidor Web é indispensável para qualquer empresa de pequeno, médio ou grande porte, que tenha um sistema desenvolvido para ser executado por meio de navegadores. Todo servidor web está suscetível a ataques, causando lentidão, ou até mesmo tornando-o indisponível, por algum tempo, ou de forma permanente, enquanto esse ataque DoS *(Denial of Service)* ainda estiver em execução. Na Figura 1, mostra estatísticas do aumento do número de incidentes reportados ao CERT.br [Freire 2015].

Figura 1. Aumento dos ataques ao longo dos anos.

Segundo está análise realizada pelo CERT.br no ano 2014 foram recebidas 1.047.031 notificações de incidentes relacionados a Internet, sendo, 223.935 ataques de negação de serviço a servidores web.

Sendo assim esse artigo visa um estudo de como os servidores Web se comportam sob ataque de negação de serviço originados de BotNets, entendendo o funcionamento e analisando o desempenho dos servidores mais utilizados, criando um comparativo entres eles.

2. BotNets

A definição de *BotNet*, consite em uma rede de computadores com a finalidade de executar algum programa ou comando, normalmente ataques de negação de serviço distríbuido, podendo ser utilizado de forma benéfica, mas também de forma ilícita. A utilização ilicíta desse tipo de rede, está ligada diretamente a ataques a servidores web, pois essa é a forma que os criminosos utilizam para tornar uma aplicação web lenta ou até mesmo inoperante.

Segundo a empresa TrendNet, [Rocha 2015] o Brasil é o quarto colocado entre a lista de países com a maior quantidade de servidores de controle de BotNets, como mostra a Figura 2.

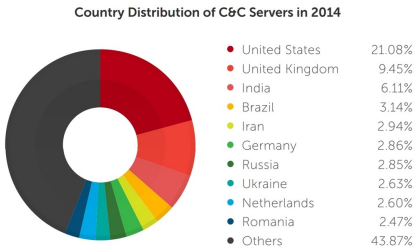

Figura 2. Ranking dos países que possuem servidores de controle de BotNets.

2.1. DoS (*Denial of Service*)

O DoS ou ataque de negação de serviço, tem como objetivo impedir que usuários utilizem algum serviço relacionado à Internet, tais como e-mails, sites de banco, comércio eletrônico, jogos e outros serviços, na medida que este ataque é bem sucedido, os usuários legítimos ficarão sem acesso e o serviço indisponível.

2.2. DDoS (*Distributed Denial of Service*)

O DDoS ou ataque de negação de serviço distribuído, consiste em atacar um servidor, utilizando vários computadores (*nodes*), comandados por um computador central, tornando um servidor, um ou mais serviços indisponíveis.

3. Tipos de Ataques e suas Ferramentas

Existem diversos tipos de ataques de negação de serviço e serão abordados os tipos mais utilizados.

3.1. *HTTP Get Flood*

Este ataque simula muitas solicitações para a mesma URL, testando a capacidade do servidor de processar as requisições HTTP GET. A ferramenta *BoNeSi* [Ulucan 2012] foi utilizada para realizar a simulação de um tráfego BotNet, desenvolvida para estudar os efeitos de ataques DDoS. Essa ferramenta gera pacotes ICMP, UDP e TCP (http), executando ataques de *flood* (inundação), sendo uma ferramenta de fácil configuração e ajustes de volume de tráfego.

3.2. *HTTP Get*

Este Teste envia requisições de conexão aos servidores web, simulando um alto tráfego de usuários acessando o servidor [W3.org 2015] A ferramenta para esse teste, foi o *httping* [Die.net 2015], para gerar conexões aos servidores web, utilizando o método GET.

3.3. *TCP Syn Flood*

O SYN flood ou ataque SYN é uma forma de ataque de negação de serviço (DoS) que consiste em enviar uma sequência de requisições SYN para um servidor [Incapsula.com 2015]. Foi usada a ferramenta *Hyenae* [Hyenae 2012] que permite reproduzir alguns cenários de ataque DoS e DDoS.

3.4. *ICMP Flood*

Este ataque consiste no envio massivo de pacotes ICMP. O Hping3 [Tomicki 2015] possibilita a realização de ataques por ICMP flood.

4. Ferramentas de Análises

Algumas ferramentas foram utilizadas para o monitoramento e coletas dos dados necessários para realizar o comparativo entre servidores web. Os recursos analisados foram a largura de banda, o processamento e tráfego protocolo HTTP.

4.1. *PRTG*

A ferramenta PRTG [Paessler 2015], analisa o tráfego das interfaces de rede dos servidores web, mostrando o fluxo de entrada e saída do dados. Na Figura 3 é mostrado um exemplo de gráfico gerado pela ferramenta PRTG.

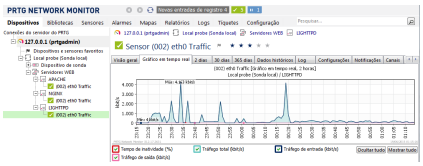

Figura 3. PRTG monitorando as três interfaces de redes dos servidores.

4.2. NtopNG

A ferramenta NtopNG[Ntop.org 2015] monitora a rede e os hosts e a quantidade de dados que foram trafegado nas interfaces de rede.

4.3. *GANGLIA*

A ferramenta Ganglia [Clusters 2008] é um sistema de monitoramento de clusters, e foi utilizada para monitorar o uso de recursos dos hosts e servidores.

5. Servidores Web

Segundo a Netcraft [Netcraft.com 2015] com dados de maio de 2015, os servidores mais utilizados na Internet foram Apache, Nginx e IIS, somando 80% do total de uso.

5.1. APACHE

O Apache Server [Apache.org 2015] é um software livre, é o mais conhecido e utilizado, possui vários módulos com recursos para uso conforme necessidade.

5.2. LIGHTTPD

O Lighttpd [Lighttpd.com 2015] é um servidor web seguro, rápido e muito flexível, projetado para ambientes de alto desempenho.

5.3. NGINX

O Nginx [Nginx.org 2015] é um servidor web rápido, leve, e com inúmeras possibilidades de configuração para melhor performance.

6. Cenário de simulação

Para fazer a simulação de ataques utilizando um BotNet, foram utilizadas algumas ferramentas em conjunto. Na Figura 4, o diagrama com os componentes da simulação dos ataques.

6.1. Cluster Beowulf

O cluster Beowuf [Neto 2008] foi utilizado para coordenar os ataques aos servidores web, utilizando o recurso MPI (*Message Passing Interface*) [Mpich.org 2015], para a troca de mensagens com os *nodes* do cluster.

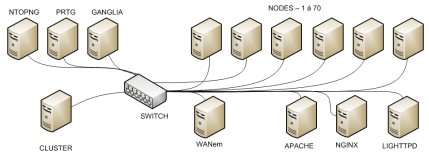

Figura 4. Organização dos servidores

6.2. VMware Player

O VMware Player 7 [Vmware.com 2015] foi utilizado para executar as máquinas virtuais, Na Tabela 1 são listadas as configurações das máquinas virtuais utilizadas.

Tabela 1. Configuração dos equipamentos utilizados

Máquina Virtual	Sistema Operacional	Cores	Mémoria Ram	Hard Disk
Cluster/Ganglia	Linux Debian 6	2	512MB	10G
Nodes	Linux Debian 6	1	256MB	10G
Servidores Web	Linux Debian 7	2	512MB	10G
NtopNG	Linux Debian 6	1	256MB	10G
PRTG	Windows 7	2	6GB	320G

Para realizar a simulação de uma rede BotNet, as máquinas virtuais utilizadas, possuem configuração baseada em serviços de hospedagem disponibilizado, na Internet, Na Tabela 2, é mostrada uma lista de configurações comuns de algumas empresas provedoras desses serviços de hospedagem.

Tabela 2. Configuração de máquinas virtuais comerciais

Empresa	Sistema Operacional	Cores	Mémoria Ram	Hard Disk
Donweb.com	SO à escolher	1 Core	1GB	10G
Hostweb.com	CentOS 6	1 Core	512MB	30G
Linode.com	SO à escolher	1 Core	256MB	24G

6.3. Conexão de Internet

Para simular uma conexão de Internet, foi utilizada uma ferramenta o WANem [WANem 2015], limitando a largura de banda entre os *nodes* e o servidor em 10MBit/s, de acordo com a pesquisa feita pela Anatel [G1.com 2015].

7. Comparativo entre os servidores WEB

Para realizar o comparativo entre os servidores, a instalação dos servidores web foi de forma padrão, sem alterações ou instalação de módulos para melhorias no desempenho.

Foi executado um teste nos servidores para definir um ambiente ideal, ou seja, carregar a URL, sem nenhum ataque ou restrição. Esse teste faz o carregamento da URL **relatorio.php**, na qual faz uma busca por todos usuários de um banco de dados Mysql. O código PHP está programado para atualizar a página de 5 em 5 segundos, calculando o tempo de carregamento. A coleta de dados para a formulação do comparativo é de 10 minutos, esse tempo foi definido, de modo que em 10 minutos o desempenho dos servidores já demostrassem se houve alteração no desempenho, ou pararam de responder aos comandos.

Para fazer a simulação de um *BotNet* foram criados três cenários, um sem ataques, outro com 45 *nodes* atacantes e outro com 70 *nodes* atacantes.

7.1. Cenário sem Ataques

Nesse cenário, foi simulado um ambiente ideal, ou seja, os servidores não sofreram nenhuma interferência ou restrição, os resultados estão listados na Tabela 3.

Tabela 3. *Sem ataques*

Servidores	Tempo de carregamento (s)
Nginx	0,04
Apache	0,05
Lighttpd	0,09

7.2. Cenário de ataques com 45 nodes

Nas Tabelas 5, 6, 7, 9, 10, 11, mostram o tempo de carregamento com os ataques. Na Tabela 4, os tempos não são mostrados pois houve perda total de comunicação.

Tabela 4. Com *Hping3*

Servidores	Tempo de carregamento (s)
Nginx	-
Apache	-
Lighttpd	-

Tabela 5. Com ataques *BoNeSi*

Servidores	Tempo de carregamentos (s)
Nginx	20,3
Apache	26,4
Lighttpd	34,2

Tabela 6. Com *Httping*

Servidores	Tempo de carregamento (s)
Nginx	7,6
Apache	8,6
Lighttpd	10,3

Tabela 7. Com ataques *Hyenae*

Servidores	Tempo de carregamento (s)
Nginx	41,2
Apache	45,3
Lighttpd	69,4

7.3. Cenário de ataques com 70 nodes

Nesse cénario foram utilizados 70 nodes para simular os ataques. Nessa situação, o tempo de carregamento da URL dos servidores, aumentou proporcionalmente, como mostram as Tabelas 9, 10 e 11. Na Tabela 8, os tempos não são mostrados pois houve perda total de comunicação com os servidores.

Tabela 8. Com *Hping3*

Servidores	Média de carregamento (s)
Nginx	-
Apache	-
Lighttpd	-

Tabela 9. Com ataques *Bonesi*

Servidores	Média de carregamento (s)
Nginx	39,1
Apache	45,3
Lighttpd	63,2

Tabela 10. Com *Httping*

Servidores	Média de carregamento (s)
Nginx	17,4
Apache	19,1
Lighttpd	30,9

Tabela 11. Com ataques *Hyenae*

Servidores	Média de carregamento (s)
Nginx	52,6
Apache	63,8
Lighttpd	89,4

As Tabela 12 mostra um comparativo dos recursos dos utilizados em cada ataque, na coluna ”PER”, estão listadas as perdas de pacotes, na coluna ”LAR”, a largura de banda utilizada.

Tabela 12. Tabela Comparativa com utilização de recursos dos servidores.

Servidor	Bonesi - %			Hyenae - %			hping3 - %			httping - %		
	CPU	PER	LAR	CPU	PER	LAR	CPU	PER	LAR	CPU	PER	LAR
Nginx	20	40	100	2,79	5	2	1	100	100	6,1	0,01	100
Apache	25	46	100	3,35	8	2	1	100	100	4,2	0,01	100
Lighttpd	28	62	100	0,45	10	2	1	100	100	14,2	0,01	100

8. Análises dos Resultados

Com os ataques das ferramentas *Bonesi* e *Hyenae* houve várias perdas de comunicação com os servidores que foram detectadas com o ping e com a ferramenta PRTG, pois os três servidores deixaram de se comunicar por vários momentos, não enviando as informações das interface de rede necessárias para a coleta dos dados para análise.

Com a ferramenta *hping3* a comunicação com os servidores foi totalmente perdida, com a execução do comando, não sendo possível a coleta do tempo de carregamento da URL.

Já com o *httping*, foi definido a execução do cluster com o comando para todos nodes, utilizando configuração padrão, foi definido que fosse executado 10 minutos, dessa forma os servidores receberam requisições Http Get simultaneamente de todos os *nodes*. Com esse teste os três servidores conseguiram responder com sucesso todas as solicitações Http Get, mesmo utilizando 100% da largura de banda.

9. Conclusões

Foi comprovado que com o aumento de *nodes*, os danos aos servidores aumentam consideravelmente. Os três servidores web apresentaram muitas perdas de pacotes, com as ferramentas Bonesi e Hyenae, causando lentidão, serviços sem responder e páginas não encontradas. O Nginx, trabalhou melhor com várias conexões, se mostrando mais robusto, considerando, que em um ambiente real, os ataques podem ser bem mais maciços, pois o número de *nodes* pode ser elevado exponencialmente.

O Apache se mostrou um pouco abaixo do desempenho, em comparação ao Nginx, nos testes efetuados, houve resposta em todas solicitações, embora com um tempo de resposta maior.

O Lighttpd, com a execução das ferramentas BoNeSi e Hyenae parou de responder, antes do final dos 10 minutos somente voltando a responder, com o final dos testes.

De acordo com os teste realizados, o servidor Nginx apresentou os melhores resultados sob ataque simulados de BotNets, em comparação aos servidores Apache e Lighttpd.

Referências

Apache.org (2015). Apache. `https://httpd.apache.org`. [Acesso em: 20-Junho-2015].

Clusters, D. (2008). Ganglia: Installation. `http://www.debianclusters.net/index.php/Ganglia`. [Acesso em: Março-2015].

Die.net (2015). httping - Linux man page. `http://linux.die.net/man/1/httping`. [Acesso em: Junho-2015].

Freire, R. (2015). CERT.br registra aumento de ataques DoS, fraudes e phishing no Brasil. `http://www.techtudo.com.br/noticias/noticia/2015/04/certbr-registra-aumento-de-ataques-dos-fraudes-e-phishing-no-brasil.html`. [Acesso em: Março-2015].

G1.com (2015). Banda larga no Brasil. `http://especiais.g1.globo.com/tecnologia/banda-larga-brasil/2015/`. [Acesso em: Junho-2015].

Hyenae (2012). How to simulate a http get botnet ddos attack. http://sourceforge.net/projects/hyenae/. [Acesso em: Abril-2015].

Incapsula.com (2015). Syn flood - ddos - incapsula. https://www.incapsula.com/ddos/attack-glossary/syn-flood.html. [Acesso em: Maio-2015].

Lighttpd.com (2015). http://redmine.lighttpd.net/projects/lighttpd/wiki. [Acesso em: Junho-2015].

Mpich.org (2015). Message passing interface. http://www.mpich.org. [Acesso em: Junho-2015].

Netcraft.com (2015). Netcraft.com. http://www.netcraft.com. [Acesso em: 20-Junho-2015].

Neto, J. L. C. (2008). Cluster Beowulf. http://www.vivaolinux.com.br/artigo/Cluster-Beowulf. [Acesso em: Fevereiro-2015].

Nginx.org (2015). Nginx. http://nginx.org. [Acesso em: Junho-2015].

Ntop.org (2015). NtopNG. http://www.ntop.org/products/traffic-analysis/ntop/. [Acesso em: Abril-2015].

Paessler (2015). PRTG Network Monitor. https://www.br.paessler.com/prtg. [Acesso em: Abril-2015].

Rocha, L. (2015). Brasil é o quarto país do mundo com mais servidores de controle de botnets. http://www.tecmundo.com.br/seguranca/75570-brasil-quarto-pais-mundo-servidores-controle-botnets.htm. [Acesso em: Junho-2015].

Tomicki, L. (2015). Ping flood. http://tomicki.net/ping.flooding.php. [Acesso em: Junho-2015].

Ulucan, C. (2012). How to simulate a http get botnet ddos attack. http://cagdasulucan.blogspot.com.br/2012/12/how-to-simulate-http-get-botnet-ddos.html. [Acesso em: Abril-2015].

Vmware.com (2015). Vmware player. http://www.vmware.com. [Acesso em: Junho-2015].

W3.org (2015). Http/1.1 method. http://www.w3.org/Protocols/rfc2616/rfc2616-sec9.html#sec9.3. [Acesso em: Abril-2015].

WANem (2015). Wanem. http://wanem.sourceforge.net. [Acesso em: Maio-2015].