MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
SECRETARIA DE CIÊNCIA E TECNOLOGIA
INSTITUTO MILITAR DE ENGENHARIA
CURSO DE GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO

Al FÁBIO LUIZ JUNIOR
Al VINÍCIUS HESSEL BENEDITO DE SOUSA

ESTUDO DE ATAQUES DISTRIBUÍDOS DE NEGAÇÃO DE SERVIÇO

Rio de Janeiro
2010

INSTITUTO MILITAR DE ENGENHARIA

AL FÁBIO LUIZ JUNIOR

AL VINÍCIUS HESSEL BENEDITO DE SOUSA

ESTUDO DE ATAQUES DISTRIBUÍDOS DE NEGAÇÃO DE SERVIÇO

Iniciação à Pesquisa apresentada ao Curso de Graduação de Engenharia de Computação como requisito parcial para a obtenção do título de Engenheiro.

Orientador: Cap Anderson Fernandes P. dos Santos – D.Sc.

Rio de Janeiro

2010

INSTITUTO MILITAR DE ENGENHARIA
Praça General Tibúrcio, 80 – Praia Vermelha
Rio de Janeiro – RJ CEP: 22290-270

Junior, Fábio Luiz; de Sousa ,Vinícius Hessel Benedito.

Estudo de ataques distribuídos de negação de serviço. Junior, Fábio Luiz; de Sousa ,Vinícius Hessel Benedito – Rio de Janeiro: Instituto Militar de Engenharia, 2010.

Trabalho (Iniciação à Pesquisa) **–** Instituto Militar de Engenharia, 2010.

1.Formação

INSTITUTO MILITAR DE ENGENHARIA

AL FÁBIO LUIZ JUNIOR

AL VINÍCIUS HESSEL BENEDITO DE SOUSA

ESTUDO DE ATAQUES DISTRIBUIDOS DE NEGAÇÃO DE SERVIÇO

Iniciação à Pesquisa apresentada ao Curso de Graduação de Engenharia de Computação como requisito parcial para a obtenção do título de Engenheiro.

Orientador: Cap Anderson Fernandes P. dos Santos D.Sc.

Aprovada em 02 de junho de 2010 pela seguinte Banca Examinadora:

_____

Cap Anderson Fernandes P. dos Santos - D.Sc.

_____

Raquel Coelho Gomes Pinto - D.Sc.

_____

Maj Sergio dos Santos Cardoso Silva - M.Sc.

Rio de Janeiro

2010

AGRADECIMENTOS

Dedicamos o referido trabalho às nossas famílias que sempre nos auxiliaram e deram incentivos para prosseguirmos em nosso projeto. Aos nossos colegas por compartilharem de bons e maus momentos, lado a lado. E, principalmente, ao Cap Anderson, nosso orientador, por fornecer a nós o devido incentivo e oportunidade para execução de trabalhos, fornecer a ajuda necessária para vencer obstáculos e, por fim, por mostrar os primeiros passos a serem dados no caminho da pesquisa.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

IP	*Internet Protocol*
TCP	*Transmission Control Protocol*
HTTP	Hypertext Transfer Protocol
ICMP	*Internet Control Message Protocol*
UDP	*User Datagram Protocol*
MBR	*Master Boot Record*
DDoS	Distributed *Denial of Service*
IRC	*Internet Relay Chat*
FBI	*Federal Bureau of Investigation*
GB	*GIGABYTE*
IDS	*Intrusion detection system*
DoS	*Denial of Service*
DNS	*Domain Name System*
EUA	United States of America
PING	*Packet Internet Groper*
Mbps	*Megabits por segundo*
ACK	Acknowledge
SYN	*Synchronize*

RESUMO

Ataques distribuídos de negação de serviço são um tema de grande interesse atual devido ao seu grande potencial de causar danos para instituições que provém serviços *onlines* e para os usuários desses serviços.

Este trabalho inclui uma ambientação ao assunto de ameaças cibernéticas com atenção especial ao tema de negação de serviço. Apresenta uma taxonomia específica para ataques de negação de serviço e inclui uma descrição detalhada do ataque 7.7 DDoS abordando características e estratégias implementadas. Os dados e estatísticas apresentados são provenientes de uma extensa pesquisa da literatura específica.

O presente documento representa uma base de conhecimento de ataques DDoS que dificilmente é encontrada em um documento centralizado.

ABSTRACT

Distributed denial of service is a topic of great current interest because of its great potential for harm that comes to institutions and online services for users of these services.

This work includes a setting to the subject of cyber threats highlighting the issue of denial of service. Presents a specific taxonomy for denial of service and includes a detailed description of the 7.7 DDoS attack approaching features and strategies implemented. The data and statistics presented are due from an extensive survey of the literature.

This document represents a knowledge base of DDoS attacks that can rarely be found in a document centered.

1 INTRODUÇÃO

O presente documento trata do planejamento e desenvolvimento do trabalho de iniciação à pesquisa (IP) tendo por tema “Estudo de ataques distribuídos de negação de serviço”. Este tipo de ataque tem se tornado cada vez mais popular nos últimos anos tendo, por consequência, causado consideráveis prejuízos econômicos. As vítimas atingidas são as mais variadas possíveis, podem ser desde sites governamentais até serviços DNS.

Com o intuito de melhor conhecer esse tipo de ofensiva, foi iniciado o estudo do assunto neste trabalho de IP. Este estudo possui por objetivo obter uma visão geral sobre os ataques distribuídos de negação de serviço e posteriormente aprofundar-se em um tipo especifico deste ataque. A ofensiva que será detalhadamente vista será a que ocorreu contra os Estados Unidos e a Coréia do Sul conhecida como 7.7DDoS.

1.1 Contextualização.

Cada vez mais comum, os ataques às redes de computadores ganham destaque em notícias nos meios de comunicações e deixam apreensivos os usuários dessas redes. Os avanços tecnológicos que permitem serviços online mais poderosos também possibilitam novos ataques, mais complexos, fazendo com que as soluções de segurança tornem-se obsoletas em curtos intervalos de tempo.

No dia 07 de julho de 2009, por exemplo, a imprensa noticiou um incidente envolvendo os Estados Unidos e Coréia do Sul. Sites desses países foram atacados por cem a duzentas mil máquinas caracterizando um ataque de negação de serviço de grande dimensão. Esse ataque chegou a consumir 25 GB por segundo de tráfego, quantidade cerca de dez vezes maior do que típicos ataques de negação de serviço.

O Brasil, com aproximadamente 60 milhões de usuários com acesso à Internet , pertence ao *rank* dos países com maiores índices de ataques cibernéticos. Na Tabela 1.1, é apresentado esse *rank* referente às ameaças virtuais, segundo registros da Symantec.

.

Tabela 1.1: Atividades maliciosas por país [10].

Rank 2008	Rank 2007	País	Média Percentual 2008	Média Percentual 2007	Rank Código Malicioso	Rank Spam	Rank Phishing	Rank Bot	Rank Origem de Ataque
1	1	E.U.A.	23%	26%	1	3	1	2	1
2	2	China	9%	11%	2	4	6	1	2
3	3	Alemanha	6%	7%	12	2	2	4	4
4	4	Reino Unido	5%	4%	4	10	5	9	3
5	8	Brasil	4%	3%	16	1	16	5	9
6	6	Espanha	4%	3%	10	8	13	3	6
7	7	Italia	3%	3%	11	6	14	6	8
8	5	França	3%	4%	8	14	9	10	5
9	15	Turquia	3%	2%	15	5	24	8	12
10	12	Polônia	3%	2%	23	9	8	7	17

Em uma pesquisa realizada pela Mindwave Research[11], 53% dos entrevistados sofreram ao menos um ataque por vírus e, entre 43% dos entrevistados existe um receio quanto a ataques realizados por *hackers*. A pesquisa também destaca a necessidade de soluções de segurança mais confiáveis e eficazes.

1.2 Relevância desse trabalho.

Uma possível abordagem para a identificação de ameaças utilizadas em típicas ferramentas de segurança, como Sistemas de Detecção de Intrusão (IDS – *Intrusion Detection System*), *Firewalls*, Antivírus, entre outros, utiliza bases de dados com assinaturas de ameaças virtuais. Devido ao constante surgimento de novos ataques e à evolução dos existentes, novas bases de dados devem ser produzidas constantemente. É neste ponto que se insere o presente trabalho. O projeto objetiva a geração de conhecimento, no estado da arte, sobre Ataques Distribuídos de Negação de Serviço permitindo a geração da base de conhecimento que será útil na modelagem e estudo prático do ataque, possibilitando a análise do comportamento de uma rede sob ataque. Dessa análise, podem-se obter assinaturas da ameaça em estudo possibilitando a geração de uma base de dados atualizada e,

consequentemente, o desenvolvimento de ferramentas de segurança mais confiáveis e eficazes.

1.3 Objetivos.

O presente trabalho objetiva a geração de conhecimento, no estado da arte, de um ataque distribuído de negação de serviço. Em uma fase inicial, uma pesquisa abrangente de ataques distribuídos de negação de serviço, na literatura, deverá fornecer subsídios para a escolha de uma técnica particular desse tipo de ataque. Uma vez que se tenha definido o tipo de ataque que será objeto de estudo, uma pesquisa profunda é conduzida explorando cada característica e peculiaridade do ataque em questão. Como resultado final dessa pesquisa, um documento contendo o estado da arte do ataque deverá ser produzido.

Seguindo esta introdução, o documento possui uma estrutura baseada em introdução, desenvolvimento e por fim uma conclusão.

No desenvolvimento, são vistos um conjunto dos ataques de rede mais conhecidos e, após isto, aborda-se o tema de negação de serviço e ataques DDoS. Apresenta-se então uma taxonomia especifica de ataques distribuídos de negação de serviço e, após, tem-se um estudo detalhado do ataque 7.7 DDoS, onde são abordados seus mecanismos de ataque, alvos, detalhes do *malware* entre outras características.

Por fim, temos a conclusão, onde é feita uma análise geral do trabalho.

2 ESTUDO SOBRE ATAQUES CIBERNÉTICOS

2.1 Tipos de ataques

Ataques cibernéticos podem ser realizados com base em engenharia social ou invasões técnicas. Os ataques com bases em engenharia social encontram-se além do escopo deste trabalho, sendo que, nesta seção, trataremos brevemente dos tipos de ataques mais comuns com base em invasões técnicas.

Ataques bem sucedidos trazem variadas consequências, em geral, negativas para a vítima. Um atacante que ganhe *status* de administrador dentro de um sistema computacional pode realizar operações no sistema, capturar ou alterar informações, corromper serviços, entre outras ações. Vários fatores podem facilitar ou dificultar a

realização de ataques bem sucedidos, como utilização de senhas ineficientes, vulnerabilidades de aplicações, má configuração e administração do sistema, etc. Seguem abaixo alguns tipos de ataques conhecidos.

2.1.1 *Buffer overflow*

Ataques de *buffer overflow* consistem em uma sobrecarga da pilha de memória, fazendo com que dados dessa pilha sejam sobrescritos. As informações manipuladas pelo sistema operacional são armazenadas na pilha de memória sendo que um ataque de *buffer overflow* bem sucedido permite a execução de código arbitrário de interesse do atacante.

Quando um programa chama uma sub-rotina, variáveis e o ponteiro de endereço de retorno são armazenados em uma pilha na memória principal. Quando a execução da sub-rotina termina, o sistema operacional usa o ponteiro de endereço de retorno para voltar à execução do processo no ponto em que este havia sido interrompido. Um ataque de *buffer overflow* tenta sobrescrever o ponteiro de endereço de retorno, fazendo com que ele aponte para uma região onde ouve uma prévia inserção de código.

2.1.2 Sequestro de sessão

Sequestro de sessão (*ou hijacking session*) é um processo de tomar o controle de uma conexão ativa entre dois *hosts*. Neste ataque, existe uma terceira máquina entre as duas que possuem conexão ativa, que assume o controle, ou “sequestra”, a conexão. Essa técnica se utiliza de vulnerabilidades de protocolos permitindo à máquina atacante se passar por um dos *hosts* que mantém a conexão enquanto indisponibiliza o outro. A grande vantagem do sequestro de sessão é que o momento do ataque ocorre após autenticação do *host* atacado que representa um processo complexo para ser forjado.

2.1.3 Negação de serviço

Ataque de negação de serviço (DoS) é caracterizado por uma tentativa explícita, por parte de atacantes, de impedir que usuários legítimos utilizem determinados serviços. Um ataque distribuído de negação de serviço (DDoS)

emprega múltiplas máquinas para alcançar esse objetivo[6]. Existem muitas técnicas para a realização de ataques de negação de serviço, sendo descritas na próxima seção.

2.2 Ataque de negação de serviço.

Ataques DoS visam tornar um sistema computacional propositalmente lento ou até completamente paralisado, para seus usuários legítimos, utilizando para isso o consumo indiscriminado de recursos (memória, poder de processamento, largura da banda, etc) da máquina vítima ou através da exploração de vulnerabilidades do sistema alvo.

Nos primórdios da Internet, ataques DoS eram lançados de um único computador. Esses ataques apoiavam-se, principalmente, na diferença de largura de banda existente entre o atacante e a vítima. Um atacante, possuindo maior largura de banda, poderia esgotar facilmente os recursos da vítima.

Com a rápida evolução da rede, protocolos e aplicações, este tipo de ataque tornou-se ineficaz. A abordagem moderna consiste em subverter computadores pessoais para os propósitos de ataque, formando, assim, o que se denomina *Botnet*.

O termo *Botnet* refere-se a um conjunto de computadores infectados (Zumbis ou Escravos) com algum software malicioso que permite a um atacante (Controlador) executar comandos arbitrários nessas máquinas. Essa comunicação entre controlador e escravos pode ocorrer de forma direta, através de conexões TCP diretas ou datagramas UDP, ou indireta, através de canais IRC ou *Proxy*. Além disso, o controlador pode usar níveis de máquinas intermediárias (Mestres) para aumentar o grau de anonimato e diminuir o tráfego necessário para o envio de comandos aos escravos. Essas máquinas mestres não realizam, necessariamente, ataques aos alvos sendo que, sua principal função é a de retransmitir a comunicação do controlador aos escravos. A **Erro! Fonte de referência não encontrada.**.1 apresenta a estrutura típica de uma *botnet*.

Assim, o atacante dispõe de um conjunto de agentes capazes de lançar um ataque DDoS sincronizado ou seguindo uma estratégia qualquer. A **Erro! Fonte de referência não encontrada.**.2 registra a quantidade e dimensão das *Botnets* descobertas no mês de março de 2010.

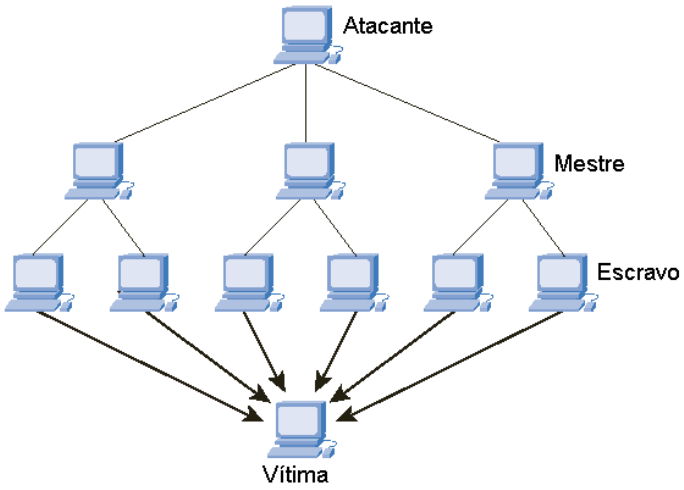

Figura 2.1: Ataque distribuído de negação de serviço.

Quando o atacante dispõem de uma *Botnet* que julga adequada, a ofensiva pode ser iniciada, bastando para isso que o atacante realize o procedimento, particular do código utilizado, de disparo do “gatilho” que pode consistir em envio de comandos, parâmetros ou arquivos de configurações. Disparado a investida, as máquinas escravas passam a executar códigos que implementam métodos de ataque. Esses códigos podem incluir métodos que colaboram para o objetivo do ataque de diversas formas: consumindo largura de banda, poder de processamento ou memória, ou exaurindo recursos associados a serviços.

A principal estratégia utilizada, atualmente, para conseguir esses efeitos, e que não depende diretamente de nenhuma vulnerabilidade existente no sistema da vítima, é a inundação de pacotes. Ataques dessa categoria consistem no envio, em grande quantidade, de pacotes para a vítima causando uma sobrecarga do sistema ou dos *links* de comunicação. A conseqüência dessa sobrecarga pode ser uma lentidão ou uma total paralisação de serviços do alvo. Este resultado dependerá, a princípio, de quantos computadores foram empregados como escravos e do quão intenso é o poder computacional da vítima.

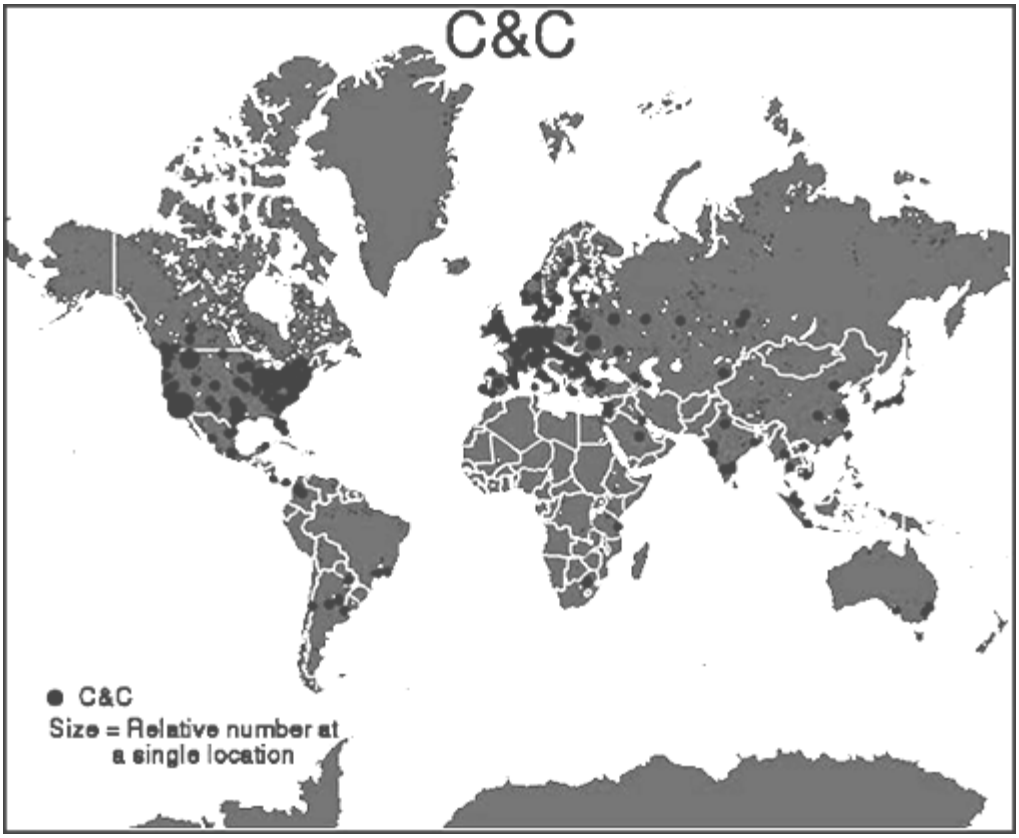

Figura 2.2: Mapa estatístico das Botnets identificadas recentemente [5].

2.3 Taxonomia de ataques DDoS.

Com o objetivo de estabelecer uma classificação consistente dos tipos de ataque DDoS para ser utilizada no decorrer do projeto, fez-se necessária a criação, ou adoção, de uma taxonomia dos mecanismos de ataques DDoS existentes. Em decorrência da rápida evolução das técnicas de ataques, é desejável que esse modelo de classificação englobe não apenas as técnicas existentes, mas também estratégias passíveis de implementação em um futuro próximo.

A taxonomia adotada foi proposta em [6], conforme Figura 2.3.

Esta taxonomia adotada classifica os ataques quanto aos seguintes itens: grau de automação, mecanismo de propagação, vulnerabilidade explorada, taxa de ataque e impacto causado.

A seguir discorreremos sobre cada um desses itens de classificação.

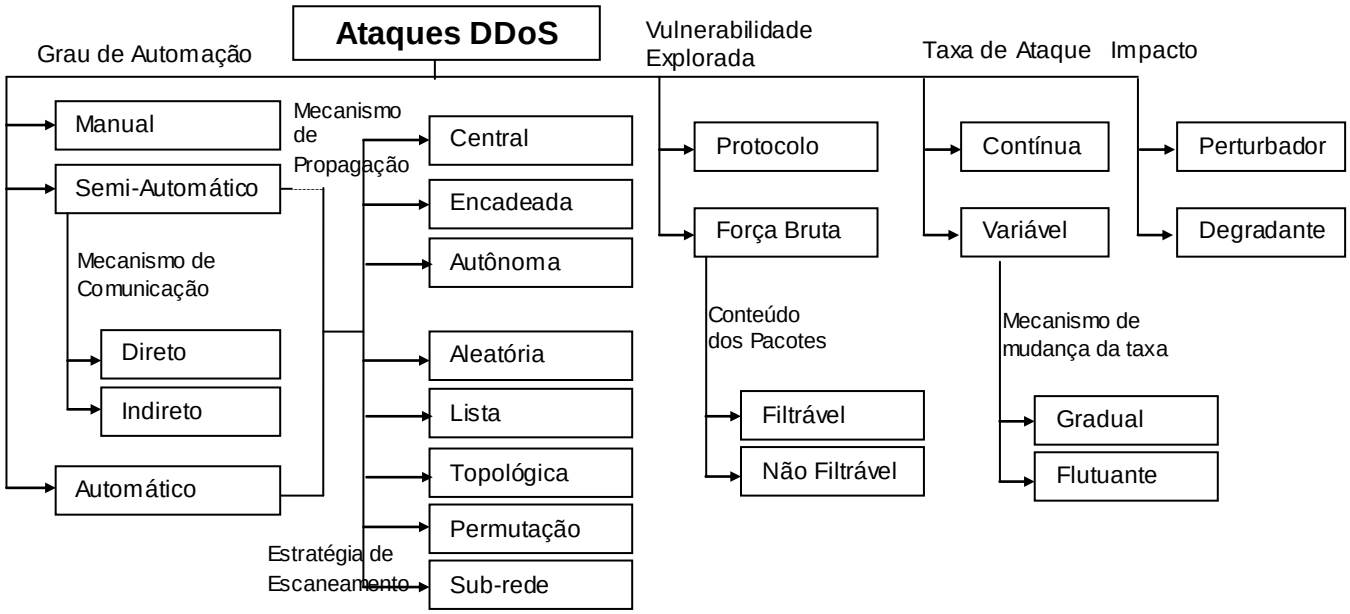

Figura 2.3: Taxonomia de ataque DDoS adotada no projeto [6].

2.3.1 Grau de Automação

O processo de preparação de um ataque DDoS envolve a localização de máquinas zumbis e o envio do comando de início de ataque para as mesmas. Segundo o grau de automação envolvido na etapa de lançamento da ofensiva, os ataques podem ser classificados em Manuais, Semi-Automáticos e Automáticos.

Manual

No atual ponto de desenvolvimento de redes, o ataque DDoS manual mostra-se impraticável. O ataque ocorre pelo escaneamento de endereços em busca de máquinas com determinadas vulnerabilidades e, ao encontrá-las, explora sua vulnerabilidade e introduz o código de ataque. Dessa maneira, a quantidade de agentes envolvidos neste tipo de ataque é limitada reduzindo drasticamente o impacto do ataque.

Semi-Automático

Em ataques semi-automáticos, a rede DDoS é, geralmente, formada por uma máquina controladora (atacante) e diversas máquinas agentes (zumbis). O atacante inicia a fase de busca de agentes e, utilizando a máquina controladora, define o tipo, a estratégia e o momento do lançamento do ataque.

De acordo com a forma de comunicação entre a máquina controladora e os zumbis, a taxonomia adotada classifica os ataques semi-automáticos em duas subcategorias:

Ataque com comunicação direta:

A comunicação entre o controlador e os agentes é feita de forma direta, através da inserção de endereços IP nos códigos de subversão. Uma vez que o código tenha sido executado na máquina alvo, ela estabelece conexão com o controlador através do IP inserido no código. Por sua vez, o controlador gera uma lista de endereços IP dos agentes que estabelecem essas conexões.

Uma deficiência desse tipo de estratégia é a facilidade de descoberta da rede DDoS, uma vez que toda máquina participante da rede possui registrado um IP de outra máquina da rede. Além disso, o controlador e os agentes mantêm uma porta de escuta para conexões TCP, tornando-se identificáveis por *firewalls* e scanners de rede.

Como ferramenta de ataque DDoS do tipo semi-automática direta, pode-se citar o Trinoo (Junho/1999). O trinoo é uma ferramenta para lançar ataques de negação de serviço coordenados do tipo UDP *flood*. A comunicação entre o controlador e os agentes é feita através da porta 1524/tcp ou 27444/udp e a conexão dos agentes é autenticada através de uma senha personalizável.

Ataque com comunicação indireta:

A comunicação entre o controlador e os agentes é feita de forma indireta utilizando-se de servidores *Proxy* ou canais IRC. Isso garante ao atacante um maior grau de anonimato se comparado à forma direta de comunicação.

O benefício imediato decorre do fato da comunicação ser feita através de portas de serviços legítimos. Isso, aliado ao fato de os agentes não necessitarem de uma porta de escuta, aumenta a dificuldade de detecção.

Em 2003 o FBI iniciou uma grande operação denominada *Cyberslam* que culminou com a descoberta de uma *Botnet* e a indiciação de *hackers* e donos de uma empresa, a *Orbit Communication*. Os donos da *Orbit Communication* pagaram a *hackers* para lançarem ataques de negação de serviço contra web sites de

empresas concorrentes. Estima-se que o ataque tenha causado um prejuízo entre U$ 200.000,00 e U$ 1.000.000,00

A *Botnet* utilizada para o ataque foi formada através de uma das primeiras versões de um *worm* de IRC, escrito em C++ e assembly, conhecida como Agobot . Esse *bot* conectava a canais IRC pré-estabelecidos e executava comandos do atacante. Entre as funcionalidades presentes naquela implementação destacam-se o *keylogger*, execuções de programas e capacidade de executar ataques SYN Flood e HTTP Flood utilizando a banda disponível do computador infectado.

De acordo com a fonte da qual o software malicioso é obtido, os ataques são classificados em Ataques com Fonte Central de Propagação, Ataques com Propagação Encadeada e Ataques com Propagação Autônoma.

Fonte Central de Propagação:

Os códigos são armazenados em servidores. Assim, durante a fase de infecção de máquinas, os códigos maliciosos são transferidos desses servidores. Esta estrutura de distribuição de código é frágil, uma vez que, sendo a fonte de código centralizada, uma falha dos servidores pode comprometer todo o processo de distribuição.

O Li0n *worm* utiliza-se dessa estratégia. Escrito em shell script, esse *worm* procura por uma vulnerabilidade na porta 53, escaneando aleatoriamente endereços IP classe B. Quando uma vítima em potencial é encontrada, a vulnerabilidade permite que um comando específico inicie um *download* da cópia do *worm* e a execução do *script* de *startup*.

Propagação Encadeada:

Os códigos utilizados para novas infecções são obtidos dos próprios agentes subversivos. Desta forma, quando uma máquina é subvertida, ela se torna a fonte de código de futuras infecções. Essa estratégia reduz a fragilidade da propagação, aumentando a dinamicidade da fonte de código.

O Ramen *worm* utiliza-se dessa estratégia. Uma vez que o computador tenha sido infectado, o *worm* envia e-mails para duas contas pré-estabelecidas do controlador e começa a varrer a rede em busca de novas máquinas com potencial de infecção.

Propagação Autônoma:

Esse tipo de estratégia não faz uso do conceito de fonte de códigos de onde as novas máquinas infectadas obtêm o *software* malicioso. A abordagem utilizada nesse tipo de propagação consiste na inserção do código malicioso no momento da exploração de vulnerabilidade. Logo, essa forma de propagação mostra-se bastante robusta já que, uma vez que um agente tenha identificado uma máquina passível de exploração, essa última será subvertida já no momento de exploração da falha, sem a necessidade de transferências adicionais.

O Warhol *worm* é um tipo de *worm* de alta velocidade de disseminação que utiliza propagação autônoma. Identificado pela primeira vez em 2002, ele é capaz de contaminar todas as máquinas vulneráveis em no máximo 15 minutos.

Automático

Nos ataques automáticos, a fase de lançamento do ataque é automatizada. As diretrizes de ataque são inseridas no código de ataque e implantadas diretamente nos agentes deixando o agressor livre da obrigação de enviar os parâmetros de ataque para os agentes antes do lançamento do ataque. Geralmente, um *backdoor* também é inserido na máquina vítima com o objetivo de permitir um futuro reuso do agente. Essa característica dos ataques automáticos isenta o atacante de um contato, direto ou indireto, com os agentes na fase de lançamento do ataque criando uma situação de maior proteção ao atacante.

Tanto em ataques semi-automáticos quanto nos automáticos, a fase de recrutamento de agentes para a formação da *Botnet* é feita de forma automatizada. De acordo com a estratégia utilizada pelos agentes na procura de novas máquinas com potencial de subversão, os ataques podem ser classificados em ataques que utilizam escaneamento aleatório, escaneamento por lista, escaneamento topológico, escaneamento com permutação e escaneamento de sub-rede local.

Escaneamento Aleatório:

Nesse tipo de busca, cada agente procura por máquinas, com potencial de subversão, utilizando-se endereços IP gerados de forma aleatória. Essa forma de

escaneamento gera um alto tráfego de pacotes na rede, além de gerar colisões na busca já que uma mesma máquina pode ser sondada por vários agentes.

Escaneamento por Lista:

Nesse tipo de busca, cada agente procura por máquinas, com potencial de subversão, segundo endereços IP contidos em uma lista fornecida por uma fonte externa. Ao subverter uma nova máquina, o agente lhe envia metade de sua lista de endereços e passa a operar sobre a outra metade. Essa estratégia elimina as colisões na busca e faz com que a velocidade de recrutamento cresça exponencialmente.

Escaneamento Topológico:

Na busca topológica, endereços de busca são obtidos dos novos agentes. Como exemplo de busca topológica tem-se os *worms* de e-mails que obtém os alvos da lista de e-mail das máquinas infectadas.

Escaneamento com Permutação:

Nesse tipo de busca, todos os agentes compartilham uma permutação do espaço de endereços de IP, ou seja, cada endereço de IP é mapeado em um índice da permutação. Cada agente inicia a busca pelo índice correspondente ao seu próprio endereço IP e segue a busca nos índices subseqüentes da permutação. Uma vez que um agente encontra um índice que corresponde a uma máquina já infectada ele retoma a busca em um índice aleatório da permutação. Essa estratégia causa um efeito de busca semicoordenado, mantendo a característica aleatória e reduzindo o número de colisões da busca e o tráfego na rede.

Escaneamento de Sub-rede Local:

Essa estratégia de busca pode ser utilizada em conjunto com os tipos de busca anteriores. Nela, os agentes buscam alvos preferencialmente na sub-rede a que pertencem permitindo a infecção de máquinas vulneráveis por trás do *firewall*.

2.3.2 Vulnerabilidade

Um ataque DDoS pode explorar falhas no sistema alvo para causar a negação de serviço. Segundo a vulnerabilidade explorada, os ataques podem ser classificados em ataques de exploração de protocolo e ataques de força bruta.

Exploração de Protocolo:

Ataques de exploração de protocolos exploram *bugs* de implementação ou características de determinados protocolos. Protocolos que demandam grande capacidade computacional dos servidores e baixas cargas de processamento do cliente são bons protocolos para exploração para ataques de negação de serviço.

Ataques do tipo TCP SYN, por exemplo, exploram o processo de estabelecimento de conexão em três vias do protocolo TCP. Para cada requisição TCP SYN, o servidor aloca recursos para o restante do processo de conexão. Nos ataques desse tipo, os agentes iniciam múltiplas conexões que nunca são concluídas, esgotando os recursos do servidor.

Em ataques de servidores de autenticação, explora-se o fato de a carga de processamento necessária para gerar uma credencial de autenticação ser menor que a necessária para processar e autenticar a credencial. Dessa forma, os agentes envolvidos nesse ataque geram mais credenciais do que o servidor é capaz de processar causando a negação de serviço a usuários autênticos.

Força Bruta:

Ataques de força bruta geram pacotes aparentemente legítimos que esgotam os recursos do servidor devido a quantidade enviada. Este tipo de ataque se utiliza da capacidade de gerar pacotes, pelos agentes da *botnet*, ser maior que a capacidade de processamento do servidor.

De acordo com o tipo de pacote gerado, o ataque de força bruta pode ser classificado como filtrável ou não filtrável.

Nos ataques filtráveis, os pacotes enviados são de serviços não críticos e podem ser filtrados pelo *firewall* sem maiores conseqüências. Exemplos desses tipos de ataques são UDP *Flood* e ICMP *Request Flood*.

Por outro lado, ataques não filtráveis geram pacotes de serviços legítimos e sua filtragem poderia levar a negação de serviço de clientes legítimos. Exemplos desses tipos de ataques são HTTP *Request Flood* e DNS *Request Flood*.

2.3.3 Dinâmica da Taxa de Ataque.

A dinâmica da taxa de ataque especifica como a ofensiva se distribui ao longo do tempo. Podem ser divididos em dois grandes grupos: os ataques com taxas contínuas e os com taxa variável.

Os ataques com taxas contínuas apresentam um constante fluxo de pacotes advindos dos agentes. Esse fluxo pode ser baixo e espalhado durante um longo período ou pode ser alto e concentrado em um curto período. Esta última estratégia é geralmente mais aplicada, porém é de fácil detecção. A facilidade de detecção vem do fato que irá existir um pico pronunciado e isolado no consumo dos recursos, que se comparado com estatísticas passadas, pode ser claramente identificado.

Um exemplo deste tipo de ataque foi o que aconteceu com o site Mininova.org. Uma *botnet* com centenas de computadores produziu um fluxo de 2 gigabits por segundo de pacotes UDP. Tal ação teve como resultado a inatividade do site por um período de 14 horas. O gráfico da Figura 2.4 mostra o tempo de carregamento do site para os usuários legítimos. Em muitos casos o tempo de carga espirava (período maior que 30 segundos).

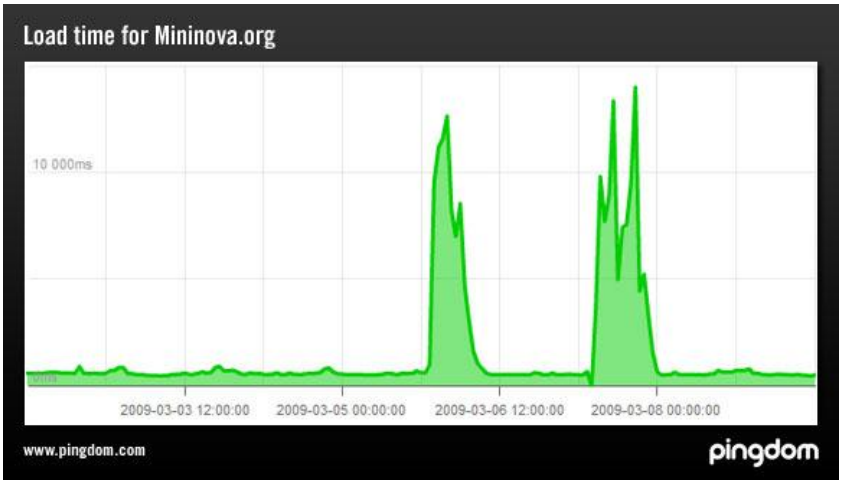

Figura 2.4: Tempo de carregamento do site Mininova.org para os usuários legítimos [14].

Este gráfico da Figura 2.5 mostra a disponibilidade do site. Pode-se verificar que, durante as duas ofensivas ao site, este teve sua disponibilidade fortemente reduzida chegando até ao caso de indisponibilidade total.

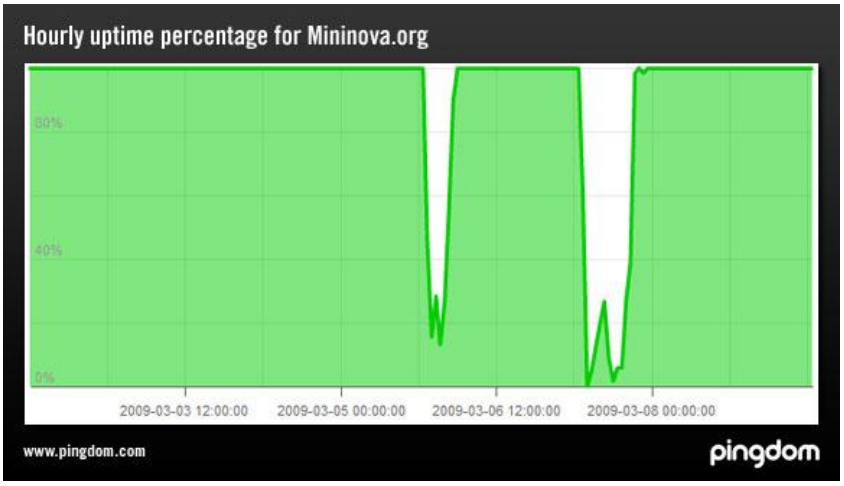

Figura 2.5: Disponibilidade do site Mininova.org para os usuários legítimos [14].

Já os ataques com taxas variadas são muito mais difíceis de detectar. Eles variam seu fluxo de dados e dificilmente empregam todo o poder de ataque de uma só vez. Essa categoria de ataques com taxa variada se divide em outras duas subcategorias: com taxa crescente e os com taxa flutuante. O primeiro começa com um pequeno fluxo de dados e vai aumentando ao longo de um grande intervalo de tempo. Isso faz muitas vezes com que os provedores do serviço pensem que há um aumento natural da demanda pelo mesmo. O segundo é uma estratégia bem mais elaborada que as outras. Utiliza uma abordagem de flutuação da taxa de ataque de maneira a se adaptar ao comportamento da vítima. Aumenta o fluxo do ataque em certos horários de maior demanda e diminuem em horários de baixa demanda.

2.3.4 Impacto

Os ataques DDoS também são classificados quanto ao nível de impacto causado à vitima. Existem duas classificações básicas: ataques perturbadores e ataques degradativos. O primeiro tipo tem como objetivo a completa paralisação da vítima. O segundo visa consumir parcialmente e por um longo período de tempo seus recursos. Ambas as investidas causam consideráveis prejuízos financeiros ao agredido por vários motivos. Um destes seria a redução do número de usuários legítimos do sistema. Tal fato ocorre pois ou o serviço será negado ou estará muito

lento. Um outro motivo seria a necessidade do agredido de fazer uma atualização de seus servidores e redes assim como também alocar dinheiro para mecanismos de proteção.

3 ATAQUE DE 7.7 DDOS.

Esta ofensiva foi selecionada entre tantas outras devido principalmente a sua magnitude. A seguir serão descritas as particularidades deste ataque.

Na manhã de 4 de Julho de 2009, uma série de *websites* comerciais e do Governo dos EUA começou a sofrer ataques de negação de serviço. Logo em seguida, *websites* Sul-Coreanos passaram a registrar ataques com as mesmas características. Já no início da análise do tráfego de ataque, percebeu-se que ambos eram alvos de um mesmo ataque que ficou marcado em virtude da importância dos alvos, das características peculiares do ataque e da projeção dada pela imprensa mundial.

Análises do tráfego de ataque apresentaram indicações dos ataques presentes na **Erro! Fonte de referência não encontrada.**3.1.

Tabela 3.1: Métodos de ataques utilizados no ataque 7.7 DDoS.

Tipo	Outras informações
ICMP Echo Request Flood	
UDP Flood	Porta do alvo: 80
TCP SYN Flood	Porta do alvo: 80
IP Protocol 0 Flood	
HTTP GET Request Flood	Requisição da pasta raiz: '/'

A estimativa inicial de *hosts* integrantes da *botnet*, segundo a *Shadowserver Foundation*, foi de 100.000 *hosts*. Posteriormente, uma equipe independente, que obteve acesso a um dos servidores de comando e controle do *malware*, estimou 200.000 *hosts* maliciosos. Cerca de 95% dos *hosts* comprometidos possuíam endereços IP da Coréia do Sul.

Equipamentos da Arbor Network registraram ataques com até 100.000 pacotes por segundo em TCP Flood consumindo uma banda de 25 Mbps até 50

Mbps. Outras esquipes divulgaram relatórios com ataques consumindo 25 Gbps. Em média, cada *host* zumbi gerava 103 pps (18.5 KB/s) sendo:

- HTTP GET Flood: 20 pps (12KB/s)
- TCP SYN Flood: 40 pps (3.6 KB/s)
- UDP 80 Flood: 20 pps (1.4 KB/s)
- ICMP Flood: 23 pps (1.5 KB/s)

O efeito do ataque pode ser mensurado a partir da **Erro! Fonte de referência não encontrada.**3.1 de consultas aos DNS da coréia do sul.

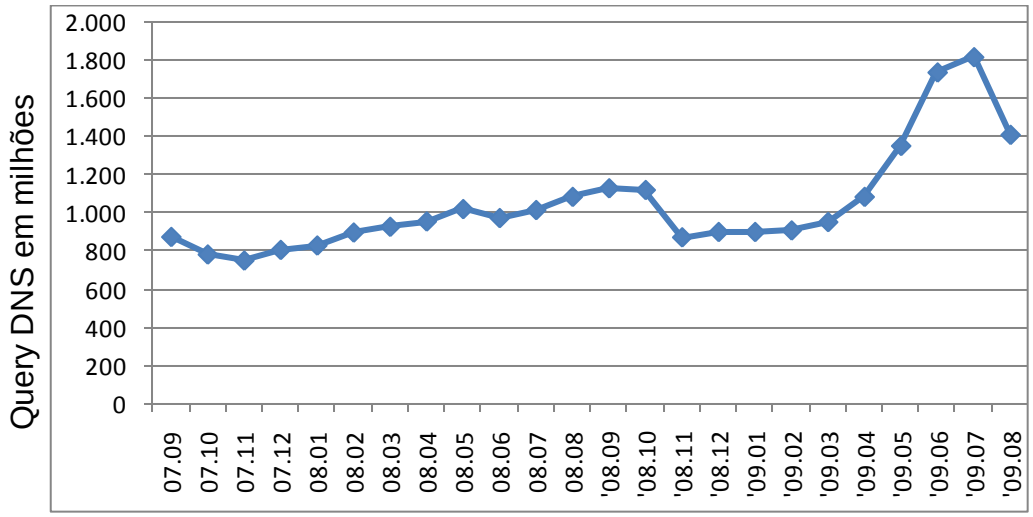

Figura 3.1: Requisições DNS na Coréia do Sul durante o ataque de 7.7 DDoS [5].

O *malware* utilizado foi uma variação do MyDoom A ou B (2003/2004) com código adaptado às necessidades do ataque. Amostras do *worm* não apresentaram proteção contra anti-vírus nem compactação de código. Os alvos iniciais estavam inseridos em arquivo de configuração na instalação inicial do *malware*. Durante o ataque, foi percebido tráfego de atualização da lista de alvos na *botnet*. A **Erro! Fonte de referência não encontrada.**.2 apresenta as atualizações de alvos nos arquivos de configuração (Vide ANEXO 1)

Tabela 3.2: Estatística de atualização dos alvos durante o ataque

Web sites dos EUA	Web sites da Coréia do Sul	Período
5	0	05/07/2009 02:00 – 05/07/2009 14:00

21	0	05/07/2009 22:00 – 06/07/2009 07:00
21	0	05/07/2009 22:00 – 06/07/2009 18:00
0	13	07/07/2009 18:00 – 08/07/2009 18:00
12	0	07/07/2009 21:00 – 08/07/2009 07:00
0	14	08/07/2009 18:00 – 09/07/2009 18:00
0	7	09/07/2009 18:00 – 10/07/2009 18:00

3.1 Classificação do Ataque

De acordo com a taxonomia adotada neste trabalho, o ataque em questão apresenta um grau de automação semi-automático com comunicação indireta devido aos níveis de servidores distribuídos de comando e controle. Quanto a propagação, o *malware* apresentou disseminação essencialmente central, com repositórios em servidores de códigos ou web sites explorados. As estratégias de ataque implementadas exploraram vulnerabilidades de protocolo, como TCP SYN *Flood*, e também força-bruta, como HTTP Get *Flood*, apresentando taxa de ataque constante. O impacto do ataque pode ser considerado perturbador já que causou a negação de serviço de diversos websites mas, vale notar que, alguns web sites tiveram seu desempenho apenas degradado.

3.2 Detalhamento do *malware*

A *botnet* utilizada nos ataques de 7 de julho apresentava uma estrutura hierarquizada. Foram utilizados cerca de 400 servidores distribuídos, para comando e controle, repositório de código e concentradores de informações. Essa estrutura se estendia por vários níveis implementando um alto grau de anonimato aos atacantes.

A análise de máquinas zumbis permitiu identificar o *malware* como uma adaptação dos worms MyDoom, variantes A e B. A infecção inicial se deu através da exploração de websites coreanos populares.

Três variações de códigos foram identificadas, mas todas apresentam a mesma anatomia do ciclo de ataque. Inicialmente o arquivo executável msiexec?.exe estabelece comunicação com um dos servidores de comando e controle, remove configurações anteriores e cria o arquivo de configuração pxdrv.nls. Além disso, nesta etapa é criado o serviço WmiConfig, responsável pela execução de ataques de negação de serviço.

Na etapa do lançamento do ataque, o serviço WmiConfig, que mantém comunicação com um servidor de comando e controle, envia a hora local para sincronismo e recebe o arquivo contendo a lista de alvos. No momento determinado, os alvos são recuperados do arquivo uregvs.nls e o ataque de negação é iniciado. No arquivo uregsv.nls, diversos parâmetros definem a estratégia de ataque.
A Tabela 3.3 apresenta algumas características implementadas no *malware*.

Tabela 3.3: Ataque implementados.

	IP de origem	IP de destino	Tipo de ataque
1	Original	Alvo	SYN
2	Forjado	Alvo	SYN
3	Original	Alvo	ACK
4	Forjado	Alvo	ACK
5	Original	Alvo	UDP
6	Forjado	Alvo	UDP
7	Original	Alvo	ICMP
8	Forjado	Alvo	ICMP
9	Alvo	Broadcast	ICMP
10	Original	Alvo	HTTP GET

Após horário e data de término do ataque, previsto no arquivo uregsv.nls, o executável wmcfg.exe é iniciado. Esse processo é responsável pela criação dos serviços mstimer.dll e wversion.exe além da criação do arquivo de configuração SERVICE.LOG.

O serviço mstimer.dll decodifica as informações contidas em SERVICE.LOG e tenta conectar, aleatoriamente, em um entre oito servidores para o *download* do arquivo flash.gif que é um arquivo executável precedido de um pequeno cabeçalho JPEG (Nesse caso, o cabeçalho e a extensão do arquivo não possuem uma correlação). Além disso, mstimer.dll envia o arquivo flash.gif através de e-mails spam para outros usuários. Esses e-mails não trazem perigo para os usuários que o recebem pois trata-se, essencialmente, de um arquivo executável corrompido.

O serviço wversion.exe é executado depois de meia noite do dia 10 de julho, sendo responsável pela inutilização do MBR da máquina infectada. Ele insere a string "*Memory of the Independence Day*" no setor de inicialização do disco. Além

disso, ele escaneia o disco em busca de arquivos com extensões específicas tornando-os inutilizáveis através de compactação para gz com senha.

Como forma de apagar as evidências, após cada etapa, os processos e serviços apagam seus próprios arquivos de origem.

Visto como o *malware* se comporta dentro da maquina infectada, serão agora abordados os mecanismos de ataque, do ponto de vista da rede, por ele utilizados. As seguintes técnicas são utilizadas: ICMP Echo Request Flood, UDP 80 Flood, IP Protocol 0 Flood, TCP SYN Flood – porta 80 e HTTP GET Request Flood para '/'.

Nas próximas seções serão vistos detalhes de cada mecanismo.

3.2.1 ICMP Echo Request Flood

A realização do ataque é simples e, a priori, os sistemas operacionais já possuem uma aplicação que propiciaria sua execução, o PING. O ataque se dá através do envio de pacotes ICMP Echo Request para a vítima sendo que essa, por sua vez, retorna um pacote ICMP Echo Response para o atacante. Assim, um atacante pode negar recursos do alvo através do consumo de recursos da CPU, envolvido no processamento dos pacotes ICMP Echo Request e geração dos pacotes ICMP Echo Response, ou através do consumo de largura de banda devido à quantidade de pacotes ICMP trafegados na rede. Pacotes ICMP menores são usados para a sobrecarga do roteamento e, consequentemente, consumo de largura de banda enquanto pacotes ICMP maiores são usados para consumo de CPU.

Pelas características inerentes ao ataque, é normalmente utilizado simultaneamente outros ataques como complementares com o principal objetivo de degradar o poder de processamento do host alvo e a largura de banda de seu enlace.

Experimento

Com o objetivo de analisar um sistema computacional em ataque, foi realizado um experimento por Sanjeev Kumar [4], de simulação de ataque ICMP Echo Request Flood em um ambiente de laboratório.

Para o experimento, utilizou-se um computador com processador Pentium-4 com clock de 2.66 GHz, sistema operacional Windows-XP e interface de rede fast Ethernet de 100 Mbps. Utilitários de rede também foram usados para o

monitoramento do ataque. A carga do ataque foi aumentada gradualmente e registros foram feitos a cada 10Mbps.

Os resultado mostraram que a demanda por processamento aumenta rapidamente com o aumento da taxa de de pacotes ICMP Echo Request por segundo envolvido no ataque.

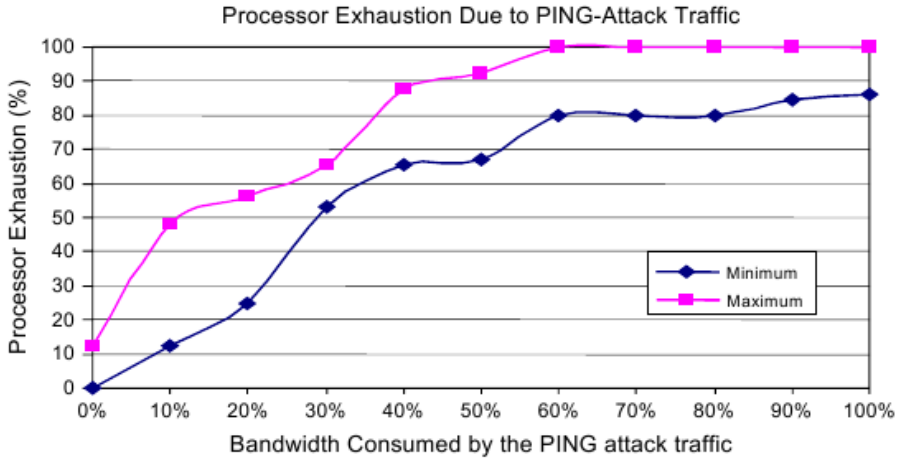

Figura 3.2: Demanda de processamento em função da largura de banda (fast Ethernet 100 Mbps) utilizada no ataque [4].

A Figura 3.2, apresenta os gráficos de mínimo e máximo processamento exigido para processar pacotes ICMP Request consumindo determinada percentagem da largura de banda. Observa-se que com 60% (60 Mbps) da largura de banda consumida, o computador apresenta picos de processamento de 100% podendo entrar em uma situação de instabilidade.

Um outro ponto importante na análise do ataque é a capacidade do host alvo em responder os pacotes ICMP Request. A Figura 3.3 registra a taxa de pacotes ICMP Request corretamente processados pelo host alvo.

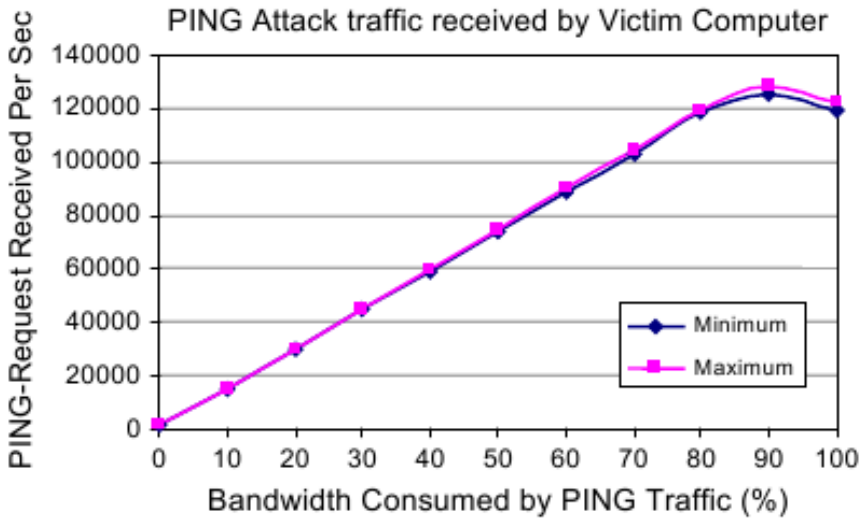

Figura 3.3: Pacotes ICMP Echo Response em função da largura de banda consumida pelo ataque [4].

Através da Figura 3.3 e da **Erro! Fonte de referência não encontrada.**3.4, pode-se perceber que a partir de uma carga de 80% (80 Mbps) o host alvo não possui poder de processamento suficiente para processar todos os pacotes ICMP Request.

Tabela 3.4: Número de mensagens PING processadas e retornadas pela máquina vítima.

Carga de Ataque	Número de PING Transmitidos	Número de PING processados e respondidos pelo host vítima.
80% (80 Mbps)	119.047	118.982
90% (90 Mbps)	133.929	122.420
100% (100 Mbps)	148.810	120.789

Da Tabela 3.4, percebe-se que com uma taxa de 119.047 pacotes/s o host alvo consegue processar e enviar pacotes ICMP Response a uma taxa de 118.982 pacotes/s. Nesse caso, o host alvo já está utilizando uma boa parcela de seu poder de processamento da CPU e capacidade da interface de rede sendo que outros *web-services* começam a sofrer degradação.

A **Erro! Fonte de referência não encontrada.** ilustra o consumo de memória no decorrer do ataque.

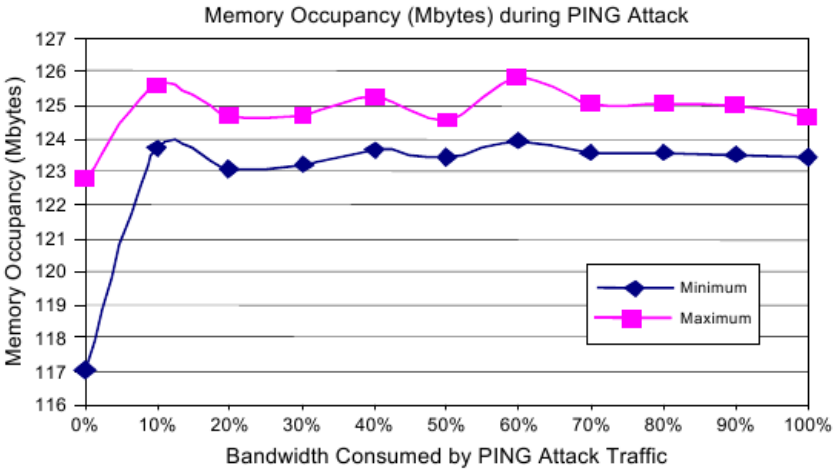

Figura 3.4: Consumo de memória em função da carga de ataque [4].

Comparada as Figuras 3.2 e 3.3, a Figura 3.4 ilustra a pouca representatividade do consumo de memória no ataque. Os impactos em processamento e largura de banda são mais relevantes se comparados ao consumo de memória.

Dos dados obtidos, pode-se perceber que o recurso que apresenta mais sensibilidade ao ataque é o processamento da CPU. O poder de processamento é degradado rapidamente com o aumento da carga de ataque. Como o padrão ICMP prevê o pacote ICMP Response de mesmo tamanho do pacote ICMP Request, pacotes ICMP maiores tendem a esgotar os recursos da CPU mais rapidamente. Por outro lado, pacotes menores facilitam a tarefa de envio dos atacantes possibilitando maiores taxas de envio, causando um maior *overhead* de roteamento e, conseqüentemente consumindo maior largura de banda.

Notou-se também que o consumo de memória não é um fator preocupante já que pequena parcela foi consumida, se comparado aos padrões de capacidade de memória atuais.

3.2.2 UDP FLOOD.

O ataque de UDP flood é, em certos aspectos, semelhante ao ataque de ICMP Request Flood. O atacante envia pacotes UDP para portas de destino aleatórias de uma vítima. A vítima, ao receber tais pacotes, utiliza parte de seu poder de processamento para processar o pacote UDP e gerar pacotes ICMP de resposta. Assim, da mesma forma que o ataque de ICMP, o ataque de UDP consome largura de banda da vítima e causa a negação de serviço pelo consumo de poder da CPU.

Como o ataque gera pacotes ICMP de resposta do host alvo, o atacante, geralmente, utiliza IP Spoofing para alterar o endereço IP de origem e assim livrar-se do tráfego de resposta (que poderia, inclusive, agir como um ataque de negação de serviço)

Efeitos do UDP Flood em Serviços.

Para ilustrar o efeito do ataque UDP Flood em diversos serviços que um servidor pode oferecer, descreveremos um experimento, realizado por Jelena Mirkovic [7], em que se registra a taxa de falha de transações do servidor em função do tráfego UDP do ataque.

Foi gerado tráfego legítimo de HTTP, FTP, DNS, Telnet e ICMP. A degradação do sistema e largura de banda pode influenciar em maior ou menor grau a qualidade de um serviço, de acordo com sua natureza. Conversações de áudio ou jogos online, por exemplo, tem requerimentos de QoS mas rígidos sendo fortemente influenciados por atrasos, perda de pacotes ou *jitter*. Por outro lado, serviços como envio de e-mails são mais tolerantes a variações da rede.

Foi produzido tráfego UDP de ataque que visava sobrecarregar o gargalo de largura de banda de 1,25 Mbps. Serviços como FTP e Telnet se mostraram mais sensíveis sendo afetados por ataques de baixa carga, enquanto DNS se mostrou resistente apenas sofrendo degradação com grande carga de ataque.

O gráfico da Figura 3.5 ilustra o resultado do experimento.

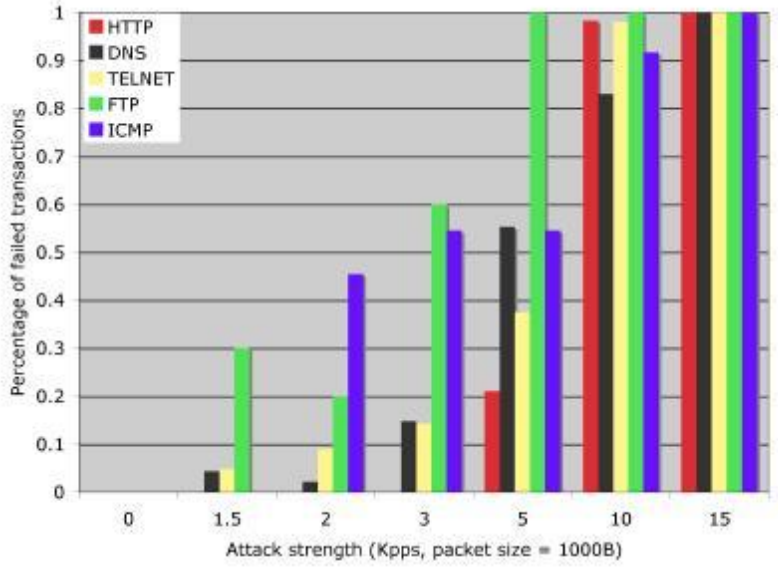

Figura 3.5: Taxa de falhas das transações em função da carga de ataque UDP [7].

3.2.3 TCP SYN Flood

O ataque de negação de serviço através da estratégia conhecida como TCP SYN Flood explora uma vulnerabilidade do protocolo TCP/IP que permite que um atacante esgote recursos da vítima através de conexões TCP semi-abertas.

Para o estabelecimento de uma conexão TCP, inicialmente o servidor realiza a “abertura passiva” que consiste na abertura de uma porta para conexão. Após a abertura passiva, um cliente que queira estabelecer a conexão pode iniciar a “abertura ativa” que é realizada em três etapas (*three-way handshake*). Inicialmente o cliente envia um SYN para o servidor que, responde com um SYN-ACK. Para finalizar o processo de estabelecimento de conexão, o cliente envia um ACK e a conexão está pronta para ser usada para a transferência de dados.

Diferentemente de uma conexão TCP comum, um ataque TCP SYN Flood ocorre quando um atacante inicia diversas conexões TCP sem, no entanto, enviar o ACK, obrigando o servidor a aguardar a finalização do processo de estabelecimento das conexões. O ataque surte efeito pois o número de conexões semi-abertas que pode ser suportada por cada porta TCP (*backlog*) é limitada e, quando o limite de conexões é atingido, o servidor ignora novas requisições de conexão até que o *time-out* das conexões as invalide, criando assim um período de negação de serviço.

A fim de dificultar a detecção do ataque e evitar uma alta carga de pacotes SYN-ACK, o atacante geralmente altera o endereço de origem do cabeçalho IP. O atacante deve assegurar-se que o endereço utilizado seja roteável, mas não alcançável. Assim o servidor será notificado (através de pacotes ICMP) de que o host encontra-se inalcançável, porém a camada TCP irá ignorar esses avisos. Por outro lado se o endereço IP utilizado é de um host alcançável, este receberá pacotes SYN-ACK e, por sua vez, responderá com pacotes RST liberando a conexão semi-aberta do servidor dificultando a tarefa do atacante.

Benchmark e Soluções

A partir de agora, será descrita um experimento, realizada por Ross Oliver [12], de diversas soluções proprietárias de proteção ao TCP SYN Flood.

Uma arquitetura de proteção a este tipo de ataque consiste em um hardware intermediário entre o servidor e o cliente, como ilustra a Figura 3.6. Dessa forma, os clientes estabelecem conexões TCP com o hardware e apenas as conexões

corretamente estabelecidas são levadas ao servidor. A vantagem neste tipo de solução é que o hardware intermediário, devido a sua finalidade, pode ser um componente mais específico e eficiente no tratamento das conexões TCP do que um servidor de uso geral.

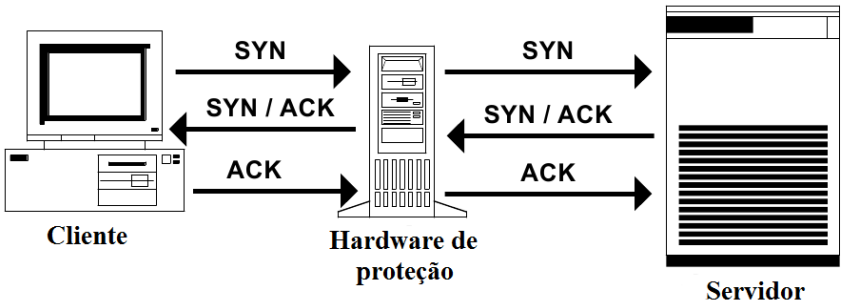

Figura 3.6: Arquitetura de solução para TCP SYN Flood

Para a realização do experimento, utilizou-se o ambiente representado esquematicamente na Figura 3.7:

- Web Server: Linux RedHat 7.2
  - Apache Web Server
- Web Client: Windows 2000
  - Script usando wget para capturas as páginas e mensurar o tempo de resposta
- Atacante: Linux RedHat 7.2
  - Gerador de SYN Flood

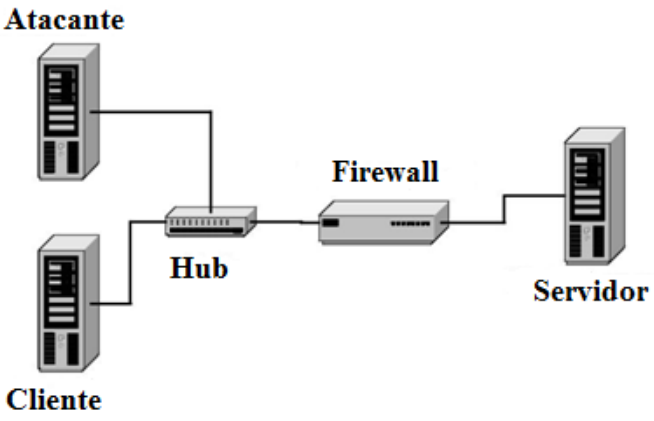

Figura 3.7: Ambiente de testes

As soluções testadas foram:

- Pix (Cisco)

- Firewall-1 (Checkpoint)
- Netscreen 100 (Netscreen)
- AppSafe/AppSwitch (Top Layer)

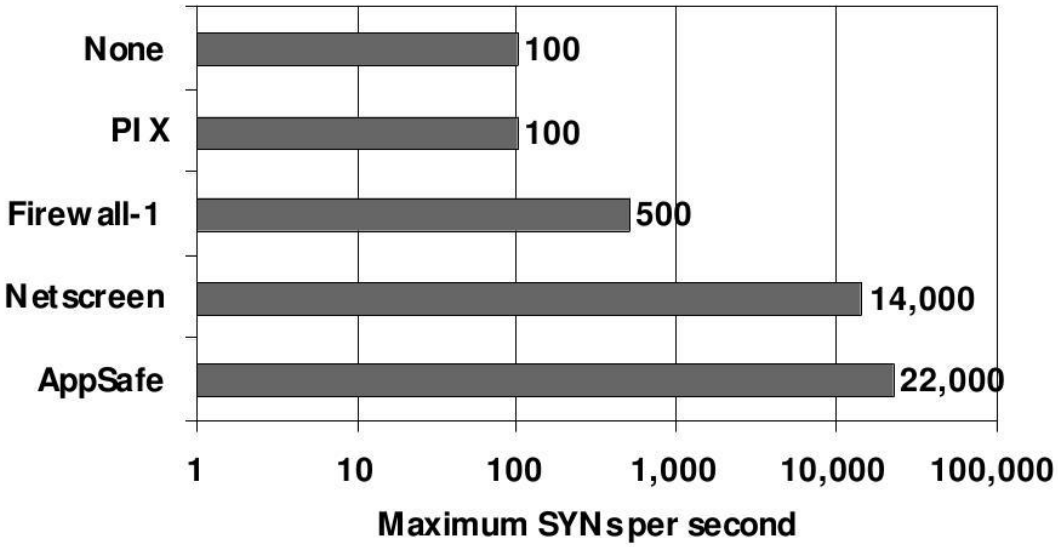

Figura 3.8: Benchmark das soluções testadas [5].

A Figura 3.8 apresenta os resultados obtidos.

O limite teórico máximo de tráfego que um atacante pode gerar é:

- Modem Analógico: 87 SYN/s
- ISDN, Cabo, DSL: 200 SYN/s
- T1: 2.343 SYN/s
- 474 sistemas hackeados: 94.800 SYN/s

3.2.4 HTTP GET Request Flood

HTTP GET Request Flood é utilizado como ataque complementar para a sobrecarga do Web Server. Essencialmente, a requisição HTTP é semelhante à de um usuário legítimo porém é executada diversas vezes e tem como objetivo consumir recursos da vítima.

3.2.5 Protocol 0 Flood

O modelo TCP/IP define um campo '*protocol'* no cabeçalho IP responsável por informar qual protocolo da camada de transporte encapsula o *payload*. Quando um protocolo não disponível na implementação da camada de transporte é

referenciado, o host de destino gera um pacote ICMP tipo 3, *Destination Unreachable*, com código 2, *Protocol Ureachable* Error, informando a impossibilidade de processamento do pacote pela camada de transporte.

Essa característica é explorada no ataque de IP Protocol Flood de forma semelhante ao ICMP Flood. A demanda de processamento na recepção do pacote IP e na criação do pacote ICMP Protocol Unreachable Error permite o esgotamento de recursos da vítima. O atacante inunda a vítima com pacotes IP com campo 'protocol' setado para um valor não implementado. Esse valor pode ser escolhido entre protocolos pouco utilizados ou através de ferramentas que varrem os protocolos disponíveis na vítima. O ataque em questão utilizou o protocol 0, HOPOPT (IPv6 Hop-By-Hop Option).

4 CONCLUSÃO

Ao longo do trabalho foram notadas dificuldade na obtenção de dados e estatísticas de ataques distribuídos de negação de serviço. Essa dificuldade se deve, talvez, ao fato de que estes dados representam um conhecimento sensível e valioso para Governos, empresas de segurança e sites comerciais. Pode-se notar que, atualmente, existe um movimento no sentido de firmar acordos de cooperação entre governos no intuito de melhorar suas defesas contra essas ameaças.

Nesse contexto, este trabalho vem contribuir com conhecimento, documentado em língua portuguesa, sobre o tema. Portanto, o objetivo motivador da pesquisa foi plenamente satisfeito.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] ARBOR NETWORK. **July, 2009 South Korea and US DDoS Attacks**. Arbor Network, 2009. 12 p.

[2] HARRIS, B. HUNT, R. **TCP/IP security threats and attacks methods**.

[3] JEONG, Hyun Cheol. **Experience with DDoS**. Korea Internet & Security Agency; 2010.

[4] KUMAR, Sanjeev. **PING Attack – How bad is it?**. Edinburg: The University of Texas-Pan American; 2005.

[5] LEE, Han Sang. **Counteracting DDoS Attack in KR**. Kuala Lumpur: Korea Internet & Security Agency; 2010.

[6] MIRKOVIC, Jelena, MARTIN, Janice, REIHER, Peter. **A Taxonomy of DDoS Attacks and DDoS Defense Mechanisms**. Los Angeles: University of California; [2004]. Technical report #020018

[7] MIRKOVIC, Jelena, REIHER, Peter, FAHMY, Sonia, et al. **Measuring Impact of DDoS Attacks.**

[8] ROMERO, Luiz Carlos, KACUTA, Luiz Yukishigue, DE OLIVEIRA, Viviane Luciane. **Segurança da Informação – Tipos de Ataques**. Campinas: Universidade de Campinas; 2003.

[9] UDHAYAN, J. ANITHA, **R. Demystifying and Rate Limiting ICMP hosted DoS/DDoS Flooding Attacks with Attack Productivity Analysis.** IEEE International Advance Computing Conference; 2009; Patiala, Índia.

[10] COLETIVA.NET [internet]. ***Ibope Nielsen Online divulga ranking da internet no Brasil*** **[**Acesso em: 05/09/2009]**.** Disponível em http://www.coletiva.net/site/noticia_detalhe.php?idNoticia=32321.

[11] INFOSECURITY TASK FORCE [internet]. **Hackers tiram o sono de 43% dos usuários de TI [**Acesso em: 05/09/2009]. Disponível em: http://www.istf.com.br/vb/noticias-de-seguranca/7889-hackers-tiram-o-sono-de-43-dos-usuarios-de-ti.html.

[12] OLIVER, Ross. **Countering SYN Flood Denial-of-Service Attacks**. Tech Mavens [internet]. Diponível em: http://www.tech-mavens.com/synflood.htm.

[13] OUAH.ORG [internet]. **Lion Internet Worm Analysis [**Acesso em 13/03/2010]. Disponível em: http://www.ouah.org/lionw.htm

[14] PINGDOM [internet]. **The anatomy of a DDoS attack [**Acesso em 07/03/2010]. Disponível em: http://royal.pingdom.com/2009/03/10/the-anatomy-of-a-ddos-attack/

[15] REDE NACIONAL DE ENSINO E PESQUISA [internet]. **Tudo o que Você Precisa Saber Sobre Ataques DDoS [**Acesso em 12/03/2010]. Disponível em: http://www.rnp.br/newsgen/0003/ddos.html.

[16] SHADOWSERVER FUNDATION [internet]. **Botnet Maps** [Acesso em 15/03/2010]. Disponível em: http://www.shadowserver.org/wiki/pmwiki.php/Stats/BotnetMaps.

[17] SOFTPEDIA [internet]. **European Botnet Runners Indicted in the FootNet DDoS Case [**Acesso em 12/03/2010]. Disponível em: http://news.softpedia.com/news/European-Botnet-Runners-Indicted-in-the-FooNet-DDoS-Case-94919.shtml

ANEXO 1 : ATUALIZAÇÃO DE ALVOS DURANTE O ATAQUE

05/07/2009 02:00 – 05/07/2009 14:00	
www.whitehouse.gov whitehouse.gov www.faa.gov	faa.gov evisaforms.state.gov
05/07/2009 22:00 – 06/07/2009 07:00	
www.whitehouse.gov www.faa.gov www.ustreas.gov www.dhs.gov www.state.gov www.dot.gov www.ftc.gov www.nsa.gov www.usps.gov www.voa.gov www.yahoo.com	www.defenselink.mil travel.state.gov www.nyse.com www.nasdaq.com www.site-by-site.com www.marketwatch.com finance.yahoo.com www.usauctionslive.com www.usbank.com www.amazon.com

05/07/2009 22:00 – 06/07/2009 18:00	
www.whitehouse.gov	www.defenselink.mil
www.faa.gov	travel.state.gov
www.ustreas.gov	www.nyse.com
www.dhs.gov	www.nasdaq.com
www.state.gov	www.site-by-site.com
www.dot.gov	www.marketwatch.com
www.ftc.gov	finance.yahoo.com
www.nsa.gov	www.usauctionslive.com
www.usps.gov	www.usbank.com
www.voanews.com	www.amazon.com
www.yahoo.com	
07/07/2009 18:00 - 08/07/2009 18:00	
www.president.go.kr	banking.nonghyup.com
www.mnd.go.kr	ezbank.shinhan.com
www.mofat.go.kr	ebank.keb.co.kr
www.assembly.go.kr	www.hannara.or.kr
www.usfk.mil	www.chosun.com
blog.naver.com	www.auction.co.kr
mail.naver.com	
07/07/2009 21:00 – 08/07/2009 07:00	
www.whitehouse.gov	www.nasdaq.com
www.faa.gov	finance.yahoo.com
www.dhs.gov	www.usauctionslive.com
www.state.gov	www.usbank.com
www.voanews.com	www.washingtonpost.com
www.defenselink.mil	www.ustreas.gov
www.nyse.com	
08/07/2009 18:00 – 09/07/2009 18:00	
www.mnd.go.kr	www.ibk.co.kr
www.president.go.kr	www.hanabank.com
www.ncsc.go.kr	www.wooribank.com
mail.naver.com	www.altools.co.kr

mail.daum.net mail.paran.com www.auction.co.kr	www.ahnlab.com www.usfk.mil www.egov.go.kr
09/07/2009 18:00 – 10/07/2009 18:00	
mail.naver.com mail.daum.net mail.paran.com www.egov.go.kr	www.kbstar.com www.chosun.com www.auction.co.kr