INSTITUTO DE PESQUISAS TECNOLÓGICAS DO ESTADO DE SÃO PAULO

Uélinton Bráulio dos Santos

Ataques Distribuídos de Negação de Serviço – Análise do Problema, Prevenção e Combate

São Paulo

2004

Uélinton Bráulio dos Santos

Ataques Distribuídos de Negação de Serviço – Análise do Problema, Prevenção e Combate

Dissertação apresentada ao Instituto de Pesquisas Tecnológicas do Estado de São Paulo – IPT, para obtenção do título de Mestre em Engenharia de Computação.
Área de Concentração: Redes de Computadores

Orientador: Prof. Dr. Frank Meylan

São Paulo

2004

Santos, Uélinton Bráulio dos
Ataques distribuídos de negação de serviço: análise do problema, prevenção e combate. / Uélinton Bráulio dos Santos. São Paulo, 2004.
91p.

Dissertação (Mestrado em Engenharia de Computação) - Instituto de Pesquisas Tecnológicas do Estado de São Paulo. Área de concentração: Redes de Computadores.

Orientador: Prof. Dr. Frank Meylan

1. Ataque distribuído de negação de serviço (Computador) 2. Ataque de negação de serviço (Computador) 3. DDoS 4. Segurança da informação (Computador) 5. Internet (Redes de computadores) 6. Tese I. Instituto de Pesquisas Tecnológicas do Estado de São Paulo. Centro de Aperfeiçoamento Tecnológico II. Título

CDU 004.491(043)
S237a

Dedico à minha esposa Tatiana, pelo amor, paciência e apoio e, aos meus filhos.

Agradecimentos

Ao professor Frank Meylan por toda a orientação, paciência, apoio e atenção.

Aos meus pais pela orientação na vida e pelo amor.

Ao amigo Alexandre pela grande ajuda.

Ao funcionários e professores do CENATEC por toda a ajuda.

A Deus por me permitir enfrentar este desafio.

RESUMO

Uma avaliação crítica e comparativa de modelos e mecanismos destinados à prevenção e combate do problema dos ataques distribuídos de negação de serviço é apresentada.

O panorama atual da problemática envolvida neste tipo de ataque é apresentado e recomendações foram elaboradas para auxiliar a sua prevenção e combate.

Um modelo que se utiliza de recursos de qualidade de serviço e possibilita a disponibilidade seletiva de sítios de comércio eletrônico foi implementado em protótipo experimental e avaliado. Nesta avaliação foram obtidos subsídios para evidenciar tal modelo como solução viável para esta disponibilidade seletiva de sítios. Os resultados obtidos na implementação do mecanismo foram comparados com outra abordagem semelhante e considerados aceitáveis em termos de perda de vazão e quanto ao objetivo de manter a disponibilidade seletiva de sítios de comércio eletrônico.

Palavras-chave: Ataques distribuídos de negação de serviço; Ataques de negação de serviço; DDoS; Mecanismos de prevenção de ataques de DDoS.

ABSTRACT

A critical and comparative evaluation on models and mechanisms destined to prevention and combat of distributed denial of service attacks problem is presented.

The current problematic panorama involved in this type of attack is presented and recommendations had been elaborated to assist its prevention and combat.

A model that utilizes quality of service resources and provide selective availability of electronic commerce sites was implemented and evaluated at an experimental prototype. In this evaluation subsidies had been gotten to evidence the model as a viable solution for selective availability of electronic commerce sites. The results obtained in the mechanism implementation had been compared with another similar boarding and considered acceptable in terms of throughput loss and regarding to the goal of provide the selective availability of the electronic commerce sites.

Key-words: Distributed denial of service attacks; Denial of service attacks; DDoS; DDoS Attacks prevention mechanisms.

Lista de Ilustrações

Lista de Tabelas

Lista de Abreviaturas

AMD	Attack Mitigation Decision-making
ACK	Acknowledgment
AF	Assured Forwarding
ARP	Address Resolution Protocol
BE	Best Effort
BGP	Border Gateway Protocol
CGI	Common Gateway Interface
CTCP	Client-to-client Protocol
CWR	Congestion Window Reduced
DCC	Direct Client to Client
DDoS	Distributed Denial of Service
DMZ	Demilitarized Zone
DNS	Domain Name Service
DoS	Denial of Service
DS	Differentiated Services
DSCP	Differentiated Services Codepoint
DSL	Digital Subscriber Line
EPM	Enhanced Probabilistic Marking
ERT	Emergency Response Team
FBI	Federal Bureau of Investigation – Birô Federal de Investigação
FTP	File Transfer Protocol
GRED	Generic Random Early Detection
HTTP	Hypertext Transfer Protocol
ICANN	Internet Corporation For Assigned Names and Numbers
ICMP	Internet Control Message Protocol
IDC	Internet Datacenter
IDS	Intrusion Detection System
IETF	Internet Engineering Task Force

IGMP	Internet Group Management Protocol
IMQ	Intermediate Queueing Device
IntServ	Integrated Services – Serviços Integrados
IP	Internet Protocol
IPsec	IP Security Protocol
IRC	Internet Relay Chat
ISP	Internet Service Provider
LVS	Linux Virtual Server
MAC	Media Access Control
Mbps	Megabit per second
MPLS	Multi-protocol Label Switching
NAT	Network Address Translation
PHB	Per-Hop Behavior – Tratamento por Salto
PPF	Preferential Packet Filtering
qdisc	queueing discipline
QoS	Quality of Service
RFC	Request for Comments
RIP	Routing Information Protocol
RPC	Remote Procedure Call
RST	Reset
SLA	Service Level Agreement
SMTP	Simple Mail Transfer Protocol
SOS	Secure Overlay Services
SOAP	Secure Overlay Access Point
SSL	Secure Socket Layer
SYN	Synchronize sequence numbers
Tcl	Tool Command Language
TCP_{rto}	Transmission Control Protocol – Ratio
telnet	Network Terminal Protocol
TFN	Tribe Flood Network
TFTP	Trivial File Transfer Protocol

TLD/tlDNS	Top Level Domain (.com, .org, .net, .biz, etc)
TLS	Transport Layer Security
TOS	Type of Service
TTL	Time to Live
VPN	Virtual Private Network

Sumário

1 INTRODUÇÃO

1.1 MOTIVAÇÕES

Nos últimos anos ocorreu um crescimento significativo no uso dos recursos de Internet em atividades comerciais, financeiras e educacionais. Os requisitos de segurança aumentaram consideravelmente, à medida que muitos novos negócios foram criados e continuam a surgir baseados nessa infra-estrutura. Mesmo negócios tradicionais tentam se adaptar às novas formas de comercialização por meio da Internet considerando também altos níveis de segurança como pré-requisito.

Ao mesmo tempo, aumentaram as facilidades para que usuários mal-intencionados, possuidores ou não de conhecimentos técnicos para isso, façam uso dos recursos da Internet para atividades ilícitas. Desde simples varredura de portas até ataques coordenados e direcionados a determinados sítios podem ser disparados utilizando ferramentas disponíveis na Internet. Entre as motivações para estes ataques estão desde a simples busca pela notoriedade no submundo da Internet, até o roubo de informações sigilosas.

Têm-se tornado cada vez mais comuns as conexões à Internet com tecnologias DSL (*Digital Subscriber Line* – Assinante de Linha Digital), entre outras chamadas de “banda larga”. Utilizando tais tecnologias, pequenas empresas, escritórios e usuários domésticos utilizando sistemas muitas vezes pouco protegidos são alvos preferenciais de invasão e exploração indevida. Estes sistemas acabam tornando-se potenciais participantes de DDoS (*Distributed Denial of Service* – Negação de Serviço Distribuída).

O comércio eletrônico e as transações financeiras via Internet exigem sistemas e recursos de rede confiáveis e de alta disponibilidade. Isso envolve a utilização de *hardware*, *software* e rede de última geração, os quais devem estar integrados de maneira correta, o que leva ao dispêndio de recursos financeiros volumosos e que certamente necessitam um alto nível de retorno às corporações.

Para garantir o bom uso destes recursos torna-se cada vez mais comum nas empresas a utilização de protocolos como SSL (*Secure Socket Layer* – Camada de Conexão Segura), ferramentas como *firewall*, IDS (*Intrusion Detection System* – Sistema de Detecção de Intrusão), VPN (*Virtual Private Network* – Rede Privada Virtual) e a definição de políticas de segurança. Mas, estes mecanismos

tradicionais de segurança não são capazes de impedir a efetivação de todos os tipos de DDoS (Dittrich, 1999a; Dittrich, 1999b; Dittrich, 1999c; CERT, 1999a; CERT, 1999b; CERT, 1999c).

A referida integração correta entre os recursos envolve o estabelecimento de relações de confiança entre provedores de serviço, relações essas baseadas em SLAs (*Service Level Agreements* - Contratos de Níveis de Serviço) estabelecidos não só entre as corporações e seus fornecedores, mas também entre parceiros, como por exemplo, os provedores de acesso à Internet que devem implementar certas políticas de segurança. Portanto, deve-se considerar um novo foco para a segurança da informação, assim como diz Nakamura (2000)

> O foco de segurança agora está em selecionar os usuários que podem entrar na rede e selecionar os direitos e os níveis de acesso de cada usuário na rede, e a certeza de que eles estão fazendo aquilo que lhes são explicitamente permitidos passa a ser essencial.

Neste sentido, estratégias de autorização para utilização de recursos devem ser consideradas, bem como ferramentas que auditem e/ou limitem esta utilização em casos de abuso podem ser ótimas alternativas. Pode-se considerar ainda que muitos dos esforços e recursos gastos na detecção, prevenção e combate a DDoS poderiam ser economizados se fossem adotadas algumas medidas que serão abordados neste trabalho, algumas das quais bastante simples (Singer, 2000) (Ferguson e Senie, 2000), por parte de administradores de redes e ISPs (*Internet Service Providers* – Provedores de Serviço de Internet).

1.2 OBJETIVO

O objetivo desta dissertação é o estudo comparativo de diferentes abordagens para prevenção e combate ao problema de DDoS. Tais abordagens propõem modelos e mecanismos para minimizar a exploração da infra-estrutura da Internet em ocorrências de DDoS, tanto para ocorrências por congestionamento da banda do enlace de acesso à Internet, quanto para aquelas que consumam totalmente os recursos de servidores.

Para atingir este objetivo é realizada uma avaliação da implementação de um protótipo baseado no modelo proposto por Meylan (2003) para demonstrar uma alternativa prática de combate ao problema.

Um panorama da problemática que envolve os ataques de DDoS também será apresentado, por meio da investigação das técnicas mais comuns de perpetração de DoS e DDoS, além do relato das suas mais notórias ocorrências e tentativas de ataques.

1.3 JUSTIFICATIVA

Ocorrências de DDoS são conhecidas e diversas já foram relatadas em vários sítios muito divulgados e acessados na Internet (Nuttall, 1999), bem como contra serviços de suma importância para o funcionamento adequado da Internet mundial (Marsan, 2002). Esses ataques resultaram na indisponibilidade de serviço nestes sítios ou na degradação dos serviços.

Organizações que tem o seu foco de atuação completamente baseado na Internet ou, que consideram este um importante canal de relação com seus clientes e parceiros, têm grande interesse em manter a disponibilidade de serviços, mesmo quando estiverem sofrendo com ocorrências de DDoS. Exemplos deste tipo de organização são bancos, sítios de comércio eletrônico, provedores de conteúdo e empresas especializadas em propaganda e publicidade pela Internet.

Em alguns países onde apostas e jogos de azar são legalizados, sítios de Internet movimentam bilhões de dólares anualmente. Recentemente, alguns destes sítios têm sido alvos de extorsões e tais ameaças são feitas com base nas possibilidades de perpetração de ataques de DDoS (Wearden, 2004 e Swartz, 2004).

Para suprir as necessidades de alta disponibilidade destes tipos de negócios, volumosos recursos financeiros são investidos em infra-estruturas com alta redundância e grandes possibilidades de rápido crescimento. Negócios foram criados nos últimos anos com foco neste segmento, como é o caso dos IDCs (*Internet Datacenters* – Centros de hospedagem de sítios de Internet*)* e empresas como Akamai que prometem 100% de disponibilidade para seus clientes, mesmo em caso de ocorrências de DDoS (Akamai, 2001), o que na prática parece não se refletir (Netcraft, 2004a).

Modelos como os selecionados para avaliação neste trabalho podem tornar-se soluções viáveis, independentes ou integradas, para não somente detectar ocorrências de DDoS, mas também para criar uma infra-estrutura resiliente

e que minimize os problemas oriundos destas ocorrências, como por exemplo a indisponibilidade total de serviços.

Na Figura 1, observa-se um esquema típico de um ataque DDoS.

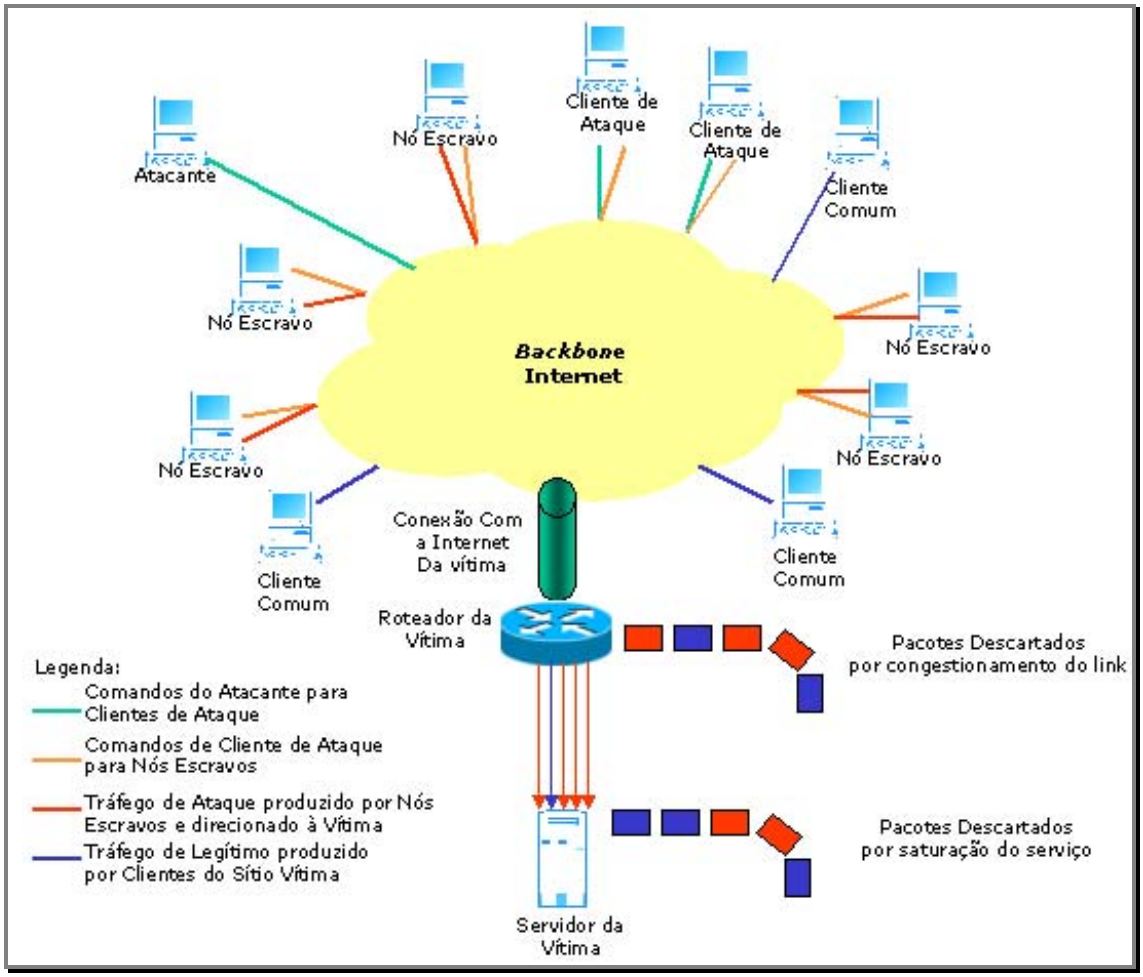

Figura 1 - Esquema de um Ataque DDoS

1.4 TRABALHOS CORRELATOS

Muitos são os modelos propostos que tentam enfrentar o problema de DDoS. A seguir alguns dos modelos identificados são relacionados, bem como suas principais características. Mais adiante será feita uma análise comparativa destes modelos e mecanismos.

Objetivando o aprofundamento do entendimento, os mecanismos propostos nos trabalhos correlatos analisados nesta dissertação foram classificados com base na taxonomia proposta por Mirkovic (2003), a qual busca classificar cada abordagem com relação aos seguintes critérios de classificação:

- Nível de atividade (NA) – Neste critério é possível diferenciar o mecanismo entre preventivo e reativo;
- Objetivo da Prevenção (OP) – Como os mecanismos podem executar medidas defensivas sozinhos ou em conjunto com outras entidades na Internet é possível diferencia-los entre autônomos, cooperativos e interdependentes;
- Objeto Assegurado (OA) – Identifica qual objeto será assegurado pelo mecanismo;
- Método de Prevenção (MP);
- Estratégia de detecção de ataque (EDA);
- Especificação de comportamento normal (ECN);
- Grau de Cooperação (GC);
- Localização de Distribuição (LD) – Os mecanismos podem ser diferenciados também de acordo com a localização onde são implementados, a qual pode ser na vítima, em uma localização intermediária ou na rede de origem do ataque;
- Estratégia de Resposta a Ataque (ERA).

A Figura 2 mostra a hierarquia proposta por Mirkovic (2003) para esta classificação.

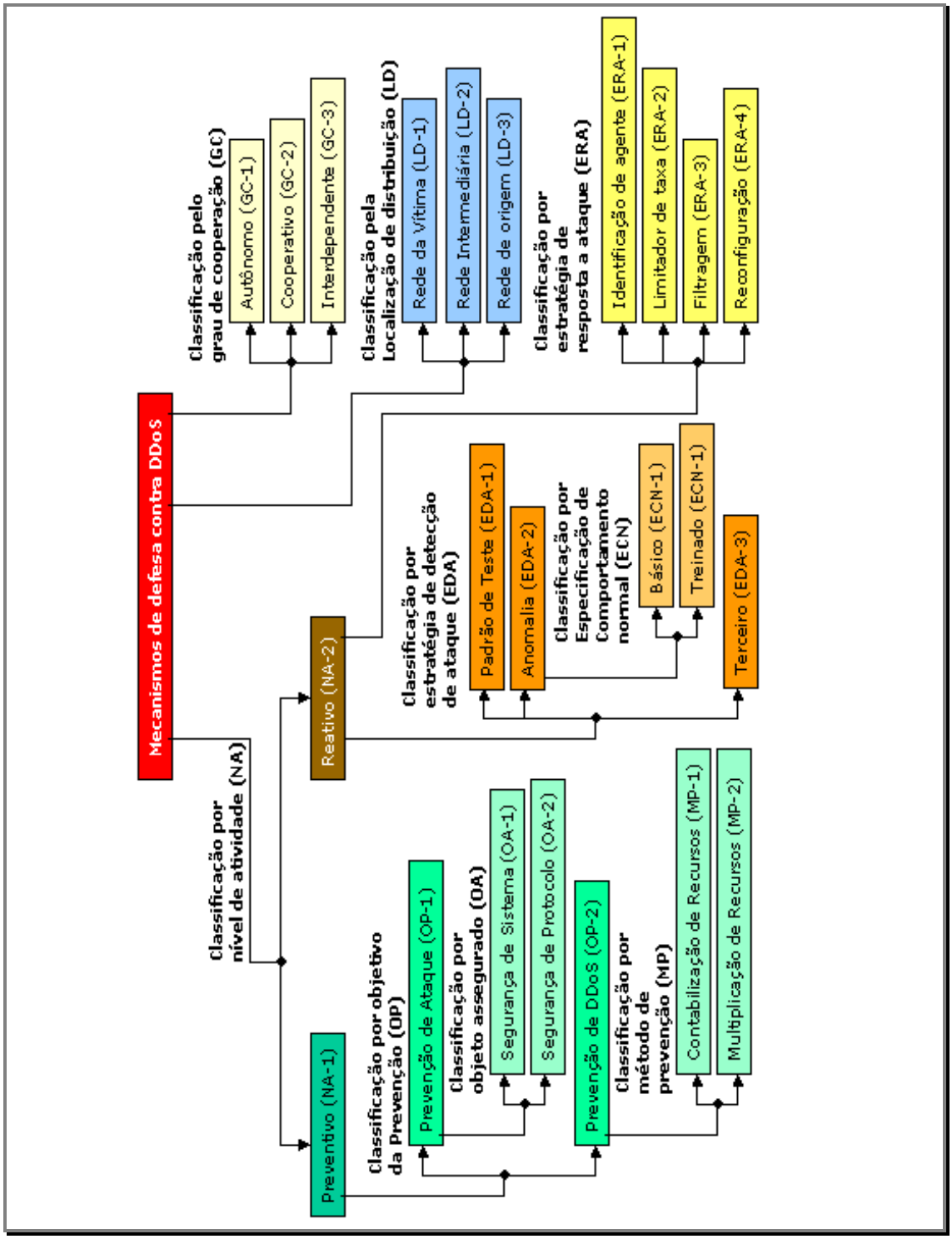

Figura 2 - Taxonomia dos mecanismos de defesa contra ataques de DDoS (Mirkovic, 2003)

1.4.1 QUALIDADE DE SERVIÇO COMO FERRAMENTA DE SEGURANÇA

Classificação conforme proposto por Mirkovic (2003): NA-1:OP-2:MP-1 e LD-1 (na classificação original AL-1:PG-2:PM-1 e DL-1).

Com o objetivo de diferenciar o tráfego recebido por determinados sítios, pode-se estabelecer uma relação de confiança entre ISPs e estes sítios, implementando a diferenciação de tráfego com requisição de QoS (*Quality of Service* - Qualidade de Serviço) para as conexões originadas em provedores "confiáveis". Desta forma, os fluxos de pacotes podem ser diferenciados no *firewall*, nos roteadores ou nos balanceadores de carga, direcionando-os para grupos de servidores divididos, em servidores dedicados ao tráfego "confiável" e servidores dedicados ao tráfego genérico. (Figura 3)

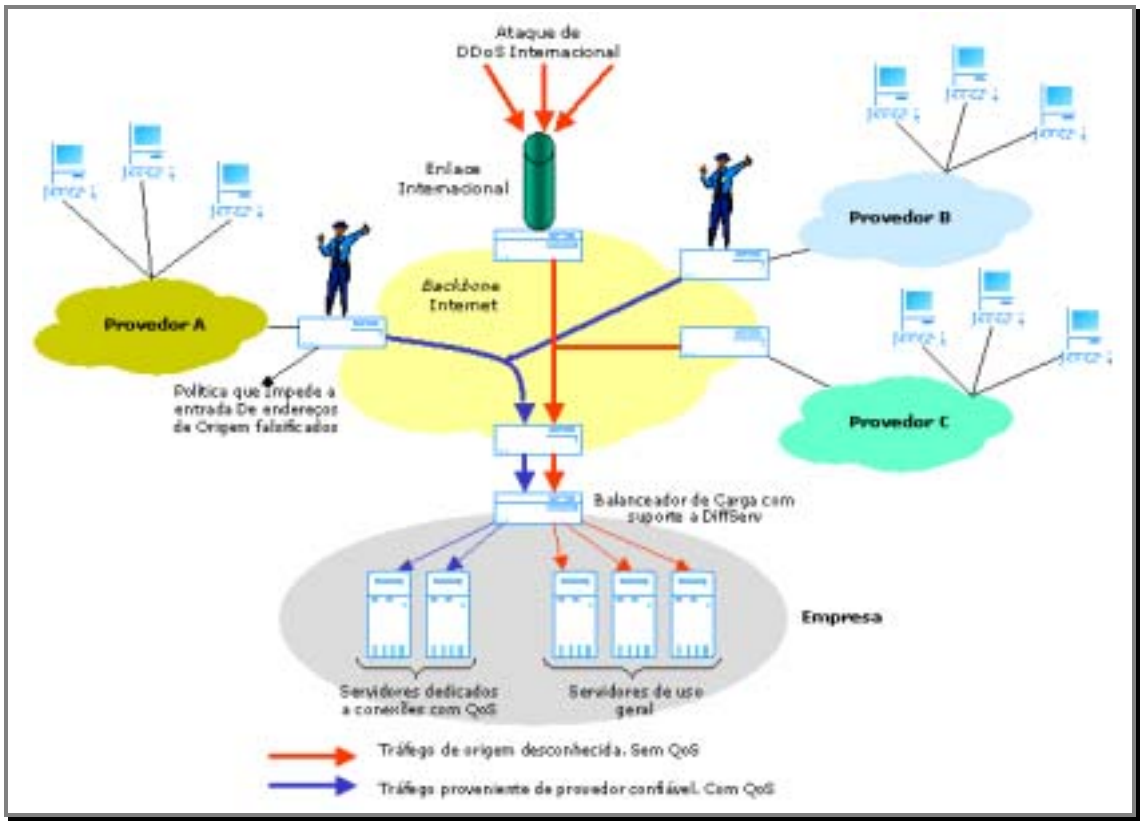

Figura 3 - Modelo de Utilização de QoS como Ferramenta de Segurança (Meylan, 2003)

Este modelo proposto por Meylan (2003) sugere a utilização de IDSs nas tarefas de identificação de ataques DDoS e na inicialização do processo de

diferenciação de tráfego. Isto resultaria em uma utilização racional dos recursos servidores disponíveis em situações de ausência de ataque.

A proposição busca minimizar os efeitos não somente de ocorrências de DDoS que tenham como objetivo o congestionamento da conexão (ou conexões) da rede com a Internet, como também evitar o consumo total dos recursos servidores disponíveis durante uma ocorrência, uma vez que DDoS bem sucedido pode ser perpetrado contra um sítio na Internet sem que para isso seja necessário consumir toda a banda de conexão que tal sítio possua com Internet.

Recomenda-se neste trabalho a caracterização dos provedores "confiáveis" baseando-se na exigência de implementação de medidas de segurança que minimizem as possibilidades de participação de clientes destes provedores em ataques de DDoS ou ainda baseando-se na utilização de sistemas especializados nesta tarefa como é o caso do D-WARD descrito e experimentado em Mirkovic, Prier e Reiher (2002) e Mirkovic (2003), o qual também é avaliado mais adiante neste trabalho.

A composição deste modelo envolve:

- Provedores confiávies – caracterizados de acordo com os critérios descritos anteriormente;
- Utilização de QoS – os provedores de backbone devem habilitar o suporte a DiffServ;
- Provisionamento de recursos computacionais – implementação de recursos destinados ao tráfego genérico e recursos dedicados ao tráfego com QoS;
- Balanceador de carga com suporte a QoS – especificamente com condições de distinguir tráfego com QoS baseado em Diffserv, o qual seja capaz de direcionar tráfego para recursos computacionais adequados a cada tipo de tráfego;
- IDS – proposto no funcionamento do sistema, este componente seria responsável por sinalizar o balanceador para situações de ataque, quando então o tráfego começaria a ser diferenciado e direcionado ao recurso computacional específico.

Sugere-se a utilização de marcação de pacotes com DSCP AF (DS *Code Point Assured Forwarding*) (Heinanen *et al.*, 1999), para os quais sejam implementados garantia de banda e controle de atraso em todo o caminho. Mas, questões vinculadas à forma de implementação prática, segurança do modelo, entre outras, permanecem em aberto.

Empresas que pretendam atingir uma grande parte de seus clientes com a implementação deste modelo deverão incentivar tais clientes a utilizar seus parceiros “confiáveis”.

Por depender da ativação do suporte a DiffServ (*Differentiated Services* – Serviços Diferenciados) (Blake *et al.*, 1998; Nichols *et al.*, 1998) para suportar as requisições de QoS e a classificação do tráfego no caminho completo por onde trafegarem os pacotes, sua implementação pode tornar-se cara, pois os provedores podem considerar um serviço diferenciado a ativação destes recursos e pretender cobrar custos mais elevados para prover isso.

Esta questão também demonstra uma necessidade de complementação da taxonomia proposta por Mirkvic (2003) quanto à Localização de Distribuição, pois este modelo depende da ativação de Diffserv em todo o caminho por onde os pacotes trafegarão, portanto seria interessante incluir nesta categoria uma classificação nomeada de Caminho Completo (*Complete Path* para o original) com a abreviação LD-4 (DL-4 para o original).

Sítios com freqüência distribuída internacionalmente também teriam dificuldades para implementar esse modelo em função da necessidade do estabelecimento das relações de confiança com muitos provedores. Já para sítios com freqüência em sua maioria restrita a uma região geográfica coberta por poucos provedores teriam maiores facilidades nesse sentido.

1.4.2 O AMBIENTE VIPNET

Classificação conforme proposto por Mirkovic (2003): NA-1:OP-2:MP-1 e LD-1 (na classificação original: AL-1:PG-2:PM-1 e DL-1).

O ambiente VIPNet é descrito, bem como os resultados de implementação experimental desenvolvida em Brustoloni (2002). Esta proposta procura implementar um modelo semelhante ao proposto por Meylan (2003) em função do uso dos recursos de qualidade de serviço, mas o ambiente VIPNet considera principalmente ocorrências de DDoS por congestionamento da conexão de um determinado sítio da internet. O modelo exige ainda a autenticação do cliente no ambiente e a aquisição prévia de recursos por parte deste cliente para utilização das características de qualidade de serviço do ambiente.

O modelo baseia-se no estabelecimento de relações de confiança em uma infra-estrutura que utiliza a requisição de qualidade de serviço para a

diferenciação do tráfego recebido pela rede incluída nesta infra-estrutura. O objetivo principal é manter disponível banda suficiente para tratar tráfego diferenciado mesmo quando sob ocorrência de DDoS de congestionamento da conexão com a Internet.

Os direitos que os usuários deste ambiente devem adquirir são limitados com relação ao tempo e à quantidade, para evitar abusos e ainda o roubo destes direitos por usuários mal-intencionados. Este recurso sugere a possibilidade de sua cobrança por parte dos sítios que implementem essa arquitetura.

Os *VIP Gates* são componentes deste ambiente, os quais são dispositivos destinados a monitorar e marcar pacotes originados em clientes que possuam direitos para utilizar o ambiente. Outro componente, opcional, chamado *VIP Monitor* pode ser utilizado para controle de sessões e priorização dinâmica. Os participantes deste ambiente comunicam-se por meio de um *VIP protocol*, protocolo próprio utilizado para funções específicas do ambiente como contabilidade de direitos, requisições, instalação e ativação. (Figura 4)

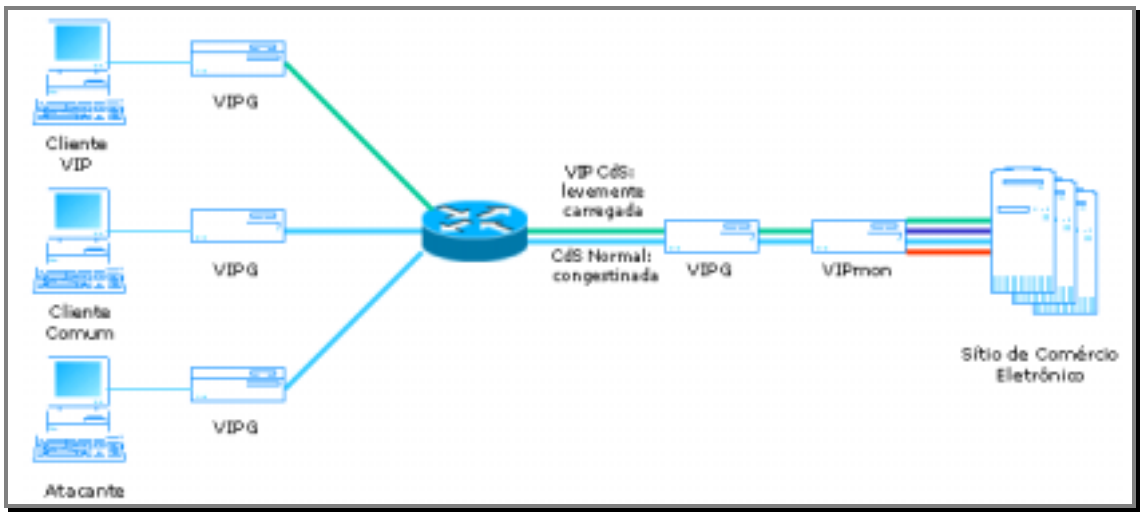

Figura 4 - Tratamento diferenciado dos clientes no ambiente VIPNet (Brustoloni, 2002)

A grande vantagem deste modelo certamente é a manutenção de disponibilidade para determinados clientes, mesmo quando o sítio estiver sofrendo com ocorrências de DDoS. Mas, a contabilização do uso dos recursos pode ser um requisito importante em sítios que ofereçam acesso e tarifação diferenciados.

Como a utilização dos recursos propostos não é absolutamente transparente para o usuário final, pode tornar-se de difícil implementação. Os componentes proprietários também podem dificultar a sua implementação.

1.4.3 CONTROLE DE AGRUPAMENTOS DE GRANDE LARGURA DE BANDA

Classificações conforme proposto por Mirkovic (2003): NA-2:EDA-2, NA-2:ERA-2, GC-2 e LD-2 (na classificação original: AL-2:ADS-2, AL-2:ARS-2, CD-2 e DL-2).

Mecanismos a serem implementados em roteadores para identificação e agrupamento de padrões de tráfego são definidos por Mahajan (2001). Esses mecanismos identificam tráfego de possíveis rajadas (*flash crowds*[1]) ou ataques de negação de serviço (DoS) e são chamados de ACC (*Aggregate-based Congestion Control* – Controle de Congestionamento Baseado em Agrupamentos). O objetivo principal é identificar e controlar agrupamentos que consumam grande quantidade de banda dos roteadores.

Nos roteadores esses agrupamentos são criados por pacotes de um ou mais fluxos que possuem uma propriedade em comum, por exemplo, como endereço de origem ou endereço de destino, pacotes TCP, pacotes ICMP (*Internet Control Message Protocol* – Protocolo de Controle de Mensagens Internet) tipo ECHO, etc.

Complementarmente e objetivando a limitação de tráfego como solução dos problemas abordados, dois mecanismos são propostos, o primeiro para controlar tráfego local (*Local ACC*) e o segundo, denominado *pushback, a ser* utilizado de forma cooperativa entre roteadores adjacentes que estejam participando de fluxos identificados como rajadas ou ataques de negação de serviço.

Estes mecanismos propõem-se à tomada de algumas decisões relativas à limitação de tráfego por agrupamento em cada roteador ou dispositivo conectado e ainda solicitar aos roteadores adjacentes a limitação de tráfego para os fluxos identificados.

As decisões são inicialmente baseadas no nível de comprometimento do roteador ou dispositivo, em função de congestionamento, como mostrado na Figura 5. E para constatar este comprometimento os autores propõem o monitoramento constante do descarte de pacotes em cada fila e a aplicação de um limiar baseado em uma política pré-definida.

---

[1] Quando uma grande quantidade de usuários tenta acessar um sítio na Internet ao mesmo tempo, em função de um grande evento, provocando um tráfego tão grande que chega a interferir no tráfego de outros usuários que não estejam acessando este sítio.

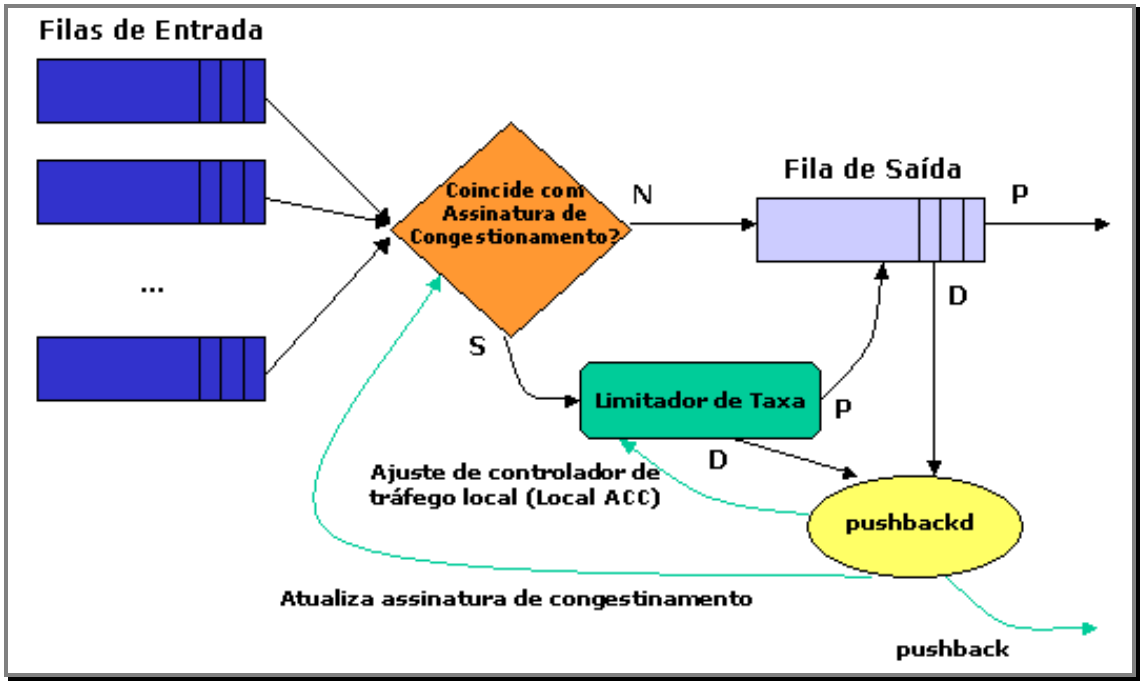

Figura 5 - **Visão parcial de um roteador com a implementação de *Pushback* (Ioannidis & Bellovin, 2002)**

Conforme o modelo proposto seria possível implementar o mesmo mecanismo de *pushback* em servidores que possuíssem capacidade de detecção de ataques. Desta forma, pode-se concluir que seria viável também integrar o mesmo mecanismo a sistemas de detecção de intrusão que possuam essa mesma capacidade, o que poderia resultar em um mecanismo mais eficiente em implementações que possuam vários servidores, por exemplo, como em uma DMZ (*Demilitarized Zone* - Zona Desmilitarizada).

A identificação e construção destes agrupamentos é o ponto chave desse trabalho, bem como as escolhas feitas para a criação destes agrupamentos, pois um mecanismo que falhe nesta implementação pode provocar o descarte de pacotes que não pertençam a um ataque ou a um congestionamento por rajada.

Um protótipo do mecanismo de *pushback* foi avaliado por Ioannidis & Bellovin (2002), com a implementação de roteadores utilizando o sistema operacional FreeBSD em conjunto com os recursos de dimensionamento de tráfego do IPFW, componente do núcleo deste sistema operacional e responsável pela filtragem e contabilização de pacotes.

Dentre os possíveis problemas na implementação deste modelo, já foi citada a política de implementação da criação dos agrupamentos de fluxos que,

ainda pode ser negativamente agravada se roteadores adjacentes definirem políticas diferentes para essa criação. Além disso, relações de confiança entre roteadores adjacentes poderiam comprometer a funcionalidade de toda a infra-estrutura se um ou mais de seus participantes, por qualquer motivo, tiver comportamento diferenciado em relação aos demais.

Em comparação aos modelos baseados em traçado de rota de ataques, este modelo é muito mais efetivo, pois implementa medidas de limitação de tráfego que podem diminuir o impacto sobre a infra-estrutura durante ocorrências de DDoS.

1.4.4 DEFESA CONTRA ATAQUES DDOS A PARTIR DA ORIGEM

Classificação conforme proposto por Mirkovic (2003): NA-2:ERA-2 e LD-3 (na classificação original: AL-2:ARS-2 e DL-3).

Mirkovic, Prier e Reiher (2002) e Mirkovic (2003) propõem um modelo baseado na detecção e bloqueio de ataques DDoS próximo à origem dos mesmos.

Os objetivos do sistema chamado D-WARD são:

- Detectar ataques de DDoS provenientes da rede que o sistema esteja monitorando e pará-los por meio do controle de tráfego de saída direcionado à vítima.
- Fornecer bons serviços para legitimar o tráfego entre a rede que o sistema esteja monitorando e a vítima enquanto o ataque estiver em curso.

A detecção e controle são realizados nos roteadores de saída das redes para a Internet, procurando evitar portanto a utilização destas redes na inicialização ou participação em ataques. Conseqüentemente, evita-se também o consumo de recursos no núcleo da Internet e a necessidade e o custo da implementação de uma maior cooperação entre roteadores neste núcleo.

O sistema de defesa proposto no projeto D-WARD, cuja arquitetura esta representada na Figura 6, deve ser implementado em todos os roteadores que provêm saída para a Internet, como ISPs. O sistema monitora os fluxos originados apenas em sua rede, classificando-os de acordo com padrões de tráfego e taxas limite para cada tipo de protocolo da camada de transporte, como TCP e UDP.

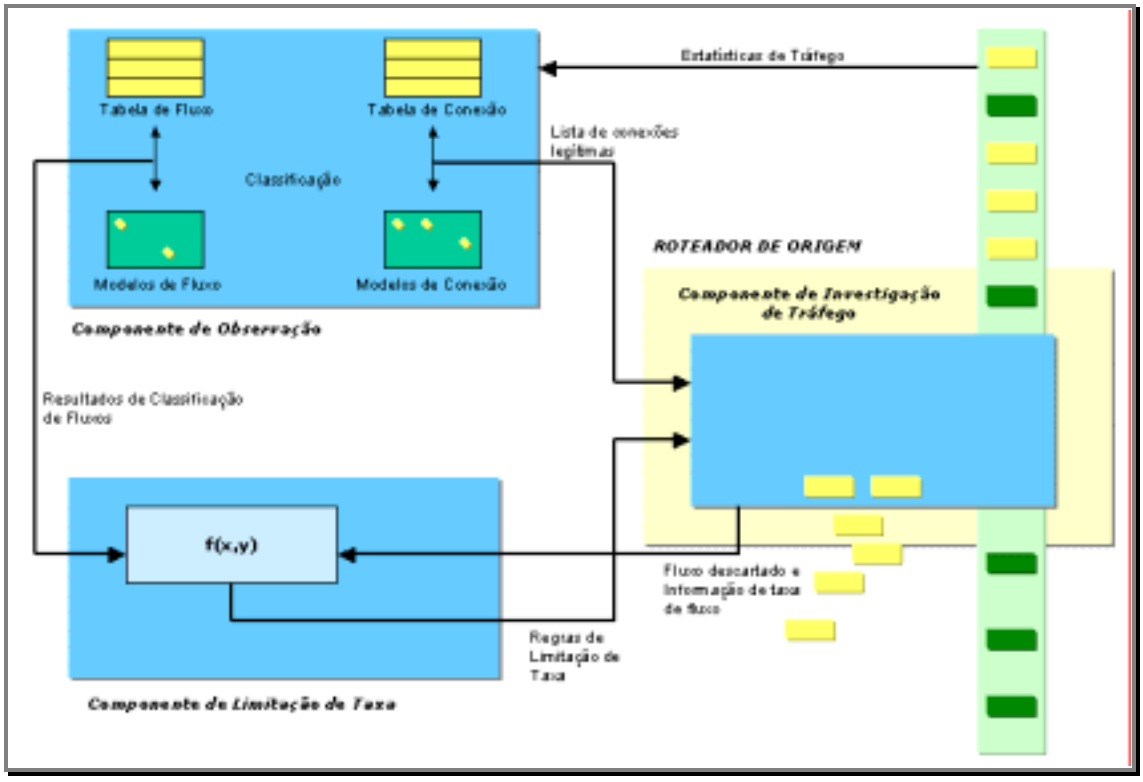

Figura 6 - A Arquitetura D-WARD
(Mirkovic, 2003)

Uma classificação dos fluxos observados é implementada dentro de um intervalo de tempo, comparando estes fluxos com padrões normais de tráfego e atribuindo-os as classificações normais, suspeitos ou ataque.

Baseado então no recurso de monitoramento o sistema responde aos ataques, limitando o tráfego para os fluxos classificados como suspeitos ou ataque.

Em TCP, em função de ser um protocolo orientado à conexão, são levadas em consideração as características que demonstram estar havendo um tráfego controlado entre origem e destino, o que deve incluir a aceitação do tráfego pelo destino. O tráfego é considerado pertencente a um ataque se extrapolar um limite máximo pré-definido (TCP_{rto}) de pacotes enviados e recebidos.

Mesmo evitando o uso de uma rede na perpetração de ataques de DDoS fazendo uso de endereçamento falsificado, este mecanismo provavelmente não evitaria o uso desta rede como refletora no caso de um ataque iniciado em uma rede que não implemente esse mecanismo. Também não é completamente efetivo quando há tráfego criptografado.

Os autores implementaram um protótipo utilizando Linux e o avaliaram sob diversas condições de ataque DDoS. Seus resultados estão descritos em Mirkovic, Prier e Reiher (2002).

Na avaliação feita, referente ao custo de implementação, detectou-se que pela necessidade da passagem dos fluxos pelo módulo de limitação de tráfego, aconteceu um retardamento nestes fluxos, o que no referido protótipo atingiu entre 1 e 10 μs mas, como algum tempo é gasto na camada de aplicação procurando manter as tabelas dedicadas à contabilização de conexões, um retardamento adicional acontece e pode chegar a 83 μs durante operações normais e em condições grandes cargas de fluxos atingir 1 ms.

Em Mirkovic (2003) foi realizado novo experimento quando chegou-se à conclusão de que o retardamento médio atingia 6 μs, tomando-se por base testes com pacotes ICMP ECHO. Apenas citou-se que na ocorrência de endereços forjados, uma carga de processamento maior existia no nível de núcleo do sistema e que tal carga tornava-se crítica sob taxas de processamento superiores a 12.000 pacotes por segundo.

Ainda segundo Mirkovic (2003), vários experimentos com soluções cooperativas utilizando o modelo proposto estão em andamento e as avaliações preliminares demonstram nível de serviço melhor quando utilizam o sistema D-WARD.

Para tornar-se efetivo este modelo deveria ser implementado pela grande maioria dos ISPs, o que resultaria em grandes restrições para que atacantes perpetrassem ataques de negação de serviço. Mas, se fosse considerada uma exigência, poderia não ser seguida por todos os países e ISPs. O grau de dificuldade de implementação em roteadores reais ainda não foi avaliado, mas certamente isso dependeria do comprometimento na implementação deste recurso por alguns dos grandes fabricantes desses equipamentos.

1.4.5 SOS: SECURE OVERLAY SERVICES

Classificação conforme proposto por Mirkovic (2003): GC-3 (na classificação original: CD-3).

O estudo elaborado por Keromytis, Misra & Rubenstein (2002) considera que, mesmo quando há ocorrência de catástrofes naturais ou ataques terroristas, pelo menos alguns usuários devem ter acesso privilegiado aos recursos

das redes. O principal objetivo da arquitetura SOS é prover comunicação entre uma origem autenticada e autorizada e um destino.

A principal motivação é a manutenção do funcionamento de ERTs (*Emergency Respose Teams* – Grupos de Atendimento de Emergências), como por exemplo, departamentos de polícia, bombeiros, defesa civil, entre outros. Mas, pode ser implementado sob diversos tipos de política de privilégios ou priorização, como a garantia de acesso aos sítios de bancos por parte de seus clientes ou aos sítios de notícias por parte de seus assinantes, entre outras. O suporte a usuários móveis também foi considerado na proposição.

Uma camada sobreposta de rede é proposta neste trabalho, a qual objetiva criar uma rede segura sobre uma infra-estrutura de rede IP. Esta rede segura também evita que serviços essenciais sejam atingidos por instabilidades ou indisponibilidades provocadas por ataques do tipo DoS.

A infra-estrutura baseia-se na autenticação e autorização prévias de tráfego entre um usuário confirmado ou aceito e um destino. Esta tarefa é feita pelo SOAP (*Secure Overlay Access Point* – Ponto de Acesso à Camada de Segurança Sobreposta), onde o usuário deve autenticar-se. Os autores indicam a utilização de protocolos seguros como IPsec (*IP Security Protocol* – Protocolo IP Seguro) ou TLS (*Transport Layer Security* - Segurança Da Camada De Transporte) para implementar esse procedimento de autenticação. Diversos SOAPs devem fazer parte da infra-estrutura, o que a reforça contra ataques, mas aumenta o tráfego de informações entre esses componentes. A definição de um esquema sobre a escolha dos pontos de acesso foi deixada para trabalhos futuros.

Uma vez identificado o tráfego como pertencente ou não da infra-estrutura SOS, decisões de filtragem de tráfego (como em um *firewall*) podem ser tomadas por parte de componentes chamados de baliza (*beacon*). Essas decisões resultarão em encaminhamento, descarte ou limitação de tráfego, garantindo assim uma priorização de tráfego para usuários da infra-estrutura. Essa filtragem deve ser feita da mesma forma por diversos nós para possibilitar redundância e maior segurança.

As *servlets*[2] secretas são responsáveis em última instância por encaminhar o tráfego para o destino desejado. Elas atuam como *proxies* ocultos

[2] Programas (geralmente em linguagem Java) que residem e são executados em um servidor para prover funcionalidade ao servidor ou processar dados no servidor.

neste processo, possibilitando aos roteadores próximos ao destino que implementem filtragem aceitando tráfego somente destas *servlets* secretas.

Um esquema próprio de roteamento, focado em decisões aleatórias de encaminhamento, visa aumentar a dificuldade de um atacante em prever o funcionamento da infra-estrutura.

O aumento no número de nós componentes da infra-estrutura reforça ainda mais sua capacidade de resistir a um ataque de DDoS, pois para que um atacante comprometa a arquitetura seria necessário conhecer todos os nós e atacá-los praticamente simultaneamente. Mas isso também aumenta consideravelmente suas exigências administrativas, o que pode ser considerado um ponto negativo para a sua implementação em larga escala.

A redundância proposta na arquitetura é um dos seus pontos fortes, pois além de garantir a comunicação, mesmo com alguns dos participantes da infra-estrutura comprometidos, possibilita ainda que os próprios componentes excluam outros de sua relação de confiança ou mesmo se retirem da infra-estrutura em situações de comprometimento por ocasião de ataques. Mas essa redundância só pode ser alcançada com a participação de um grande número de componentes de cada tipo. A Figura 7 mostra a arquitetura básica do modelo SOS.

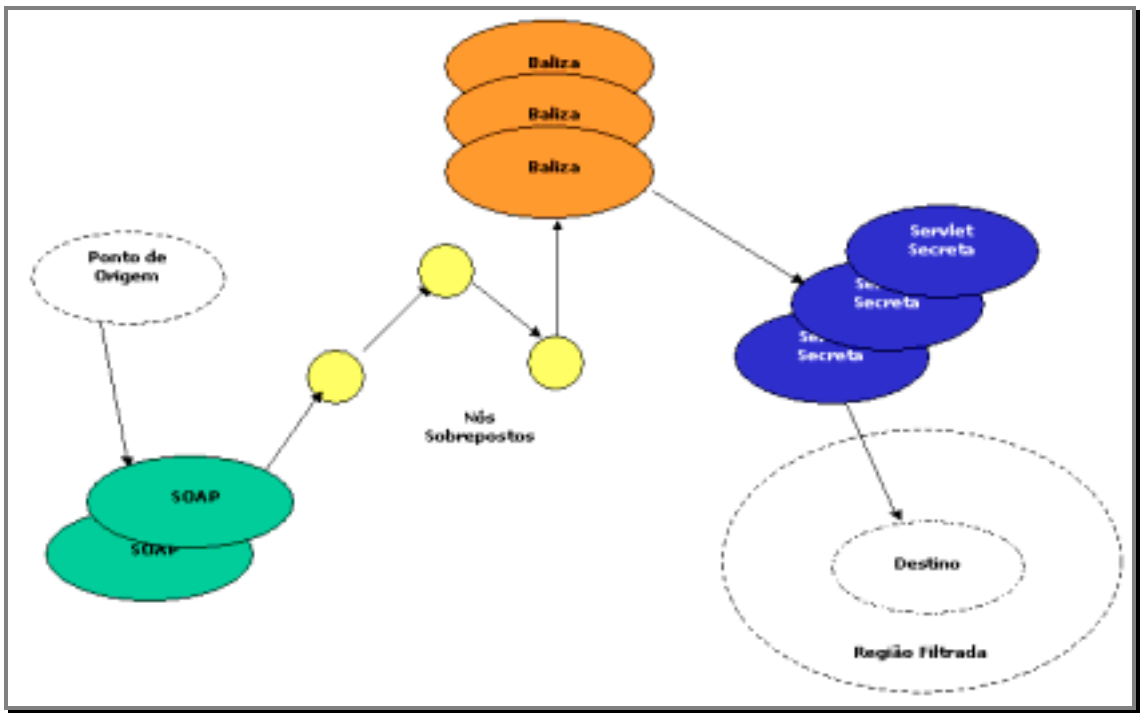

Figura 7 - Arquitetura Básica do SOS
(Keromytis, Misra & Rubenstein, 2002)

Pode-se considerar uma deficiência a necessidade de alteração e notificação do esquema de filtragem próxima aos nós de destino, quando da alteração de endereçamento IP deste nó (ocorrida pela movimentação do nó em uma rede sem fio ou pela mudança de ISP).

WEBSOS

Em Cook *et al.* (2003) uma implementação baseada no modelo SOS é apresentada. Denominada WebSOS, cuja arquitetura esta representada na Figura 8, o principal objetivo de sua infra-estrutura é distinguir tráfego entre autorizado e não autorizado.

Os nós que compõem esta infra-estrutura são *proxies* destinados a propósitos específicos e que possuem capacidades de autenticação baseada em certificação digital de chaves públicas, manipulação de tráfego em conexões SSL, além das funcionalidades de baliza e *servlet*, propostas no modelo SOS.

Uma implementação experimental também foi avaliada no referido trabalho e demonstrou uma carga adicional significativa com a utilização da infra-estrutura sobreposta. Conforme relatado no trabalho o aumento da latência no tráfego fim-a-fim variou entre fatores de 5 a 10, mas isto foi considerado aceitável levando-se em consideração a possibilidade de manter a disponibilidade de serviços mesmo sob DDoS.

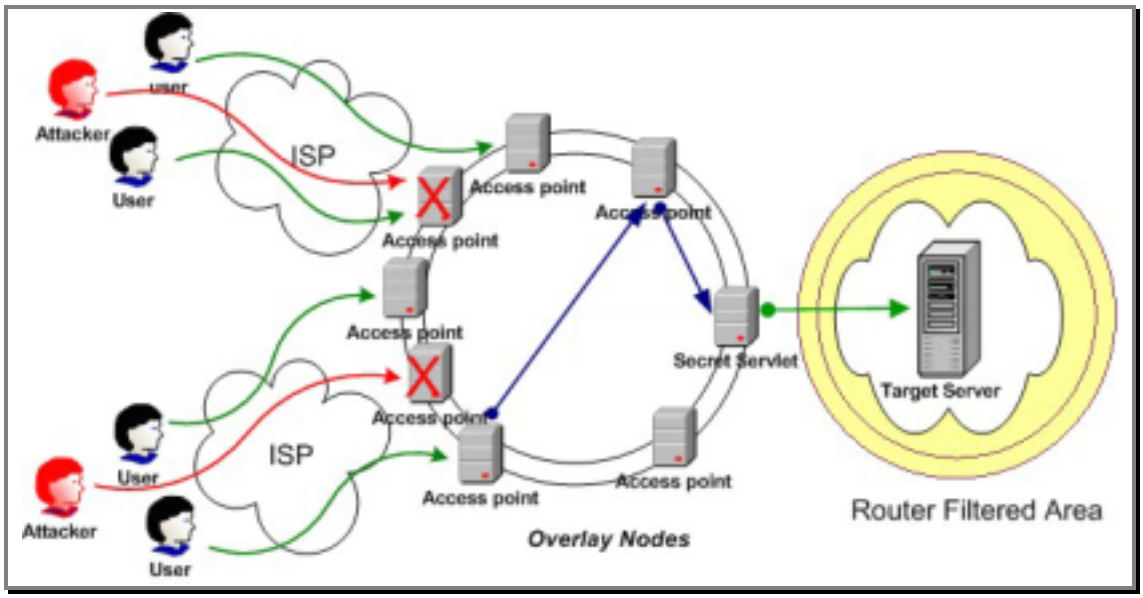

Figura 8 - Arquitetura WebSOS sob ataque DDoS (Cook *et al.*, 2003)

1.4.6 PROTOCOLO DE VALIDAÇÃO FORÇADA DE ENDEREÇO DE ORIGEM

Classificação conforme proposto por Mirkovic (2003): NA-2:ERA-1 (na classificação original: AL-2:ARS-1).

A proposição feita em Li *et al.* (2001) define o protocolo SAVE (*Source address validity enforcement protocol* – Protocolo de validação forçada de endereço de origem) de filtragem de endereço de origem para validar o tráfego em roteadores, o qual procura evitar o tráfego de pacotes que estejam utilizando endereços IP forjados (*spoofing*). A arquitetura deste protocolo é mostrada na Figura 9.

Muitos dos ataques distribuídos de negação de serviço utilizam esse tipo de estratégia para evitar a identificação de sua origem. Outra possibilidade é que estes ataques estejam sendo iniciados com a utilização de máquinas agindo como refletores, isto é, o ataque se inicia com o endereço forjado da vítima e direcionado a sítios da Internet que gerarão respostas de conexões à vítima em grande quantidade provocando a negação de serviço.

A implementação envolve a construção de uma tabela em cada roteador, a qual estabelece uma relação entre endereços de origem e interfaces, onde pacotes provindos destes endereços de origem poderão ser recebidos. Esta tabela é construída e atualizada a partir das tabelas de encaminhamento criadas pelos protocolos de roteamento.

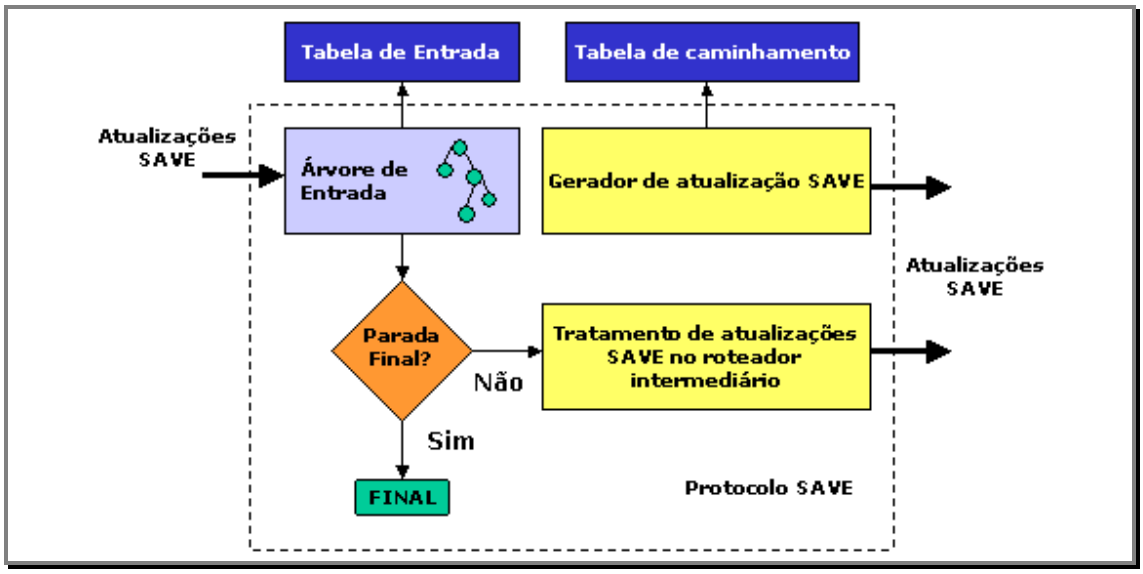

Figura 9 - Arquitetura do Protocolo SAVE (Li *et al.*, 2001)

Conforme o artigo, outras vantagens podem ser obtidas com a implementação do protocolo, tais como simplificação de mecanismos de Detecção de Intrusão ou de Diagnóstico de Redes, entre outros.

Como qualquer nova definição de protocolo, são abordados os diversos aspectos relativos ao projeto e implementação. São considerados os aspectos como crescimento em escala, coexistência com dispositivos que não implementem tal protocolo (dispositivos legados), além de simulações de implementações e avaliações das mesmas.

Alguns pontos foram deixados para trabalhos futuros: aspectos relativos à segurança do protocolo; a criação de atualizações SAVE agregadas; a implementação gradual de novos roteadores baseados em filtragem e a incorporação deste mecanismo de filtragem, juntamente com outras técnicas de rede como por exemplo, o tunelamento IP, PHBs (*Per-Hop Behavior* – Tratamento por Salto) em DiffServ (*Differentiated Services*) e o roteamento de múltiplos caminhos e MPLS (*Multi-protocol Label Switching* – Chaveamento por Rótulo para Múltiplos Protocolos).

Uma vez que esse trabalho introduz um novo protocolo, para que sua implementação em larga escala seja possível é necessário que os dispositivos o suportem para que se torne efetivo.

1.4.7 FILTRAGEM INTELIGENTE DE PACOTES BASEADA EM TRAÇADO DE ROTA IP

Classificação conforme proposto por Mirkovic (2003): NA-2:ERA-1 (na classificação original: AL-2:ARS-1).

O trabalho desenvolvido por Sung & Xu (2002) utiliza conceitos já estabelecidos de traçado de rota de tráfego IP para criar um modelo probabilístico que proteja redes contra ataques distribuídos de negação de serviço. Seu objetivo é criar uma política de descarte baseada em informações de probabilidades coletadas por meio de um algoritmo distribuído executado em diversos roteadores. Esta política prioriza o descarte de pacotes originados em rotas classificadas como “infectadas” e que possam estar sendo utilizadas para perpetrar um ataque de DDoS. A Figura 10 apresenta o modelo de roteador de perímetro considerado neste trabalho.

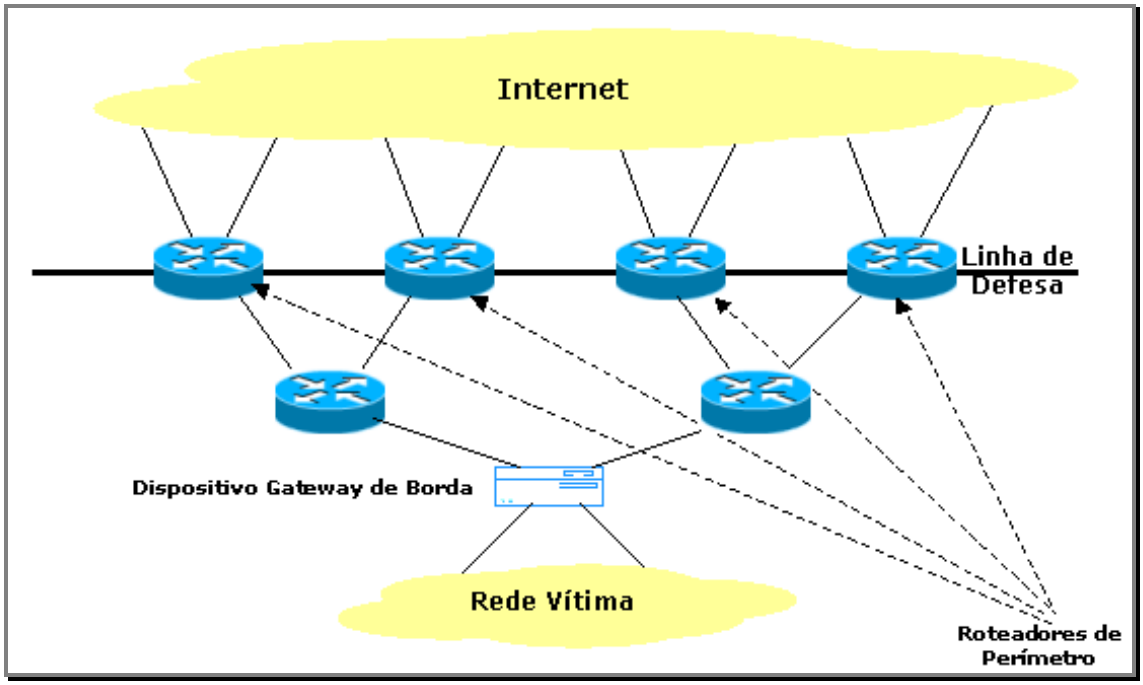

Figura 10 - O modelo de roteador de Perímetro (Sung & Xu, 2002)

Este modelo propõe um esquema composto pelos seguintes módulos:

- EPM (*Enhanced Probabilistic Marking* – Marcador Probabilístico Avançado) – Propõe-se que este módulo seja executado em cada roteador da infra-estrutura de Internet, mesmo não havendo um ataque de DDoS. Ele tem a função inserir uma marcação no cabeçalho IP, fazendo uso de campos não muito utilizados deste cabeçalho, como por exemplo, os campos usados para fragmentação. Dois tipos de marcação são utilizados, onde o primeiro tipo será utilizado pelo módulo AMD e o segundo pelo módulo PPF.
- AMD (*Attack Mitigation Decision-making* – Tomada de Decisão de Suavização de Ataque) – Este módulo é executado no dispositivo de borda (por exemplo, no firewall) do sítio da vítima e é responsável por reconstruir rotas de ataque e executar tomadas de decisões sobre a marcação probabilística dos pacotes.
- PPF (*Preferential Packet Filtering* – Filtro de Pacotes Preferenciais) – Deve ser executado em cada roteador de perímetro e é responsável pela filtragem diferenciada dos pacotes.

Segundo as avaliações contidas no trabalho, apesar de conseguir melhorar o tráfego “legítimo” de três a sete vezes durante um ataque, o modelo em

alguns casos acaba penalizando, mesmo que em pequeno número, fluxos que deveriam ser considerados “legítimos”.

A necessidade de implementação em todos os roteadores da infra-estrutura de Internet é outro ponto negativo, pois por tratar a marcação de pacotes de maneira muito particular sua efetividade torna-se bastante dependente de implementação em larga escala.

Segundo os autores a grande vantagem do modelo em relação a outros que implementam o traçado reverso de roteamento IP (IP *traceback*) encontra-se no algoritmo probabilístico e na marcação dos pacotes, o que possibilita um traçado muito preciso, uma vez que inclui o par de endereços de origem e destino.

1.4.8 ANÁLISE COMPARATIVA

Uma análise comparativa dos mecanismos propostos nos trabalhos correlatos apresentados será consolidada no Capítulo 3.

1.5 METODOLOGIA

Foi efetuado inicialmente um levantamento bibliográfico, o qual evidenciou a atualidade do problema de DDoS e a grande diversidade de mecanismos dispostos a minimizar os problemas causados por tal problema. Alguns destes mecanismos, apesar de restritos a situações bastante específicas, demonstram ser eficazes.

No estudo dos trabalhos correlatos, a identificação de algumas de suas vantagens e desvantagens possibilitou vislumbrar que a integração ou cooperação entre alguns mecanismos pode tornar-se muito eficaz na solução do problema genérico de DDoS, levando-se em consideração além de implementações de detecção e combate, algumas medidas de prevenção. Este aspecto foi explorado por Mirkovic (2003), mas ainda carece de implementações comerciais que possam ser adotadas em larga escala.

Dentre as abordagens avaliadas, o modelo proposto por Meylan (2003) foi implementado.

Um levantamento para identificar as possibilidades de implementação deste modelo foi realizado e escolheu-se para realizar o experimento *software* como Linux, iptables (Russel & Welte, 2002), iproute2 (Hubert *et al.*, 2003), *patch*

IMQ (*Intermediate Queueing Device* – Dispositivo Intermediário de Enfileiramento) e LVS (*Linux Virtual Server* – Servidor Virtual Linux) (Zhang, 2000).

Esta implementação é avaliada neste trabalho e inclui os seguintes aspectos:

- Comparação do mecanismo em situação de ataque sem a sua utilização;
- Comportamento do mecanismo quando da sua ativação e desativação;
- Taxa de vazão entre os componentes.

Um comparativo desta implementação com outra abordagem é apresentado na conclusão deste trabalho.

1.6 CONTRIBUIÇÕES

Este trabalho mostra o atual desenvolvimento dos mecanismos para prevenção de ataques distribuídos de negação de serviço nas suas mais diversas abordagens e ainda uma avaliação de alguns modelos propostos. Nesta avaliação comparativa foram elaboradas proposições para melhor aproveitamento dos mecanismos.

Ao final de uma análise mais aprofundada de alguns modelos e mecanismos foi possível sugerir soluções cooperativas, visto que a diversidade de abordagens inicialmente direciona para a busca desta integração, a qual possa combater com maiores chances de êxito o problema da DDoS em alguns casos.

Também contribui este trabalho com a implementação experimental de um protótipo baseado no modelo proposto por Meylan (2003) e uma avaliação do mesmo sob ataque simulado. Foram identificadas também possibilidades para trabalhos futuros que explorem principalmente o uso da qualidade de serviço como ferramenta de segurança no combate ao problema de DDoS.

1.7 SUMÁRIO ESTRUTURADO

Cinco capítulos compõem esta dissertação, a qual foi estruturada de maneira a apresentar um panorama sobre o problema dos ataques distribuídos de negação de serviço, os mecanismos para combatê-lo e uma análise comparativa dos mesmos, além de uma implementação experimental de um dos mecanismos abordados.

No Capítulo 1, mecanismos específicos abordados em trabalhos correlatos para combate do problema foram apresentados.

O Capítulo 2, apresentará o problema enfocando as técnicas atualmente utilizadas para perpetração de ataques, bem como alguns ataques notórios e de grandes impactos econômicos.

As maneiras de combater o problema, com soluções simples ou mais elaboradas, serão apresentadas no Capítulo 3, quando também serão comparados os mecanismos abordados no Capítulo 1. A implementação e metodologia de um protótipo experimental de mecanismo de combate a DDoS e seus resultados serão explanados no Capítulo 4.

Esta dissertação é finalizada com o Capítulo 5, no qual são apresentadas as conclusões, as limitações do experimento realizado e as proposições para trabalhos futuros.

2 ATAQUES DE NEGAÇÃO DE SERVIÇO

Há pouco tempo tinha-se a impressão de que os impactos da negação de serviço, para certos serviços como o HTTP, eram limitados (Zwicky, Cooper & Chapman, 2000, p. 385), mas com o aumento do uso da Internet, o seu fortalecimento como meio para comércio e transações financeiras, além da concepção de novos padrões e serviços como os *WebServices* pode-se avaliar que os impactos, principalmente financeiros, podem ser catastróficos.

Neste capítulo serão apresentados os principais ataques de negação de serviço, incluindo os ataques distribuídos, objetivando fornecer ao leitor uma visão abrangente do problema, suas causas e conseqüências relacionadas.

Alguns casos notórios de negação de serviço serão relatados para evidenciar a atualidade do problema e os grandes impactos econômicos que estas ocorrências podem provocar.

2.1 ATAQUES DISTRIBUÍDOS DE NEGAÇÃO DE SERVIÇO

A negação de serviço provocada por agentes distribuídos utiliza geralmente uma ou várias das técnicas que serão mencionadas neste capítulo, mas a característica principal é a utilização de vários agentes no ataque de maneira a dificultar tanto a sua detecção quanto sua investigação.

Por implementar essa distribuição na perpetração de ataques, o DDoS também pode atingir maiores proporções, conseqüentemente com maiores chances de atingir seus objetivos. Estes ataques podem ser iniciados por qualquer usuário de uma rede de computadores com pouco conhecimento e poucas ferramentas. Pode-se encontrar atualmente de maneira bastante fácil na Internet ferramentas e condições para iniciar ataques de DDoS.

A arquitetura típica dos ataques distribuídos de negação de serviço, uma vez feita a escolha da vitima, envolve pelo menos três fases:

1. Recrutamento – o atacante trabalha na identificação de máquinas vulneráveis que servirão de máquinas clientes ou mestres e máquinas escravas ou agentes no processo de inicialização de um ataque distribuído. Geralmente é escolhido um número reduzido de máquinas que atuarão como clientes ou mestres e o critério de escolha inclui aquelas máquinas que dispõe de largura de banda considerável e estejam constantemente disponíveis, como é o caso de servidores de aplicações legadas, os quais geralmente possuem sistemas

operacionais obsoletos e com muitas vulnerabilidades conhecidas. As máquinas escravas receberão comandos, a partir das máquinas clientes, para disparar pacotes contra a vítima.

2. Preparação do ataque – nesta fase o atacante executa a instalação de ferramentas para coordenação dos ataques nas máquinas cliente e escravas;
3. Inicialização do ataque – um comando enviado pelo atacante à(s) máquina(s) cliente(s) dará início imediatamente ou programará um ataque DDoS a ser perpetrado pelas máquinas escravas contra a(s) vítima(s). O tempo de duração e periodicidade deste ataque também pode ser programado.

Uma classificação proposta pela Mirkovic (2003) para os mecanismos de ataques DDoS é apresentada na Figura 11, a qual será utilizada para categorizar alguns mecanismos que serão discutidos em seguida.

Conforme Mirkovic (2003) os critérios utilizados para conceber esta classificação levaram em consideração as formas de preparar e executar ataques, as características de ataque em si e os efeitos provocados à vítima.

2.2 FOCOS DE EXPLORAÇÃO

A negação de serviço como um problema de segurança da informação também está relacionada com as causas comuns deste tipo de problema, quais sejam falhas de especificação, implementação ou configuração. E estas falhas são geralmente consideradas como focos de exploração pelos atacantes.

Nakamura (2000) considera que os desenvolvedores seriam os maiores responsáveis por ataques de negação de serviço, pois implementações elaboradas de maneira incorreta podem criar condições para a exploração de serviços ou aplicações.

Muitas vezes falhas em implementações de serviços e aplicações são exploradas para criar e disseminar códigos maliciosos, os quais por sua vez podem conter também algumas falhas que provocam efeitos indesejáveis. Um exemplo deste tipo de ocorrência foi o *worm* (verme) *Slammer* ou W32.SQLExp.Worm (Knowles, 2003).

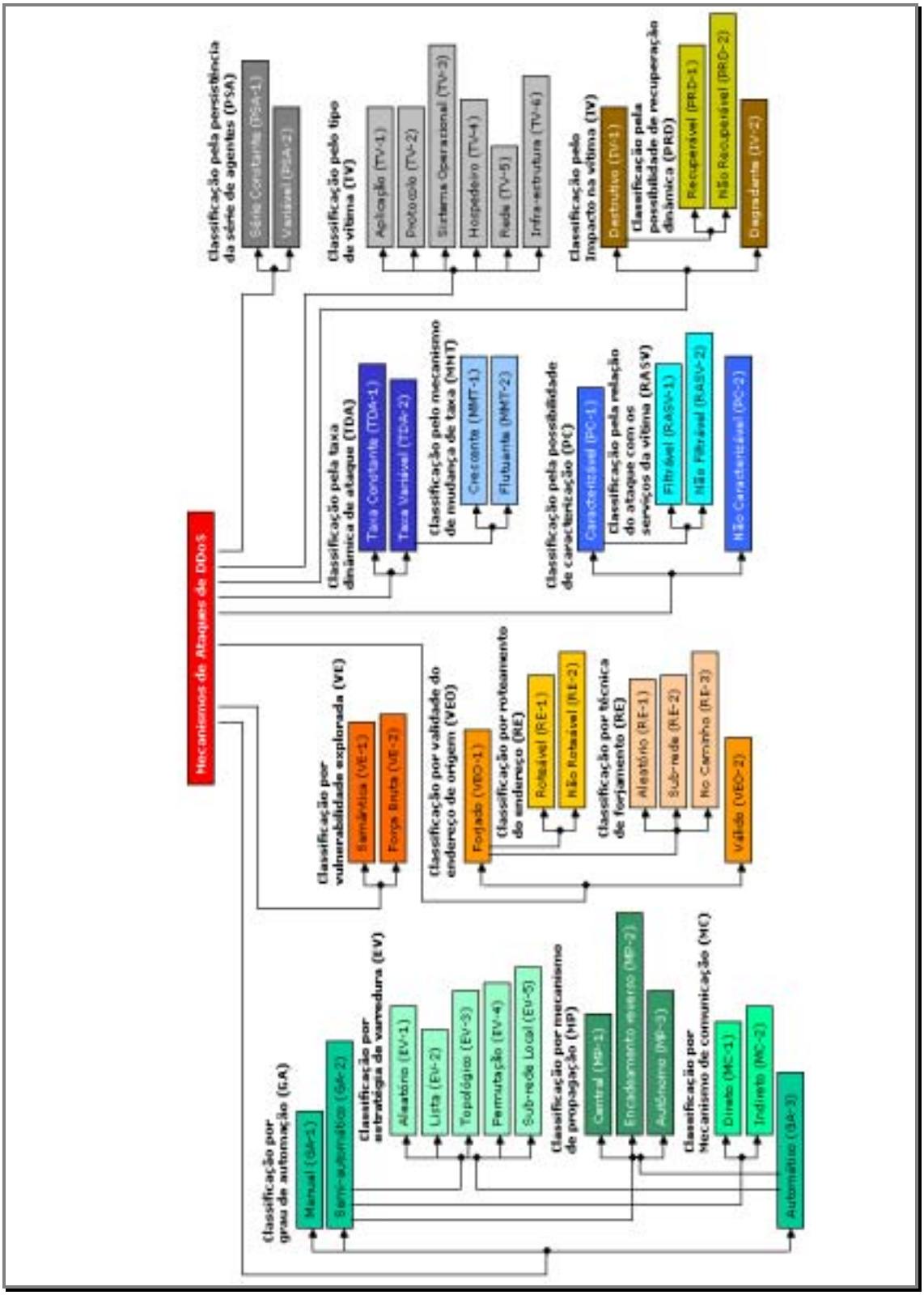

Figura 11 - Taxonomia de mecanismos de ataques de DDoS (Mirkovic, 2003)

A especificação TCP/IP é considerada deficiente por facilitar a perpetração de ataques como será apresentado a seguir. Mas, a incapacidade de autenticação, a ausência de garantia de integridade e privacidade dos dados são considerados os problemas mais sérios desta especificação. Neste sentido, especificações de protocolos, serviços ou aplicações são focos de exploração bastante utilizados.

A negação de serviço pode se propagar a recursos que não estejam diretamente disponíveis na Internet, mas que compõem determinadas aplicações ou serviços, como os sistemas de autenticação/autorização, serviços de diretório, gerenciadores de banco de dados, entre outros. Isso é possível em função de falhas de implementação ou configuração nas aplicações que utilizam ou que disponibilizam tais recursos.

Para Corsaire e Vries (2004), “Atacantes, ainda, não exploraram toda a gama de vulnerabilidades presentes em muitos dos serviços *online* – particularmente ataques direcionados à camada de aplicação e processamento de dados.”.

No referido trabalho, algumas classes de ataques direcionados às aplicações e já conhecidas são apresentadas, mas cabe ser enfatizado que algumas delas ainda são pouco utilizadas por dependerem de conhecimentos específicos sobre o potencial alvo de ataque. Por outro lado, a perpetração de um ataque focado na aplicação pode exigir muito menos recursos do que os ataques direcionados a roteadores, *firewalls* ou sistemas operacionais. Mas, se este potencial alvo é muito conhecido e visitado, ou lida com volumosos recursos financeiros, o interesse de atacantes pode aumentar.

2.3 TÉCNICAS DE ATAQUES

Com o objetivo de expor a diversidade de causas para o problema de negação de serviço, além de aprimorar o seu entendimento serão apresentadas adiante algumas das técnicas mais comuns de ataques de negação de serviço. Para algumas destas serão elaboradas algumas recomendações para resolver ou contornar tais problemas. No Capítulo 3 outras recomendações mais simples e mais específicas para enfrentar tais problemas de maneira mais abrangente serão apresentadas.

2.3.1 INUNDAÇÃO SYN

Conexões TCP sempre são iniciadas por um protocolo chamado *three-way-handshake* (cumprimento de três vias), como representado na Figura 12.

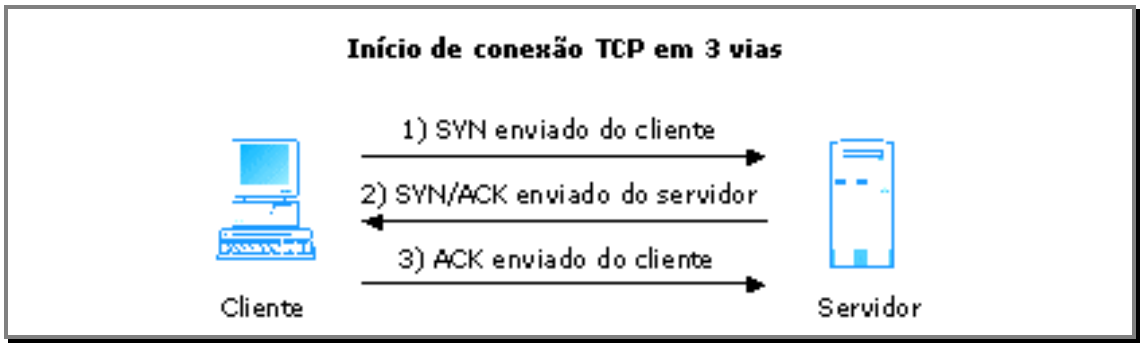

Figura 12 - Conexão SYN

Este protocolo usado para sincronização entre cliente e servidor é iniciado por um processo cliente que envia uma requisição de conexão TCP SYN (*synchronize* ou sincronize) a um processo servidor e aguarda uma resposta deste servidor. Esta resposta é feita quando o processo servidor envia ao cliente um TCP SYN/ACK (*synchronize*/*acknowledgment* ou sincronize/reconhecimento). O terceiro passo deste processo é o estabelecimento da conexão TCP propriamente dita com o cliente enviando um ACK.

Uma possível exploração deste protocolo comum a todas as implementações de TCP é a requisição de diversas conexões com SYN, geralmente com endereços de origem forjados. Isso é feito até que a pilha de requisições mantida pelo servidor seja exaurida ou até que o link até este servidor seja saturado. Essa exploração é conhecida com SYN *flooding* ou inundação SYN (CERT, 1996b).

Em um servidor é possível tratar tais tipos de explorações com um recurso conhecido como SYN *cookies* (Bernstein), o qual lança mão de recursos de criptografia para tratar requisições TCP SYN, eliminando a necessidade de manutenção de uma tabela em memória para tais requisições. Mas, isso não impede o consumo de banda até o servidor.

Também é possível implementar o recurso TCP *intercept* (interceptar) em roteadores ou *firewalls* (Cisco Systems). O dispositivo que implementa esse recurso intercepta conexões TCP direcionadas a servidores e as valida como se fosse tais servidores antes de possibilitar o estabelecimento desta conexão diretamente com estes. Abaixo um exemplo da utilização do recurso, com a

definição de uma *access list* (lista de acesso) 101 que força o software a interceptar os pacotes direcionados à sub-rede 192.168.1.0/24.

```
ip tcp intercept list 101
!
access-list 101 permit tcp any 192.168.1.0 0.0.0.255
```

Em *firewalls* construídos com iptables (Russel & Welte, 2002) é possível implementar limitação de tráfego no encaminhamento de pacotes aos servidores, o que evita que tal tipo de ataque provoque negação de serviço nestes.

Outra possibilidade seria a implementação do FDS (*Flooding Detection System* - Sistema de Detecção de Inundação) proposto por Haining, Zhang e Shin (2002), mas ainda depende de implementações comerciais e estudos mais aprofundados para investigar, principalmente, sua aplicação em sítios de alta disponibilidade que dispõem de mais de uma conexão com a Internet.

O synful (Synful.c) é um exemplo de ferramenta bastante simples que pode ser utilizada para produzir um ataque de inundação SYN com endereços e portas de origem forjados gerados aleatoriamente. Para utilizá-la basta fornecer o endereço da vítima como parâmetro e, por padrão do código fonte, 1000 pacotes serão gerados contra o serviço HTTP (porta 80) da vítima. Os pacotes gerados por esta ferramenta caracterizam-se por ter TTL (*Time to Live*) igual a 255, seqüencial igual a zero, tamanho de janela igual a 512 e utilização da *flag* CWR (*Congestion Window Reduced* – Janela de Congestionamento Reduzida). Em conexões reais o seqüencial possuir valor diferente de zero e o *flag* CWR é muito pouco utilizado.

Na “Classificação pelo tipo de vítima” proposta por Mirkovic (2003) esta técnica de ataque refere-se ao “Protocolo” (TV-2).

2.3.2 FRAGMENTAÇÃO DE PACOTES IP

Relatado pela primeira vez em 1996 e chamado inicialmente de *Ping o' Death* (Kenney, 1997) ou ping da morte, não atingia apenas pacotes ICMP ECHO, mas também TCP e UDP. Grandes pacotes fragmentados provocavam negação de serviço no sistema destino por estouro de memória temporária durante a reconstrução dos fragmentos destes pacotes. Um simples comando ping poderia provocar negação de serviço em sistemas comumente usados na época, como o Windows 95 ou NT:

```
ping -l 65510 <sistema_destino>
```

As versões mais atuais dos sistemas operacionais já não são suscetíveis ao *Ping o' Death*, mas ataques que exploram a fragmentação de pacotes IP ainda são possíveis.

O teardrop é um exemplo de ferramenta que explora a fragmentação de pacotes sobrepondo fragmentos, como mostrado nas Figuras 13 e 14, para construir pacotes com conteúdo que objetivam produzir um ataque ou uma invasão. (CERT, 1997).

Nos sistemas operacionais Windows 95 / 98/98SE/2000Beta/NT 4.0/NT 4.0 TSE (Microsoft Corporation, 2004) uma vulnerabilidade associada a pacotes IGMP podia provocar acesso indevido da pilha TCP/IP ao segmento de memória do computador e causar degradação de performance ou indisponibilidade do sistema operacional. Isso era provocado por pacotes IGMP fragmentados que eram inadequadamente processados por estes sistemas e era conhecida como inundação IGMP (IGMP *flood).*

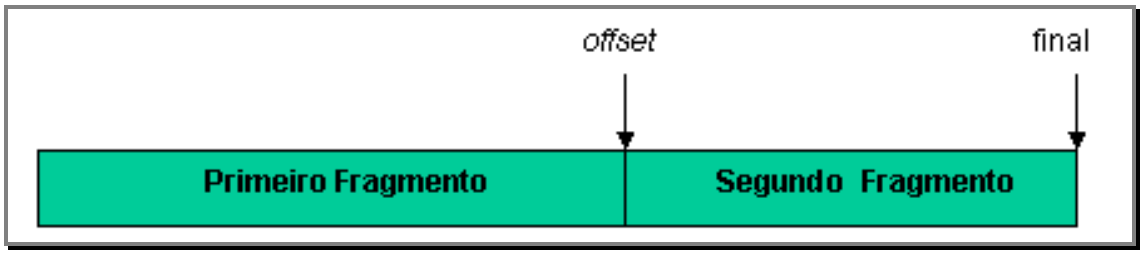

Figura 13 - Fragmentação normal de pacotes IP

Figura 14 - Fragmentação em um ataque teardrop (Hoggan, 2000)

Esta técnica também encaixa-se na “Classificação pelo tipo de vítima” proposta por Mirkovic (2003) na categoria “Protocolo” (TV-2) ou em alguns casos “Sistema Operacional” (TV-3).

2.3.3 PACOTES DIRECIONADOS AOS ENDEREÇOS DE DIFUSÃO

Endereços de difusão (*broadcast)* das sub-redes podem ser utilizados para a propagação de pacotes ICMP ECHO (no caso de um ataque conhecido como Smurf) ou UDP (no caso de um ataque conhecido como Fraggle), com endereço de origem falsificado como se a vítima houvesse os enviado. Essa técnica provoca o envio de muitos pacotes direcionados à vítima, em resposta aos pacotes falsificados. Seu esquema típico está representado na Figura 15.

Para evitar que roteadores tornem-se amplificadores destes tipos de ataques, restringir acesso a endereços de *broadcast* é considerada uma boa prática, como recomendado na RFC-2644 (*Request for Comments* – Pedido de Comentários) (Senie, 1999).

Um sítio que não implemente esta boa prática em seus roteadores pode ser explorado em ataques de DoS ou DDoS para que atue como refletor.

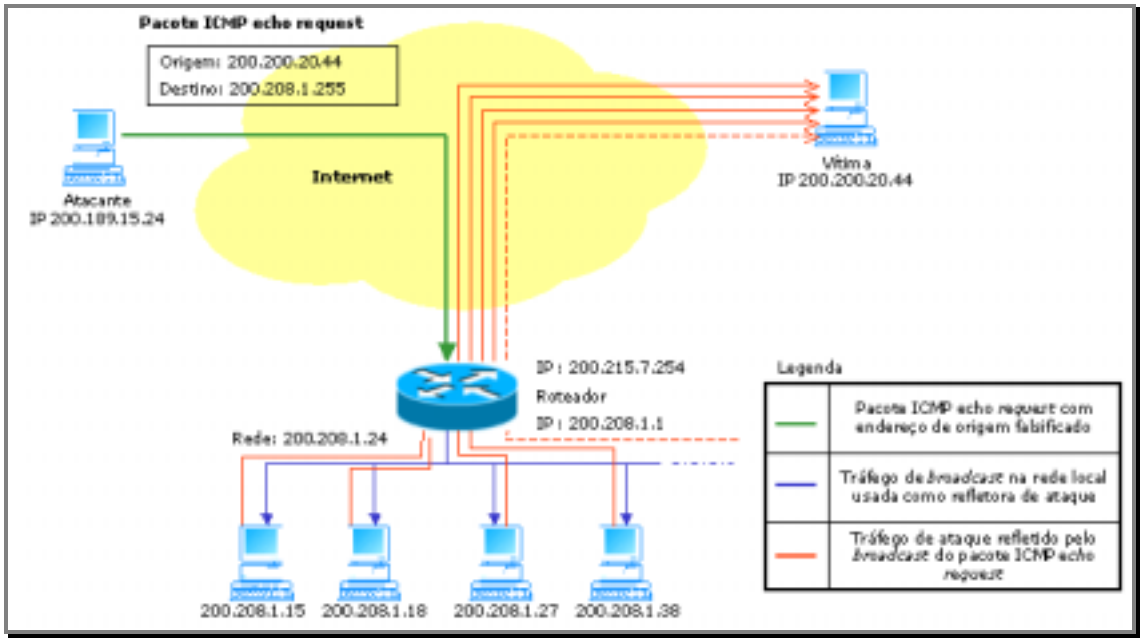

Figura 15 - Esquema da Técnica de Ataque que Explora Endereços de *Broadcast* com Pacotes ICMP

Pode-se classificar esta técnica como proposto por Mirkovic (2003) na “Classificação pelo tipo de vítima” na categoria “Aplicação” (TV-1) e ainda, na “Classificação por validade do endereço de origem” em “Forjado” (VEO-1) e “Roteável” (RE-1).

2.3.4 ENDEREÇOS DE ORIGEM E DESTINO IDÊNTICOS

Nesta técnica pacotes TCP SYN com endereçamento IP idêntico de origem e destino são utilizados, onde o endereço especificado é o da vítima. O número de porta TCP de origem e destino também são idênticos e geralmente referem-se a um serviço conhecido e que esteja ativo no sistema da vítima. Land foi o nome dado a este tipo de ataque quando apenas uma porta é utilizada (Hoggan, 2000). O ataque LaTierra funciona de maneira semelhante ao Land, mas envia o pacote TCP à mais de uma porta e mais de uma vez (Huovinen & Hursti, 1998).

Esta técnica de ataque divide-se em quatro fases como mostrado na Figura 16:

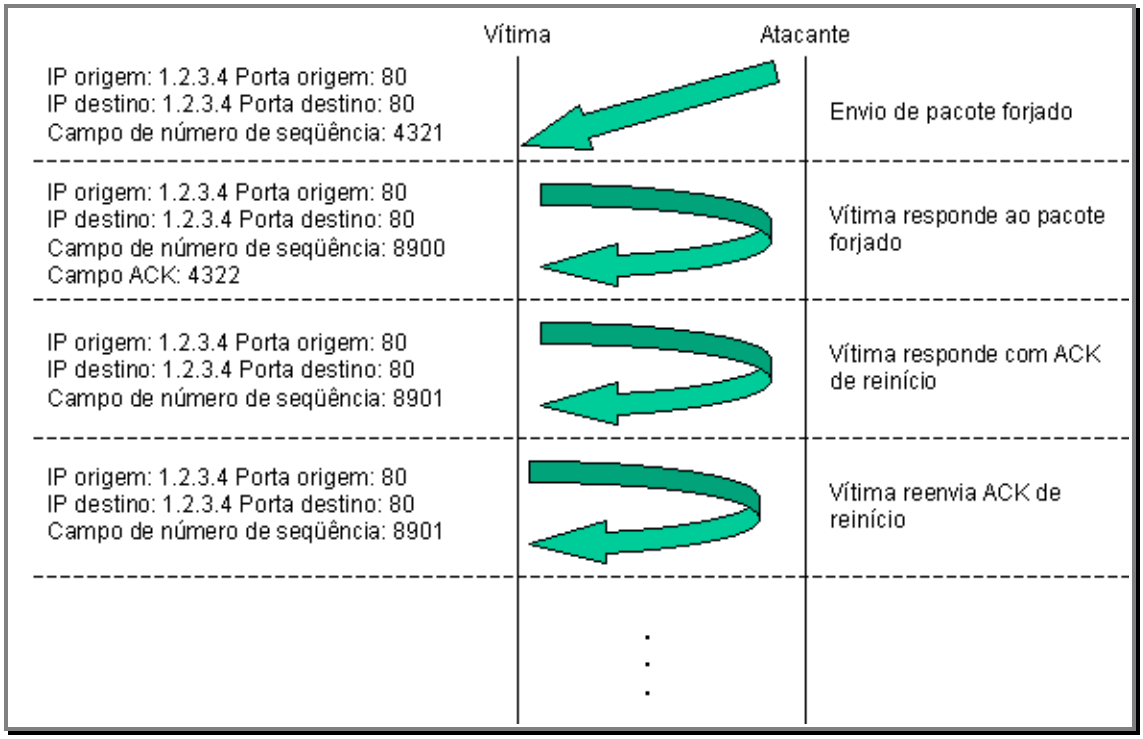

Figura 16 - **Técnica de Ataque com Endereços Origem e Destino Idênticos**

1. O atacante envia uma requisição de conexão forjada à vitima, onde o número de seqüência contém qualquer valor e o campo ACK contém zero.
2. A vítima continua com o cumprimento de três vias para início de conexão (*tree-way handshake*) enviando à origem um pacote com o seu número de seqüência inicial e incrementando o número de seqüência recebido colocando-o no campo ACK.
3. Como a origem é a própria vítima, esta recebe o pacote enviado na fase 2, o qual não contém o número de seqüência esperado e então a vítima envia um

pacote ACK à origem reiniciando o número de seqüência e de reconhecimento (*acknowledgement*) esperados.

4. A vítima recebe o pacote ACK que acabou de enviar e verifica que o número de seqüência não é o esperado e então o processo se repete nesta fase indefinidamente.

Esta técnica pode ser classificada como proposto por Mirkovic (2003) na “Classificação pelo tipo de vítima” na categoria “Protocolo” (TV-2) e ainda, na “Classificação pela possibilidade de caracterização” em “Caracterizável” (PC-1) e “Filtrável” (RASV-1).

2.3.5 INUNDAÇÃO HTTP

Os ataques HTTP *flood*, como o próprio nome já evidencia, são destinados ao serviço HTTP da vítima.

Ataques deste tipo não são possíveis com a utilização de endereços forjados, pois é necessário o estabelecimento de conexões TCP. Portanto, para ser possível provocar condição de negação de serviço à vitima o cliente utilizado para o ataque deve possuir significativa capacidade de recursos para conseguir saturar os recursos da vítima. Outra possibilidade é a utilização de diversos agentes em um cenário de DDoS.

Na classificação proposta por Mirkovic (2003) pode-se classificar esta técnica na “Classificação pelo tipo de vítima” na categoria “Aplicação” (TV-1), na “Classificação por vulnerabilidade explorada” em “Força Bruta” (VE-2) e ainda, na “Classificação por validade do endereço de origem” em “Válido” (VEO-2).

2.3.6 INUNDAÇÃO UDP

A Figura 17 mostra o esquema do ataque de laço infinito baseado nos serviços *echo* e *chargen* que utilizam pacotes UDP (CERT, 1996a).

Um laço infinito (*looping*) entre “Alvo 1” e “Alvo 2” provoca uma inundação UDP (UDP *flooding*) que consome todos os recursos dos sistemas provocando negação de serviço em ambos.

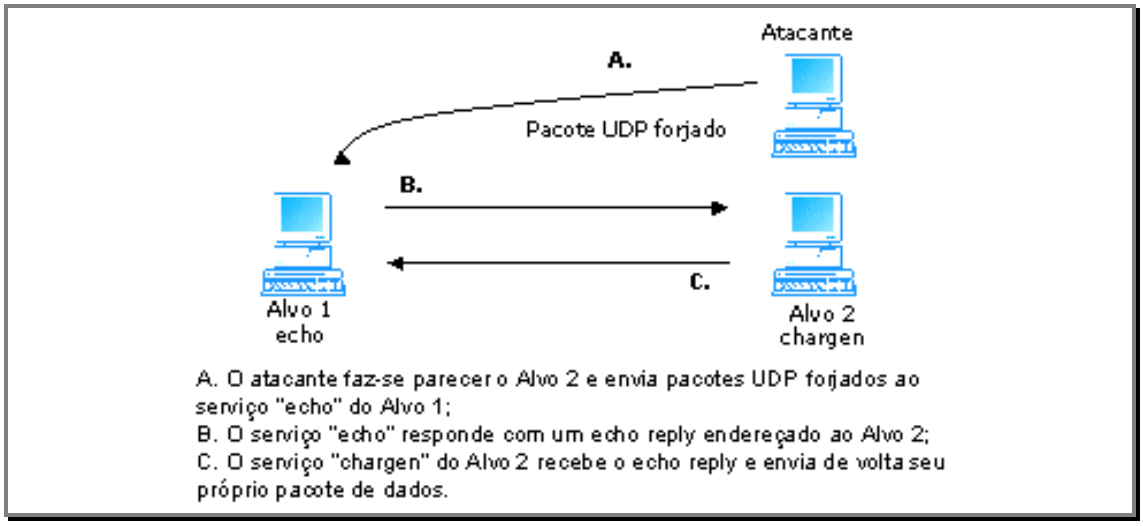

Figura 17 - Ataque de laço infinito em portas UDP (Simcock, 2002)

Se pacotes UDP forem direcionados ao endereço de difusão (*broadcast*) de uma sub-rede e esta estiver conectada de maneira redundante, isto é, existirem dois caminhos físicos entre dois pontos desta sub-rede, uma inundação UDP, mais especificamente chamada de *broadcast storm* (tempestade de difusão), poderá provocar a degradação de performance ou mesmo negação de serviço nesta sub-rede.

Um exemplo de ferramenta que utiliza esta técnica para construir ataques distribuídos de negação de serviço é o Trinoo, a qual para instalar-se procura explorar vulnerabilidades nos serviços RPC (*Remote Procedure Call* – Chamada Remota de Procedimento) "statd", "cmsd" e "ttdbserverd", instalando em seguida instâncias de programas servidores para atuarem como mestres ou escravos na tarefa de perpetração do ataque. Uma rede criada com trinoo, totalizando 227 sistemas, dos quais 114 eram sítios da Internet2, foi utilizada em 17 de agosto de 1999 para saturar um único sistema na University of Minnessota, inundando a rede e tornando-a indisponível por mais de dois dias (Dittrich, 1999a).

Esta técnica pode ser classificada como proposto por Mirkovic (2003) na “Classificação pelo tipo de vítima” em “Aplicação” (TV-1) e no caso da utilização de ferramenta como o trinoo na “Classificação por grau de automação” em “Semi-automático” (GA-2) e na “Classificação por Mecanismo de comunicação” como “Direto” (MC-1).

2.3.7 RÁPIDO INÍCIO E ENCERRAMENTO DE CONEXÕES TCP

Muitas conexões TCP podem ser iniciadas e encerradas tão rapidamente de maneira que os recursos da vítima sejam consumidos rapidamente e novas conexões não mais sejam aceitas pelo servidor. Esta técnica é conhecida como *open/close* (abrir/fechar).

Serviços disponibilizados por inetd[3] são vítimas preferenciais deste tipo de ataque, pois o número de conexões possíveis dentro de um pequeno período de tempo é estaticamente definido no código deste serviço. Alguns dos serviços comumente disponibilizados por inetd são: FTP (*File Transfer Protocol*), telnet, TFTP (*Trivial File Transfer Protocol*), echo, finger, time, daytime, talk, entre outros.

Também é possível classificar esta técnica como proposto por Mirkovic (2003) na "Classificação pelo tipo de vítima" na categoria "Aplicação" (TV-1) e na "Classificação por vulnerabilidade explorada" como "Força Bruta" (VE-2).

2.3.8 EXPLORAÇÃO POR MENSAGENS ICMP

Esta técnica utiliza determinados tipos de mensagens ICMP para provocar negação de serviço.

Quando mensagens ICMP *unreachable* (inalcançável) são utilizadas, o ataque provoca o encerramento de conexões TCP válidas, pois o atacante interfere nas conexões enviando tais mensagens forjadas com o endereço do real cliente do serviço como mostrado na Figura 18.

Conexões TCP são então reiniciadas e mais mensagens ICMP *unreachable* são enviadas pelo atacante com endereço forjado, até o momento que ocorre uma condição de negação de serviço.

Também é considerada uma estratégia para negação de serviço em roteadores. Se um atacante envia muitas requisições de conexão, provavelmente com endereço de origem forjado, a um serviço não habilitado em um roteador, por padrão este dispositivo responde com uma mensagem ICMP *unreachable* para cada requisição. Dependendo do número de requisições e dos recursos disponíveis a performance pode ser degradada neste dispositivo ou poderá ocorrer uma condição de negação de serviço.

---

[3] O Internet *Daemon* ou muitas vezes chamado de *SuperServer* (super servidor) é um processo único executado em máquinas Unix, o qual cria múltiplos *sockets* para vários serviços e aguarda por requisições para estes serviços. Quando uma nova requisição é recebida o processo inetd cria um processo "filho" para tratar dela.

Um atacante ainda pode enviar várias mensagens ICMP *redirect* (redirecionar) para um roteador e com isso saturar toda a memória deste roteador de tal modo que o mesmo não seja capaz de adicionar novas rotas à sua tabela ou mesmo responder às requisições de seus serviços.

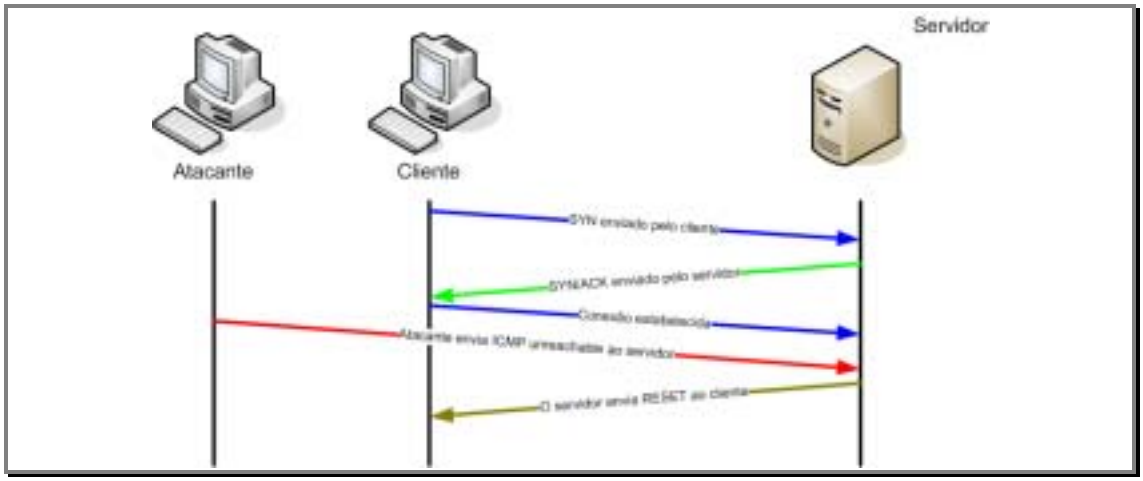

Figura 18 - Ataque com ICMP *unreachable*

Outra exploração possível e que provoca negação de serviço é a utilização de mensagens ICMP *redirect* para alimentar tabelas de roteamento em sistemas específicos (Volobuev, 1997) com endereços inalcançáveis ou com endereços de roteadores comprometidos.

Esta técnica pode ser classificada como proposto por Mirkovic (2003) na “Classificação pelo tipo de vítima” como “Protocolo” (TV-2).

Um exemplo de ferramenta que utiliza mensagens ICMP na construção de ataques é o TFN (Tribe Flood Network). Analisado por Dittrich (1999b) e CERT (1999a) possui componentes cliente e servidor. Também é capaz de provocar inundações ICMP, SYN e UDP, e ainda, ataques tipo Smurf, além disso possibilita acesso por meio de porta TCP para execução de comandos com usuário privilegiado em sistemas Unix. Explora as mesmas vulnerabilidades que o trinoo para instalação.

2.3.9 REDIRECIONAMENTO ARP

Um atacante que faça parte da mesma rede da vítima pode tornar esta vítima inalcançável fazendo uso de mensagens ARP (*Address Resolution Protocol* – Protocolo de Resolução de Endereço) *redirect* (redirecionar) que modifiquem no roteador o mapeamento de endereços MAC (*Media Access Control* – Controle de Acesso ao Meio) para endereços IP (Volobuev, 1997). Desta forma, conexões

direcionadas à vítima podem ser redirecionadas para o atacante (se este estiver no mesmo segmento de rede local) ou a um “buraco negro” (*black hole*), isto é, uma rede que não existe ou, um endereço inalcançável.

É possível classificar esta técnica como proposto por Mirkovic (2003) na “Classificação pelo tipo de vítima” como “Protocolo” (TV-2) e ainda na “Classificação por vulnerabilidade explorada” em “Semântica” (VE-1).

2.3.10 ROTEAMENTO

A negação de serviço baseada em roteamento envolve a manipulação de tabelas de roteamento pelo atacante. Entradas nestas tabelas são alteradas de maneira que os fluxos trafeguem através da rede do atacante ou diretamente para um “buraco negro”.

Isto acontece por meio da exploração da inexistência ou franquezas dos mecanismos de autenticação de protocolos de roteamento como o RIP v1 (*Routing Information Protocol* – Protocolo de Informação de Roteamento) e BGP v4 (*Border Gateway Protocol* – Protocolo de Passagem de Borda). (McClure, Scambray & Kurtz, 1999, p. 342-343.)

Na classificação proposta por Mirkovic (2003) esta técnica refere-se à “Classificação pelo tipo de vítima” em “Protocolo” (TV-2) e ainda na “Classificação por vulnerabilidade explorada” em “Semântica” (VE-1).

2.3.11 DNS

Os ataques de DoS em servidores de nomes de domínios (DNS - *Domain Name Server*) buscam ludibriar a vítima de maneira que esta armazene em sua memória *cache* mapeamentos inválidos de nomes para endereços. Desta forma, o atacante pode redirecionar clientes que requisitem tais informações ao servidor ludibriado a um sítio determinado pelo atacante ou mesmo a um “buraco negro”.

Na proposta de classificação de Mirkovic (2003) esta técnica refere-se à “Classificação pelo tipo de vítima” em “Protocolo” (TV-2) e ainda na “Classificação por vulnerabilidade explorada” em “Força Bruta” (VE-2).

2.3.12 BUFFER OVERFLOW

Uma das falhas de implementação mais utilizadas para perpetrar ataques de negação de serviço se refere à memória temporária para armazenamento de dados (*buffer*), a qual, se não controlada adequadamente pelo desenvolvedor, possibilita a um atacante inserir código malicioso no espaço de endereçamento do programa explorado.

A Figura 19 mostra as localizações de memória onde pode ocorrer *buffer overflow.*

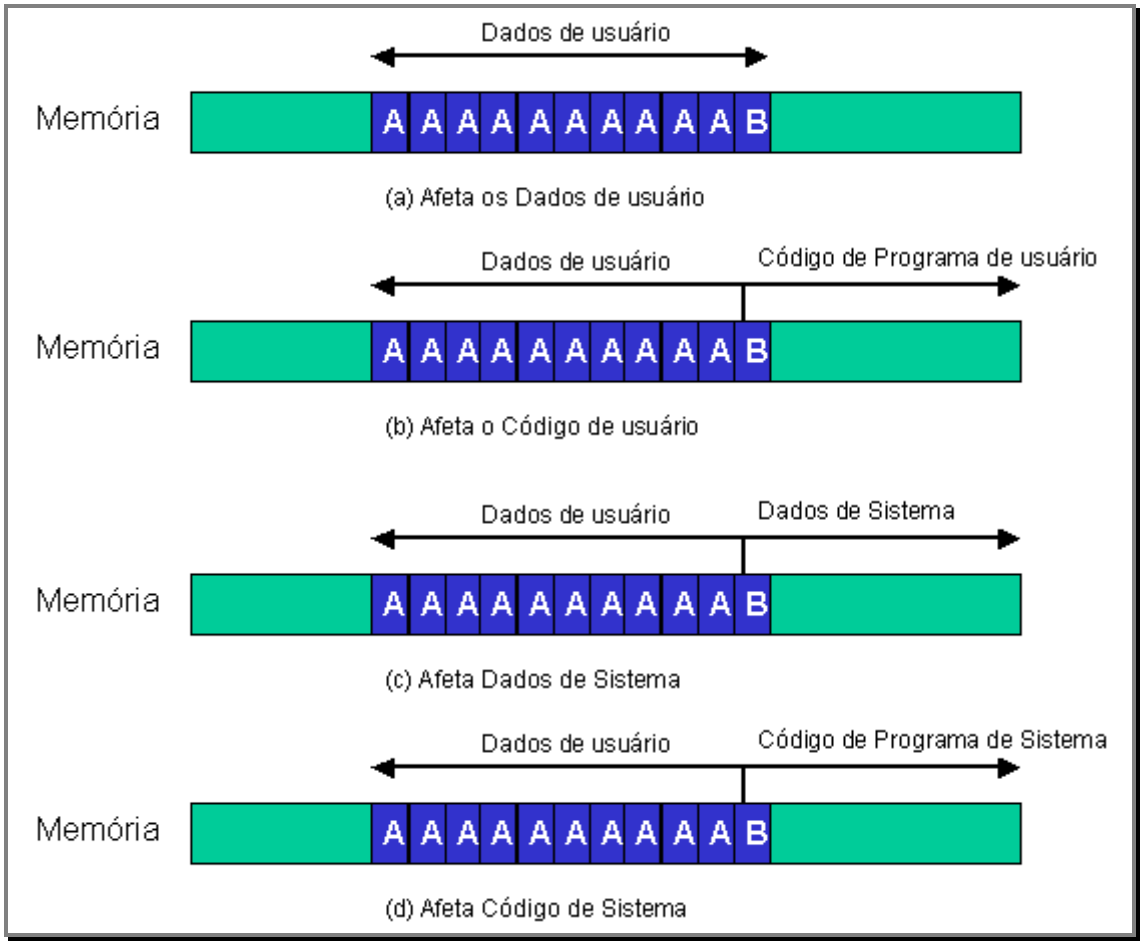

Figura 19 - Locais onde *Buffer Overflow* pode ocorrer (Pfleeger & Pfleeger, 2003)

Mais da metade dos boletins relatando vulnerabilidades referem-se a *buffer* overflow. (Nakamura, 2000, p. 73).

Conforme Pfleeger & Pfleeger, 2003:

> *A buffer overflow is the computing equivalent of trying to pour two liters of water into a one-liter pitcher: Some water is going to spill out and make a mess. And*

> *in computing, what a mess these errors have made!* (*Um estouro de memória temporária para armazenamento de dados é o equivalente em computação a tentar colocar dois litros de água em um jarro de um litro: Alguma água irá derramar e fazer uma bagunça. E em computação, que bagunça estes erros podem causar!)*

As possíveis explorações com este tipo de técnica são diversas e dependentes da implementação ou configuração de protocolo, serviço ou aplicação vulnerável. Desta forma, cada tipo de exploração pode ser classificada diferentemente dentro da proposição de Mirkovic (2003), mas na maioria das vezes receberão uma "Classificação por vulnerabilidade explorada" como "Força Bruta" (VE-2).

A seguir é mostrado um exemplo que permite a exploração por *buffer overflow* (estouro da memória temporária para armazenamento de dados) (Wheeler, 2003).

```
/* 1) signedness - DO NOT DO THIS. */
char *buf;
int i, len;

read(fd, &len, sizeof(len));

/* OOPS! We forgot to check for < 0 */
if (len > 8000) { error("too large length"); return; }

buf = malloc(len);
read(fd, buf, len); /* len casted to unsigned and overflows */
```

2.3.13 REINICIALIZAÇÃO DE CONEXÕES TCP

Uma preocupante possibilidade de exploração vinculada à forma como certos fabricantes, em conformidade com a RFC 793, implementam a aceitação de requisições de finalização de conexão foi avaliada por Watson (2004) e posteriormente alertada por NISCC (2004).

O comportamento normal de requisições de finalização (RST) em conexões TCP é mostrado nas Figuras 20 e 21.

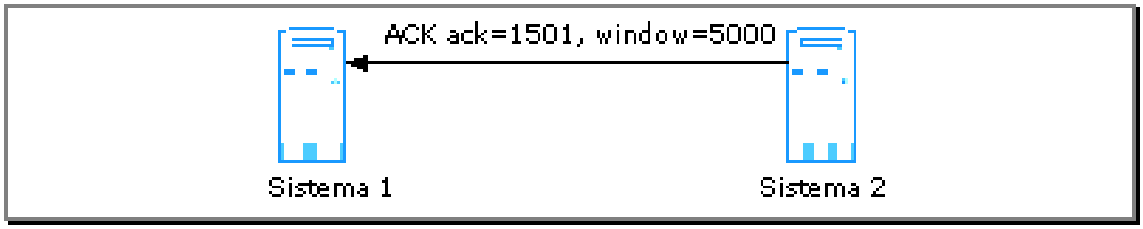

Figura 20 - Comportamento normal de uma conexão TCP

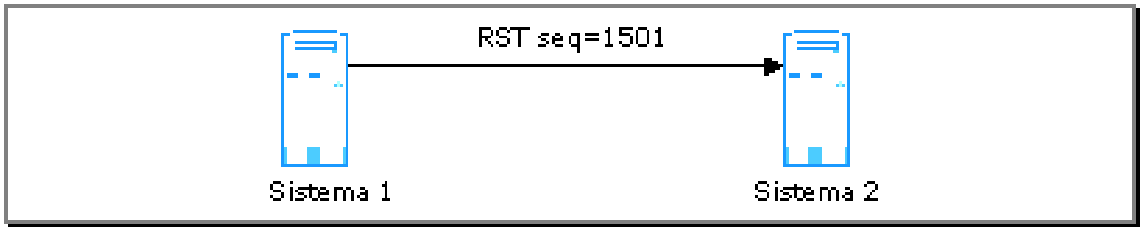

Figura 21 - Finalização de sessão com próximo seqüencial esperado

Muitas implementações de TCP/IP esperam que em requisições RST o seqüencial informado esteja dentro da janela (window) especificada. Portanto é possível finalizar uma conexão TCP de maneira semelhante à representada nas Figuras 22 e 23.

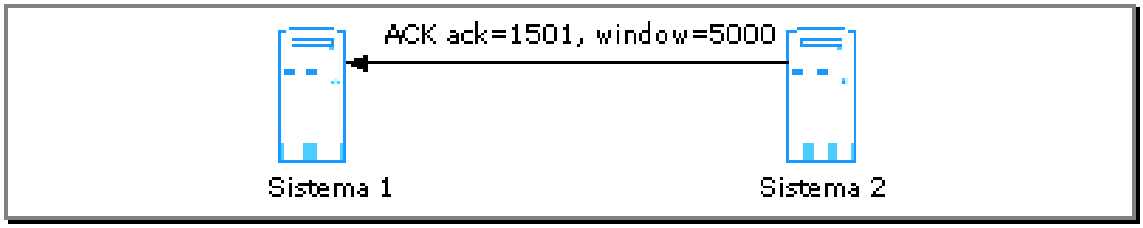

Figura 22 - Sistema 2 informa próximo seqüencial esperado na conexão TCP

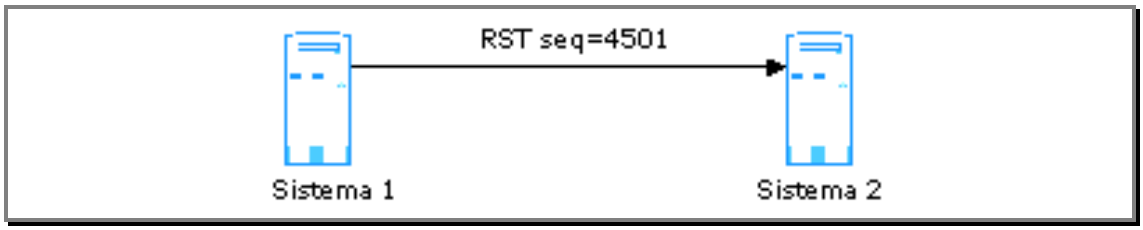

Figura 23 - Finalização de sessão com seqüencial dentro da janela especificada

Mas, além de um número de seqüência dentro da janela especificada, para que seja possível um atacante finalizar conexões entre dois sistemas são necessárias mais quatro informações referentes a estas conexões, quais sejam:

- o endereço de origem;
- o endereço de destino;
- a porta de origem, e;
- a porta de destino.

Neste caso a informação mais difícil de se obter provavelmente seja a porta de origem, uma vez que esta é atribuída dinamicamente.

A previsibilidade de algumas implementações facilita o trabalho do atacante, pois alguns sistemas utilizam incrementos constantes na atribuição de portas para iniciar uma conexão com pacotes TCP.

Dentre os protocolos alvos de exploração deste tipo de ataque provavelmente o que sofreria maior impacto seja o BGP. Uma simples seqüência de quebras de conexões BGP dentro de um determinado período de tempo pode tornar indisponíveis porções significativas de sistemas na Internet.

Esta técnica pode ser classificada como proposto por Mirkovic (2003) na "Classificação pelo tipo de vítima" na categoria "Protocolo" (TV-2) e na "Classificação por vulnerabilidade explorada" em "Força Bruta" (VE-2).

2.3.14 ATAQUES UTILIZANDO INFRAESTRUTURA IRC

As redes IRC (*Internet Relay Chat* – Rede de Bate-papo na Internet) (Oikarinen & Reed, 1993) são potenciais infra-estruturas para a perpetração de ataques de DDoS, inclusive contra elas próprias. Fazendo uso de *botnets*[4] criadas por meio de cavalos-de-tróia (*trojan horses)* ou não, comandos DCC (*Direct Client to Client* ou Direto Cliente a Cliente) ou CTCP (*Client-to-Client Protocol* ou Protocolo Cliente-a-Cliente) (Zeuge, Rollo & Mesander, 1994), além de *scripts* complexos com Tcl (*Tool Command Language* ou Ferramenta de linguagem de comando) é possível perpetrar ataques de DDoS automatizados utilizando infra-estrutura IRC.

São exemplos de ferramenta automatizadas que podem ser utilizadas para ataques DDoS: knight.exe, GTbot (Global Threat Bots), X-DCC, Litmus (ZoneLabs, 2002) e o IRC/Flood (McAfee,2000).

Dentro da classificação proposta por Mirkovic (2003) ataques que utilizam esta técnica encaixam-se na "Classificação por grau de automação" na categoria "Semi-automático" (GA-2) e na "Classificação por mecanismo de comunicação" na categoria "Indireto" (MC-2).

---

[4] Botnets são redes de bots. Bot (do inglês roBot – robô) é um programa que tem a função de manter-se conectado a um canal IRC e executar comandos de seu administrador. As redes de bots são criadas com programas como o Eggdrop para diversas finalidades, como por exemplo o compartilhamento de arquivos, mas também podem ser utilizados para criar agentes de ataques DDoS.

2.3.15 ATAQUES COM O USO DE REFLETORES

O problema do uso de refletores na coordenação de ataques de negação de serviço é tratado por Paxson (2001), trabalho no qual são analisadas as possibilidades de detecção desta classe de tráfego e sugeridas possíveis soluções para o problema. São considerados refletores todos os sistemas que respondem a uma tentativa de conexão TCP, seja com SYN ACKs ou RSTs, como servidores Web, DNS, roteadores, entre outros, ou ainda respondam a mensagens ICMP.

Os ataques de negação de serviço, que utilizam este tipo de estratégia, geralmente implementam um esquema de negação de serviço baseado em inundação de pacotes.

Refletores são considerados servidores cujos endereços são utilizados por um atacante para ampliar o poder de um ataque em relação a sua distribuição, tornando-o muito mais difícil de detectar e combater.

Antecedendo ao ataque propriamente dito, o atacante (Mestre) instala em máquinas escravas ferramentas que serão utilizadas para coordenar o ataque. O recrutamento dessas máquinas escravas é feito por meio da exploração de vulnerabilidades dos sistemas implementados nessas máquinas.

Para iniciar o ataque são geradas requisições, por intermédio dos escravos, direcionadas aos refletores previamente selecionados – numa ordem de grandeza suficiente para perpetrar um ataque efetivo, por exemplo, em torno de um milhão de máquinas - forjando o endereço de origem destas requisições, como se fora a vítima. Os refletores por sua vez, respondem às requisições, mas, direcionando suas respostas ao real proprietário daquele endereço de origem, isto é, a vítima. Esta então necessita tratar estas respostas de requisições não solicitadas.

A Figura 24 mostra um esquema típico de um ataque DDoS implementado com o uso de refletores.

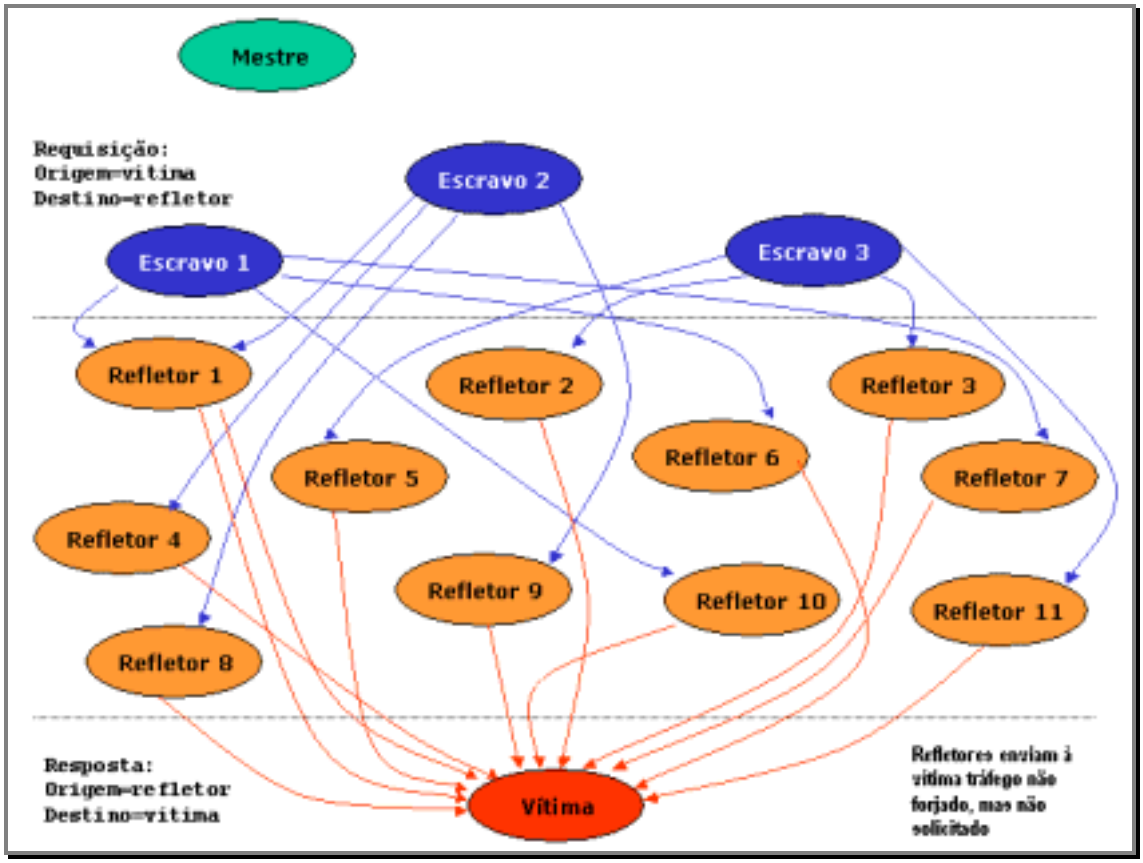

Figura 24 - Usando refletores para ampliar a distribuição de um DDoS (Paxson, 2001)

Estratégias diferentes devem ser implementadas para a defesa contra ataques que utilizem refletores, pois neste tipo de ataque, aplicar simplesmente traçado reverso para identificar a origem de um fluxo participante de um ataque não é um método muito efetivo, uma vez que os endereços serão, na maioria das vezes, forjados e não identificarão o real atacante.

Na classificação proposta por Mirkovic (2003) esta técnica pode referir-se à "Classificação por vulnerabilidade explorada" na categoria "Força Bruta" (VE-2), à "Classificação por validade do endereço de origem" na categoria "Forjado" (VEO-1).

2.4 ATAQUES NOTÓRIOS

Ataques registrados nos últimos anos demonstram que é crescente a utilização de técnicas de negação de serviço. As motivações são variadas e os métodos utilizados também.

A seguir serão relatados alguns dos ataques mais notórios.

2.4.1 SERVIDORES DE NOMES RAIZ

Uma das mais preocupantes séries de ataques de DDoS que se tem notícia ocorreu em outubro de 2002. Considerados dos maiores e mais complexos ataques já relatados contra a infra-estrutura da Internet, tinham como objetivo interromper, a partir da raiz, a resolução de nomes de domínios utilizados na Internet.

O primeiro ataque ocorreu em 22 de outubro (McGuire & Krebs, 2002a) e teve como alvo os servidores da raiz de resolução de nomes com TLDs (*Top Level Domains* – Domínios de Primeiro Nível) .com, .org e .net. Dos treze servidores espalhados pelo mundo, pelo menos nove deles foram afetados pelo ataque. Segundo os administradores do serviço de resolução de nomes isto não provocou qualquer lentidão na resolução de nomes e não foi percebido pelos usuários da Internet. Ainda conforme Naraine (2002) e Baranowski (2003) este ataque utilizou um alto volume de requisições com pacotes ICMP numa freqüência bastante grande contra os servidores de nomes raiz, gerando um *ping-flooding* (tempestade de ping).

No dia seguinte - 23 de outubro - outro ataque (McGuire & Krebs, 2002b) direcionado especificamente aos servidores de empresas como Verisign, responsável pela administração da base de nomes de domínios de TLDs .com, .org, .net, entre outras e também pela resolução de muitos desses nomes de domínios e; Afilias Ltd., responsável pela administração da base de nomes de domínios com TLD .info. Segundo as empresas as medidas defensivas utilizadas por elas foram capazes de tratar com êxito a ocorrência sem provocar qualquer tipo de problemas à resolução de nomes de domínios mantidos por elas.

2.4.2 AKAMAI

Apesar de prometer a seus clientes 100% de disponibilidade, mesmo sob ataques DDoS, a Akamai tornou-se alvo de um destes ataques. Segundo seus técnicos isto aconteceu devido à crescente utilização de *botnets* (Dalnet IRC Network, 2003) criadas por cavalos-de-tróia.

O ataque de 15 de junho de 2004 teve como alvo os servidores de DNS da companhia e como conseqüência tornou indisponíveis servidores de pelo menos 4% dos seus clientes (Netcraft, 2004b).

A motivação parece ser demonstrar que a empresa não pode cumprir o que promete em termos de disponibilidade.

2.4.3 DOUBLECLICK

Um ataque similar ao perpetrado contra a Akamai em 15 de junho de 2004 foi direcionado, no dia 27 de julho de 2004, aos servidores de DNS da Doubleclick, responsável por prover *banners* de anúncios para diversos clientes ao redor do mundo (Netcraft, 2004a). Como conseqüência alguns de seus clientes, os quais incluem estes *banners* dinamicamente junto com seu próprio conteúdo na *web*, tiveram a performance de seus sítios afetada. Em alguns casos o código que carregava os *banners* foi retirado até que o ataque cessasse.

Não são muito claras as motivações para este ataque mas os prejuízos financeiros e de imagem da empresa foram consideráveis.

2.4.4 MICROSOFT

Em 4 de agosto de 2003 alguns dos principais endereços da Microsoft na *web* foram alvos de ataque de negação de serviço que causou sua indisponibilidade por quase duas horas. (Hill, 2003)

Alvo de grandes disputas judiciais, comerciais, dentre outras é difícil supor qual a real motivação dentre tantas para este ataque. O certo é que a empresa, após o ocorrido e o surgimento de algumas outras ameaças, resolveu suportar seus sítios pela solução provida pela Akamai.

2.4.5 SCO

Na madrugada do dia 10 de dezembro de 2003 uma inundação SYN de aproximadamente 34.000 pacotes por segundo foi registrado contra o sítio do SCO Group. No dia seguinte por volta de 2:50h um ataque se iniciou contra os servidores de FTP da mesma empresa e juntamente com os servidores *web* sofreram um ataque de inundação SYN de mais de 50.000 pacotes por segundo. Esse ataque causou a indisponibilidade do sitio *web* da empresa, bem como do serviço de FTP (Caída, 2003).

No início de 2004 os ataques contra a SCO continuaram e sob condições e estratégias muito mais ousadas e complexas. Uma variante do vírus Mimail, conhecida como MyDoom infectou rapidamente muitos computadores com

sistema operacional Windows, os quais por sua vez iniciaram ataques coordenados contra o endereço principal da SCO na web, www.sco.com, o qual ficou praticamente inacessível por 16 horas entre os dias 31 de janeiro e 1º. de fevereiro de 2004 em função destes ataques. (Lemos, 2004 e Pelline, 2004)

A SCO resolveu brigar judicialmente com a IBM alegando transferência indevida de código de sua propriedade para o sistema operacional Linux sem respeitar os direitos de autoria da SCO. Desde então, a empresa começou a ser vista como inimiga do Linux pela comunidade de usuários, o que pode justificar a sua escolha como alvo desta onda de ataques.

2.4.6 ORBIT COMMUNICATION CONTRA SEUS CONCORRENTES

Este caso foi relatado pelo FBI (*Federal Bureau of Investigation* – Birô Federal de Investigação) em agosto de 2004 como o primeiro caso detectado de utilização de técnicas de DDoS com objetivos mercadológicos e também como o primeiro caso registrado de contratação de serviços de terceiros para executá-lo (Poulen, 2004).

Os objetivos eram simples, provocar negação de serviço nos sítios dos concorrentes da Orbit Communication, empresa que comercializa acesso via satélite a canais de televisão. Calcula-se que as perdas relacionadas aos ataques foram superiores a dois milhões de dólares.

As técnicas utilizadas para perpetração dos ataques envolveram a propagação de uma versão modificada do *worm* Agobot, a construção de IRC *botnets* que totalizavam cerca de 18.000 agentes, inundação com pacotes SYN e inundação com pacotes HTTP.

2.5 CONCLUSÃO

As técnicas de negação de serviço, aliadas às comprovadas fraquezas da especificação ou das implementações de TCP/IP ou, às ferramentas de automatização de ataques distribuídos, caracterizam as diversas possibilidades de perpetração de ataques com resultados de impactos significativos. Por outro lado, algumas das técnicas de ataque poderiam ter seus efeitos minimizados ou eliminados com algumas medidas até certo ponto simples a serem adotadas pelas potenciais vítimas e pelos provedores de serviços de Internet como será discutido no próximo capítulo.

Em simulação desenvolvida por Chen e*t al.* (2001) concluiu-se que os ataques distribuídos de negação de serviço são fundamentalmente mais severos quanto ao seu impacto na rede da vítima que um ataque simples de DoS. E isso não ocorre simplesmente porque um maior número de sistemas é utilizado para o ataque, mas também porquê:

- Em um ataque simples o atacante está limitado à largura de banda disponível entre este e a Internet, portanto se esta for menor do que a largura de banda disponível para a vítima, o ataque pode não surtir os efeitos esperados;
- Clientes que compartilhem conexões utilizadas em um ataque distribuído sofrerão efeitos negativos provocados pela inundação do tráfego de ataque;
- Mesmo que alguns dos atacantes ou agentes sejam identificados e bloqueados durante um ataque é muito difícil bloqueá-los completamente.

Por outro lado, se consideramos que o objetivo do atacante pode ser simplesmente negar serviço de um determinado sítio, sem necessariamente saturar os recursos de rede da vítima, existem, como visto, muitas possibilidades atualmente de utilizarem-se falhas ou vulnerabilidades em aplicações para atingir tal objetivo.

Em avaliação experimental executada na infra-estrutura de um ISP regional, Hussain, Heidemann & Papadopulus (2003) verificaram que os ataques com múltiplas origens, isto é, ataques DDoS eram significativamente mais severos em quantidade de pacotes e bytes gerados do que as demais classes de ataques avaliadas.

As conclusões preocupantes relativas aos ataques registrados nos últimos anos devem-se à diversidade de razões que motivaram tais ataques, aos registros de contratações de *hackers* para construir ataques e às estorções baseadas em ameaças de perpetração de ataques de DDoS.

Outra grande preocupação que começou a ser considerada mais fortemente após os ataques terroristas de 11 de setembro de 2001 nos Estados Unidos refere-se ao chamado terrorismo cibernético (*Cyberterrorism)*. Mesmo que os serviços críticos sejam suportados por redes dedicadas e não sobre a infra-estrutura da Internet, considera-se que ataques com impactos significativos

poderiam ser utilizados como componente para distrair as autoridades enquanto ataques terroristas (no mundo físico real) são organizados ou iniciados. Mas, há ainda a preocupação com a infra-estrutura de telecomunicações, a qual suporta a infra-estrutura de Internet e é considerada crítica quanto à segurança nacional e econômica e poderia ser impactada por ataques massivos.

Como disse Eugene Kaspersky , “Talvez não seja amanhã ou depois de amanhã – mas cedo ou tarde, terroristas usarão a Internet como outra arma em seu arsenal.” (Viruslist.com, 2004).

Então é possível avaliar que o problema está deixando o âmbito da “comunidade *underground*”, onde ataques são geralmente motivados pela curiosidade ou necessidade de reconhecimento dentro da comunidade, para caracterizar-se mais fortemente como crime, motivado por dinheiro ou convicções políticas e/ou religiosas.

3 PREVENÇÃO E COMBATE A ATAQUES DE NEGAÇÃO DE SERVIÇO

Segundo Zwicky, Cooper & Chapman, 2000, p. 385, proteger-se completamente contra ataques de negação de serviço é impossível.

Apesar de ser considerado por muitos um problema sem solução completa, existem muitas possibilidades de prevenção de ataques de DDoS, as quais, se não garantem proteção completa, evitam muitos dos tipos de ataques já conhecidos.

Neste capítulo serão discutidas algumas das principais possibilidades de prevenção.

As classificações realizadas são baseadas na proposta de Mirkovic (2003) e referem-se à Figura 2 apresentada no Capítulo 1.

3.1 MECANISMOS TRADICIONAIS

Os mecanismos tradicionais são componentes de segurança muito importantes na estratégia de defesa contra qualquer tipo de ataque.

O conjunto destes mecanismos geralmente representam as políticas de seguranças estabelecidas em uma organização.

3.1.1 BOAS PRÁTICAS DE ADMINISTRAÇÃO DE SISTEMAS E REDES

Mecanismos tradicionais, bastante eficientes mas absurdamente ainda pouco valorizados, são as boas práticas de administração de sistemas e redes, as quais recomendam, dentre outras coisas, a atualização dos sistemas imediatamente após a descoberta de falhas ou erros e a disponibilização das correções.

Outra recomendação importante refere-se à desativação de serviços não necessários em servidores, bem como a configuração dos serviços necessários modificando suas definições padrão quando exigido, como por exemplo, usuários e senhas.

Serviços como SNMP, geralmente disponíveis em equipamentos como *firewalls*, roteadores, *switches*, servidores de rede local, *no-breaks*, entre outros, muitas vezes estão disponíveis e com seus níveis de segurança configurados no padrão recebido de fábrica. É altamente recomendável que as configurações

padrão sejam alteradas antes de disponibilizar tais equipamentos em uma rede de computadores.

Oito recomendações básicas são feitas por Singer (2000), as quais se implementadas, minimizam consideravelmente as possibilidades de ocorrência de ataques de DoS ou mesmo a investigação de tais ocorrências.

Recomendações específicas para produtos comerciais como roteadores, *firewalls*, entre outros, podem ser obtidas facilmente junto aos seus fornecedores.

Este mecanismo encaixa-se na classificação proposta por Mirkovic (2003) na “Classificação por nível de atividade” dentro da categoria “Preventivo” (NA-1), na “Classificação por objetivo da Prevenção” na categoria “Prevenção de Ataque” (OP-1) e na “Classificação por objeto assegurado” na categoria “Segurança de Sistema” (OA-1).

3.1.2 FIREWALL

Como definido por Nakamura (2000):

> um *firewall* é um ponto entre duas ou mais redes, ponto este que pode ser um componente ou um conjunto de componentes, por onde passa todo o tráfego, permitindo que o controle e/ou autenticação e registros de todo o tráfego seja realizado. Assim, esse ponto único constitui um mecanismo utilizado para proteger, geralmente, uma rede confiável de uma rede pública, não-confiável.

A prevenção de ataques de negação de serviço em *firewalls* pode ser implementada com filtragem de requisições forjadas ou mal formadas ou ainda limitando recursos disponíveis ou mesmo balanceando o tráfego disponibilizado evitando que uma única origem monopolize serviços. (Zwicky, Cooper & Chapman, 2000, p. 327)

Avalia-se que um *firewall* tradicional não é completamente capaz de prevenir ataques distribuídos de negação de serviço. Mas, se os mecanismos de identificação, prevenção e combate de ataques de DDoS forem considerados como componentes de uma Arquitetura de *Firewall* e integrados de maneira cooperativa à ela, estes podem tornar-se mais eficazes.

Pode-se classificar este mecanismo dentro da proposta de Mirkovic (2003) na “Classificação por nível de atividade” dentro da categoria “Preventivo” (NA-1), na “Classificação por objetivo da Prevenção” na categoria “Prevenção de Ataque” (OP-1) e na “Classificação por objeto assegurado” na categoria “Segurança de Sistema” (OA-1).

3.1.3 IDS

Sistemas de detecção de intrusão baseiam-se em assinaturas, as quais são padrões de comparação que representam intenções maliciosas ou suspeitas.

Um IDS pode investigar tráfego direcionado a uma rede ou tráfego direcionado a um sistema específico, sendo assim os IDSs podem ser classificados em dois grupos IDS de rede (*Network* IDS) e IDS de sistema (*Host* IDS).

Firewalls tradicionais, por serem elementos situados entre as redes, se forem “derrubados” por um atacante em um ataque de DDoS, se bem configurados, continuarão protegendo a rede, já o IDS de rede é suscetível ao ataque de DDoS pois são elementos passivos e se forem “derrubados” tornam a rede desprotegida. (Ptacek & Newsham, 1998)

Classifica-se este mecanismo conforme a proposta de Mirkovic (2003) dentro da “Classificação por nível de atividade” na categoria “Preventivo” (NA-1), na “Classificação por objetivo da Prevenção” na categoria “Prevenção de Ataque” (OP-1) e na “Classificação por objeto assegurado” na categoria “Segurança de Sistema” (OA-1). Em alguns casos pode ser classificado na “Classificação por nível de atividade” como “Reativo” (NA-2), na “Classificação por estratégia de detecção de ataque” nas categorias “Padrão de Teste” (EDA-1) ou “Anomalia” (EDA-2).

3.2 PREVENÇÃO A PARTIR DAS APLICAÇÕES

Como mostrado no Capítulo 2, todos os esforços para conceber aplicações adequadas ao ambiente inóspito da Internet e conseqüentemente mais seguras devem ser garantidos. Algumas das recomendações para isso seriam:

- Não oferecer anonimamente recursos que exijam muito esforço computacional, como CGIs (Common Gateway Interface – Interface Comum de Comunicação), *servlets*, consultas a bases de dados relacionais ou geração e manutenção de identificadores de sessão, implementando portanto autenticação e autorização prévia à oferta de tais recursos;
- Limitar a oferta dos recursos computacionais, evitando que apenas um usuário consuma-os totalmente se abusar de uma aplicação;
- Implementar fortes esquemas de validação e filtragem, os quais devem ser aplicados para as entradas de dados feitas pelos usuários das

aplicações evitando a exploração indireta de outros recursos, como por exemplo os sistemas gerenciadores de banco de dados;

- E claro, escrever programas que controlem adequadamente sua memória temporária de armazenamento de dados (*buffers*).

Conforme a proposta de Mirkovic (2003) este mecanismo classifica-se dentro da “Classificação por nível de atividade” na categoria “Preventivo” (NA-1), na “Classificação por objetivo da Prevenção” na categoria “Prevenção de Ataque” (OP-1) e na “Classificação por objeto assegurado” na categoria “Segurança de Sistema” (OA-1).

3.3 PROTEÇÃO PARA USUÁRIO DOMÉSTICO E DE PEQUENOS NEGÓCIOS

Como é possível perceber o usuário doméstico e de pequenos negócios, os quais têm à sua disposição atualmente não só uma grande capacidade computacional como também cada vez mais capacidades de conexão (banda) com a Internet, necessitam de mecanismos mais eficientes de proteção para tais recursos.

Não é objetivo deste trabalho discutir a responsabilidade de cada participante na relação comercial de provimento de infra-estrutura para tais usuários, mas sim recomendar boas práticas que busquem evitar que os mesmos sejam participantes ou mesmo perpetradores de ataques de negação de serviço.

Todas as recomendações feitas no item 3.1.1 são válidas neste ítem, mas como nestes cenários tratam-se muitas vezes de um ou dois computadores conectados à Internet por um *cable modem* (modem de cabo) ou modem ADSL, pode-se complementar com os itens que se seguem.

A maioria das recomendações feitas se classificam na proposta de Mirkovic (2003) na “Classificação por nível de atividade” na categoria “Preventivo” (NA-1), na “Classificação por objetivo da Prevenção” na categoria “Prevenção de Ataque” (OP-1) e na “Classificação por objeto assegurado” na categoria “Segurança de Sistema” (OA-1).

POLÍTICAS DE SEGURANÇA

Primeiramente deve-se considerar a definição de uma política de segurança, mesmo que seja aplicável a apenas um computador, este é um ótimo ponto de partida.

A base para tal política devem ser as políticas dos provedores de infra-estrutura ou de acesso e conteúdo. Tais políticas podem (na verdade deveriam) incluir muitas das recomendações que serão apresentadas.

DESATIVAÇÃO DE SERVIÇOS DESNECESSÁRIOS

Sistemas operacionais quando instalados com suas opções padrão provenientes de fábrica muitas vezes ativam serviços que o usuário doméstico ou de pequenos negócios nunca utilizará. Estes serviços podem ser uma porta de entrada ou ainda um facilitador para atacantes que conheçam possíveis vulnerabilidades destes serviços.

Assim sendo, a desativação destes serviços é uma boa prática que é utilizada em servidores corporativos, a qual também deve ser utilizada em computadores domésticos ou de pequenos negócios, principalmente se conectados diretamente à Internet e por longos períodos de tempo.

ATUALIZAÇÃO/CORREÇÃO DE SOFTWARE E FIRMWARE

O usuário deve atualizar-se freqüentemente sobre as correções feitas pelos fabricantes de software, aplicando sempre que possível tais correções ou seguindo as recomendações destes fabricantes para evitar manter sistemas vulneráveis conectados à Internet.

O *firmware*[5] de equipamentos como *cable modem,* modem ADSL, roteadores, *bridges*, entre outros que estejam sendo utilizados para conectar os computadores à Internet também são suscetíveis a falhas e conseqüentes correções e atualizações. Portanto, o usuário deve manter-se atualizado também quanto a estes elementos.

ANTI-VÍRUS, ANTI-ADWARE/SPYWARE/MALWARE

A utilização de software antivírus, *anti-spyware*[6] (*adware* ou *malware*) e atualizá-los com freqüência é altamente recomendável.

---

[5] *Software* (programas ou dados) que foi escrito na *read-only memory* (ROM). *Firmware* é uma combinação de software e hardware. ROMs, PROMs e EPROMs que possuem dados ou programas gravados neles são firmware (Webopedia.com).

[6] Um *spyware* é qualquer *software* que secretamente captura informações do usuário de Internet sem o seu conhecimento, geralmente para propósitos de anúncios. São tipicamente embutidos como componentes ocultos de software gratuito ou de domínio público que podem ser obtidos via Internet (Webopedia.com). O *anti-spyware* busca por *spywares* instalados nos computadores e os elimina.

O recurso *anti-spyware* está sendo adicionado aos softwares de antivírus, mas já existem softwares específicos para eliminar este tipo de ameaça. Muitas vezes o simples hábito de eliminar com freqüência *cookies* e arquivos temporários associados ao navegador pode evitar que a grande maioria destas ameaças provoquem algum dano.

Existem ótimas opções de software gratuitos disponíveis na Internet para tratar destas ameaças. Mas, estes sistemas podem ajudar mas não são efetivos em todos os casos, principalmente quando da ocorrência de novos vírus ou cavalos-de-tróia.

Este tipo de mecanismo pode ser classificado conforme a proposição de Mirkovic (2003) na “Classificação por nível de atividade” dentro da categoria “Reativo” (NA-2) e na “Classificação por estratégia de detecção de ataque” na categoria “Padrão de Teste” (EDA-1).

FIREWALL PESSOAL

Se possível, utilizar um *firewall* pessoal complementa o mínimo necessário em termos de ferramentas de *software*.

Apesar de serem ferramentas que exigem um certo conhecimento técnico para configurá-las apropriadamente, são ótimas opções para evitar problemas com possíveis cavalos-de-tróia instalados ou mesmo para evitar a exploração de serviços vulneráveis.

Algumas recomendações são feitas por Hazari (2000) para a seleção de um bom *firewall* para este tipo de usuário.

E-MAIL

Critérios básicos para a utilização de sistemas de e-mail devem ser obedecidos, tais como:

- Evitar abrir e-mails de origens desconhecidas;
- Tratar e-mails com anexos com cautela evitando abrir tais anexos sem a adequada verificação por um anti-vírus;
- Não abrir apontadores para sítios desconhecidos ou suspeitos;
- Mesmo e-mails com origem conhecida devem ser cautelosamente tratados pois podem ter sido forjados.

IRC

Se o usuário for participante ou operador de canais IRC a recomendação básica, além das já descritas, é utilizar-se de ferramentas como o SwatIt (http://swatit.org/index.html), as quais podem ser indispensáveis para remover cavalos-de-tróia como o GTbot (NoHack.net).

Algumas dicas específicas para usuários de canais IRC podem ser encontradas nas páginas dos seguintes grupos:

- DALnet *Exploits Team* (Time de Explorações da DALnet) dedicado a avaliar possíveis explorações da infra-estrutura deste serviço. Sua localização atual é http://kline.dal.net/exploits/.;
- NoHack.net, o qual é mais dedicado a auxiliar na detecção e remoção de cavalos-de-tróia. A localização atual deste grupo é: http://www.nohack.net/.

3.4 PROTEÇÃO NOS PROVEDORES DE SERVIÇOS DE INTERNET

Grandes instituições de ensino e pesquisa, órgãos governamentais e grandes corporações, formam a infra-estrutura que atualmente constitui o que é conhecido por Internet.

Os recursos para prover esta infra-estrutura são realmente bastante dispendiosos financeiramente e eram dimensionados no passado para suprir demandas esperadas de até 5 anos ou mais, mas com o crescimento exponencial do seu uso estes recursos acabam atingindo sua capacidade em dois ou três anos.

Alguma economia é feita por certas corporações e a vida útil destes recursos é estendida da pior maneira possível, isto é, mantendo-se equipamentos que não suportariam o tráfego atual se implementados os requisitos de segurança exigidos atualmente. Desta forma, deixa-se de inserir filtros de tráfego muitas vezes básicos em equipamentos que poderiam ter sua performance comprometida criando possibilidades para diversos tipos de exploração da infra-estrutura da Internet.

Esta negligência só está se modificando por causa dos impactos criados por ataques de DDoS aos negócios destas corporações. Algumas destas, grandes ISPs do mercado mundial, vêm investido em iniciativas para melhorar este quadro (Pappalardo, 2003 e Techweb News, 2004). Mas, ainda há muito o que ser feito e

alguns consideram que algumas imposições legais deveriam ser feitas a estes fornecedores de infra-estrutura.

Além dos mecanismos tradicionais já descritos no item 3.1, algumas medidas baseadas na RFC-2827 (Ferguson e Senie, 2000) devem ser tomadas pelos ISPs para contribuir com o bom funcionamento e a boa utilização da Internet.

Como visto no Capítulo 2, a negação de serviço nos servidores de nomes (DNS) raiz pode ser catastrófica e a preocupação tem sua razão de ser, pois isso já foi tentado.

Karl Auerbach, o qual foi membro da ICANN (Corporação da Internet para Atribuição de Nomes e Endereços, Internet Corporation For Assigned Names and Numbers) até junho de 2003, propôs uma série de recomendações de segurança para proteger o sistema de resolução de nomes da Internet (Auerbach, 2001).

Dentre as recomendações de Auerbach, a confecção de um esqueleto de regras de filtragem a serem inseridas nos roteadores dos ISPs, segundo ele, minimizariam o tempo de recuperação do sistema para apenas alguns minutos ao invés de horas. Espera-se que tais recomendações tenham sido seguidas.

Os mecanismos citados podem ser classificados conforme a proposição de Mirkovic (2003) na “Classificação por nível de atividade” dentro da categoria “Preventivo” (NA-1) e na “Classificação por objetivo da Prevenção” nas categorias “Prevenção de Ataque” (OP-1) e “Prevenção de DDoS” (OP-2).

3.5 MECANISMOS DE COMBATE, PREVENÇÃO E RASTREAMENTO DE ATAQUES DE DDOS

Além das boas práticas de administração de sistemas, dos mecanismos tradicionais de segurança da informação e das recomendações específicas para provedores de serviço e usuários de Internet, existem mecanismos específicos, como visto no Capítulo 1, destinados a tratar o problema de ataques distribuídos de negação de serviço.

Mirkovic (2003) realizou análises mais específicas para a categorização mais detalhada destes mecanismos, mas de uma maneira geral é possível categorizar tais mecanismos em alguns poucos grandes grupos de abordagem:

- Mecanismos baseados na vítima;
- Mecanismos baseados em roteadores;
- Mecanismos baseados na origem, e;

- Soluções cooperativas.

Nos itens que se seguem serão discutidas cada uma destas abordagens.

3.5.1 MECANISMOS BASEADOS NA VÍTIMA DE ATAQUES

O principal interessado em evitar ataques de negação de serviço sem dúvida é a vítima. Em comparação com a abordagem baseada em roteadores, esta tem a vantagem de poder ser implementada imediatamente (Jin, Wang & Shin, 2003).

Os mecanismos que adotam este tipo de abordagem utilizam-se de variadas técnicas como validação de endereços de origem por traçado reverso, agrupamento e contabilização de fluxos e limitação de tráfego para fluxos suspeitos.

Porém, muitos destes mecanismos são reativos e as estratégias empregadas para combater o problema não podem estar restritas a apenas detectar ocorrências de DDoS, elas devem ser capazes de minimizar os impactos destas sem com isso interferir significativamente no tráfego legítimo.

O maior problema deste tipo de abordagem é que, apesar de demonstrarem-se efetivos (Kong *et al.*, 2003), os recursos geralmente são consumidos pelo menos até o ponto onde o mecanismo é implementado dentro do caminho dos fluxos, isto é, haverá consumo desnecessário de recursos neste caminho até o ponto da implementação do mecanismo e com isso os fluxos legítimos podem ser de alguma forma impactados.

Estes mecanismos devem ser capazes também de identificar que tipo de estratégia esta sendo utilizada no ataque, isto é, se este ataque envolve uma única origem ou diversas origens. Isto é necessário para ser possível escolher a estratégia adequada de combate, a qual consuma a menor quantidade recursos possível. Neste sentido, um *Framework* como o proposto por Hussain, Heidemann & Papadopoulus (2003) seria bastante útil.

De qualquer forma, mecanismos para identificar anomalias de tráfego devem ser utilizados por todos os sítios dentro de uma arquitetura de segurança resiliente, primeiramente porque são potenciais vítimas e também porque as possibilidades de ocorrência de eventos de negação de serviço não estão restritas

apenas a eventos de ataques. Tais mecanismos devem auxiliar também no controle de disponibilidade dos sítios, bem como no planejamento de capacidades destes.

Alguns fabricantes como Riverhead Networks (recentemente adquirida pela Cisco Systems) têm oferecido soluções com esta abordagem (Riverhead Networks), as quais já estão sendo utilizadas por grandes provedores de serviço de Internet e vêm sendo agregadas ao portfólio de serviços diferenciados destes ISPs.

Nesta categoria encontram-se mecanismos como SYN *cookies*, (Bernstein), FDS (Flooding Detection System - Sistema de Detecção de Inundação) proposto por Haining, Zhang & Shin (2002), entre outros.

3.5.2 MECANISMOS BASEADOS EM ROTEADORES

Roteadores são componentes estruturais da Internet, isto significa dizer que sem o seu auxílio não é possível a comunicação entre as redes que a constituem.

As relações de confiança são inerentes aos serviços que os roteadores provêem, pois por meio de protocolos padronizados como RIP, BGP, OSPF, entre outros, estes trocam diversas informações necessárias para realizar seus serviços. Para tanto, é necessário que cada qual confie nas informações oferecidas pelo outro.

Nesse sentido, pode-se avaliar que mecanismos de combate ao problema de DDoS e que se utilizem destas relações de confiança para tal finalidade podem se tornar bastante efetivos. Cabe no entanto, aos fabricantes e à comunidade da Internet, propor e conceber mecanismos para reforçar tais relações de maneira a evitar que estas sejam exploradas indevidamente.

São exemplos deste tipo de abordagem o mecanismo SAVE (*Source address validity enforcement protocol* – Protocolo de validação forçada de endereço de origem) proposto por Li *et al.* (2001), o traçado reverso de roteamento IP (IP *traceback*) proposto por Sung & Xu (2002) e o *pushback* proposto por Mahajan (2001).

O mecanismo *pushback*, se implementado em todos os roteadores, poderia se tornar bastante efetivo na limitação de tráfego de fluxos de ataques distribuídos de negação de serviço. Mas, este é bastante dependente da referida relação de confiança que deve existir entre os dispositivos.

Neste ponto, além do reforço necessário a estas relações, existe um paradigma a ser quebrado, pois muitos consideram não ser possível confiar em dispositivos para os quais não se exerça poder administrativo. Provavelmente somente proposições que provem ser realmente seguras poderão derrubar tal obstáculo. Este assunto está fora do escopo deste trabalho, mas existem possibilidades de reforço nas relações de confiança utilizando-se, por exemplo, Keynote (Blaze *et al.*, 1999).

3.5.3 MECANISMOS BASEADOS NA ORIGEM DE ATAQUES

A estratégia comumente utilizada para a resolução de outros problemas também pode ser aplicada para o problema de DDoS: a solução de um problema deve visar inicialmente sua origem.

Como já foi avaliado neste trabalho, algumas condições básicas existem na Internet para a construção de ataques de DDoS.

Avaliando-se estas condições é possível verificar que muito pode ser feito próximo à origem dos ataques para evitar que estes sejam iniciados. Portanto, algumas ações imediatas podem ajudar muito no combate do problema, mesmo sem a implementação de mecanismos especializados, como aquelas recomendadas pelo Mitre Cyber Resource Center Team (2000).

A proposição feita por Mirkovic (2003), com o sistema D-WARD reforça este tipo de abordagem, a qual é muito mais racional em termos de economia de recursos pois reduz o seu consumo minimizando a propagação para a Internet dos efeitos de possíveis tentativas de ataques.

3.5.4 SOLUÇÕES COOPERATIVAS

As soluções cooperativas envolvem a implementação de mecanismos em roteadores e servidores, tanto de potenciais vítimas como potenciais origens ou participantes de ataque e mesmo em provedores de serviços de Internet. Estas soluções integram mecanismos que devem trabalhar cooperativamente no controle e minimização dos efeitos de ataques de DDoS.

Nesta abordagem alguns problemas podem ser levantados:

a) a necessidade de reforçar as relações de confiança;
b) dependendo da implementação, a necessidade de instalação e configuração de suporte a novos protocolos ou novas funcionalidades.

Apesar destes problemas as soluções cooperativas sugerem ser mecanismos mais efetivos e mais rápidos no combate ao problema do que aqueles implementados isoladamente.

Dentre as avaliações realizadas por Mirkovic (2003) uma delas integrou o sistema D-WARD com o sistema de detecção de DDoS chamado COSSACK. Nesta integração o sistema COSSACK trabalhou cooperativamente de forma a auxiliar o sistema D-WARD na limitação de tráfego de fluxos originados nas redes onde D-WARD era implementado. Nas várias simulações de ataques conduzidas a solução cooperativa demonstrou-se bastante efetiva.

3.5.5 ANÁLISE COMPARATIVA

Na introdução deste trabalho alguns modelos e mecanismos de combate, prevenção e rastreamento de ataques de DDoS foram citados, os quais são apresentados na Tabela 1 quanto suas vantagens e desvantagens. Também estão incluídas nesta tabela proposições quanto ao aproveitamento mais adequado ou aprofundamento do estudo de cada modelo/mecanismo.

Tabela 1 – Comparativo entre os mecanismos

Modelo/Mecanismo	Vantagens	Desvantagens	Proposição
Qualidade de Serviço como Ferramenta de Segurança	- Para casos específicos pode garantir a disponibilidade de serviços em sítios da Internet mesmo sob ataques de DDoS; - Pode ser implementado de forma transparente para o usuário final.	- Muito dependente de relações de confiança estabelecidas por SLAs e de recursos de Qualidade de Serviço ainda não amplamente implementados.	- Integrar com mecanismos baseados na origem de ataques DDoS, capazes de evitar que sistemas iniciem ataques ou participem destes
VIPNet	- Pode garantir a disponibilidade de serviços em sítios que o implementem, mesmo sob ataques de DDoS.	- Depende da utilização de componentes proprietários e da “aquisição” de recursos por parte do usuário final, portanto não é transparente.	- Utilizar em ambientes onde a contabilização do uso dos recursos seja importante tanto para o fornecedor como para seu cliente ou o uso sob demanda seja um

			serviço diferenciado e tarifado à parte.
Controle de Agrupamentos de Grande Largura de Banda	- Limita tráfego em situações de ataque; - Pode ser implementado tanto em dispositivos de borda, quanto em servidores.	- A política de implementação da criação dos agrupamentos de fluxos pode interferir negativamente na infra-estrutura se roteadores adjacentes definirem políticas diferentes de criação de agrupamentos. - Dependente do bom funcionamento dos roteadores da infra-estrutura para ser efetivo.	- Utilizar em roteadores de núcleo sob a mesma administração, onde o estabelecimento das relações de confiança entre roteadores e critérios de agrupamento de fluxos possam ser definidos globalmente.
D-WARD	- Cria maiores dificuldades para a utilização de computadores da rede que o implementa em ataques DDoS.	- No caso do uso de servidores da rede como refletores somente critérios de agrupamento baseados em endereçamento de destino podem ser efetivos; - Não detecta a maioria dos ataques com endereços não forjados que utilizam pacotes UDP.	- Pode ser utilizado como pré-requisito na avaliação de “confiabilidade” de ISPs. - Integrado a outros mecanismos pode obter um nível maior de efetividade.
SOS/WEBSOS	- Alto nível de redundância garantindo disponibilidade de serviço; - Voltado para o fornecimento de serviços seguros.	- Dependente de relações de confiança; - A implementação de uma infra-estrutura com muitos componentes, cria uma grande carga administrativa; - Não deve ser utilizado se a	- Considerar sua implementação apenas em ambientes onde além da alta disponibilidade, redundância também seja um pré-requisito.

		latência for um problema para o serviço disponibilizado; - A autenticação dos nós é baseada em certificados emitidos pela mesma autoridade de certificação (CA). Neste caso o comprometimento da PKI destruiria a rede sobreposta.	
Protocolo de validação forçada de endereço de origem	- Procura evitar a utilização de endereços falsificados (spoofing). - Simplificação de IDSs.	- Necessita implementação em larga escala para ser considerado efetivo.	- Considerar sua implementação em provedores que se interligam a outros provedores, principalmente internacionais para validação de endereçamento do tráfego originado fora de seus limites.
Filtragem Inteligente de Pacotes Baseada em Traçado de Rota IP	- Aumenta a velocidade no tráfego de fluxos “legítimos” durante um ataque de DDoS. - O traçado reverso de roteamento pode ser reconstruído de maneira mais precisa.	- Também necessita implementação em larga escala para ser considerado efetivo. - Alterações necessárias na marcação de campos do cabeçalho IP podem acarretar efeitos inesperados.	- Avaliar mais detalhadamente os critérios probabilísticos de classificação dos fluxos.

3.6 CONCLUSÃO

O objetivo maior de qualquer mecanismo de defesa ou prevenção deve ser minimizar os impactos de ataques de negação de serviço sem com isso negar serviço ou causar problemas aos clientes legítimos.

Muitas são as abordagens para os problemas de DoS e DDoS, mas muitos problemas poderiam ser evitados ou minimizados com algumas soluções

simples. Por outro lado, mesmo estas soluções simples não eliminam as possibilidades de perpetração destes ataques, os quais estão se tornando cada vez mais sofisticados e freqüentes como abordado no Capítulo 2.

Os problemas causados pela negação de serviço simples aparentemente seriam também mais simples de se resolver. Mas, com a crescente disponibilização de aplicações na Internet, desenvolvidas sob os mais variados níveis de rigor técnico, as soluções caracterizam-se mais complexas.

Mirkovic (2003) concluiu que a seletividade e efetividade da resposta aos ataques melhora a medida que o mecanismo de defesa é movido para mais perto da origem, distanciando-se da vítima, mas desta forma a exatidão na detecção é deteriorada.

Percebe-se portanto, que uma solução distribuída e cooperativa que envolva minimamente políticas e mecanismos baseados na origem e, políticas e mecanismos implementados nas potenciais vítimas é a abordagem mais adequada para o conjunto de problemas relacionados com a negação de serviço.

4 AVALIAÇÃO EXPERIMENTAL DE PROTÓTIPO DE MECANISMO DE PROTEÇÃO CONTRA ATAQUES DE DDOS

4.1 OBJETIVOS

Este capítulo destina-se a descrever a avaliação experimental de protótipo de um dos modelos propostos para a minimização dos impactos de ataques de DDoS. A meta desta avaliação experimental é demonstrar uma implementação efetiva do modelo proposto por Meylan (2003), além de evidenciar uma possibilidade de minimizar os referidos impactos.

4.2 DIFFSERV

A arquitetura de Serviços Diferenciados (DiffServ) foi concebida para suprir as deficiências encontradas em outra arquitetura de QoS, o IntServ (Serviços Integrados), principalmente no que tange à escalabilidade.

A introdução do conceito de agregações de fluxos ou BA (*Behavior Aggregate*), o provisionamento de capacidades de recursos para os BAs e a separação das funções dos roteadores de borda e de núcleo objetivam buscar esta escalabilidade.

Na Figura 25 a arquitetura lógica do DiffServ mostra alguns dos seus componentes, onde Domínio DS é uma rede que suporta requisições de QoS com DiffServ.

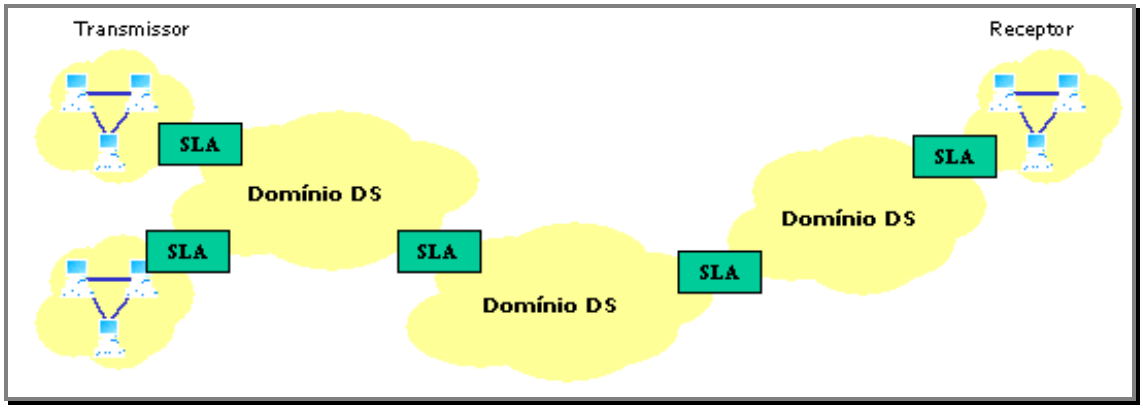

Figura 25 - Arquitetura lógica DiffServ

Kamienski e Sadok, 2000

Em uma comunicação fim-a-fim entre um Transmissor e um Receptor é possível que os fluxos trafeguem entre diversos Domínios DS e para que sejam garantidos os requisitos de QoS para tais fluxos é necessário o estabelecimento de SLAs entre esses domínios, os quais incluam detalhadamente os parâmetros de desempenho de serviço esperados, bem como as formas de cobrança e tarifação.

Os fluxos são agrupados de acordo com a identificação do DSCP (DS *Code Point*) que é registrada no campo DS do cabeçalho IP. O agrupamento é tratado em cada roteador do Domínio DS da mesma forma e essa forma de tratamento para encaminhamento é chamada de PHB (*Per-Hop Behavior* – Tratamento por Salto).

Grupos de PHBs foram padronizados pelo IETF (*Internet Engineering Task Force* – Força Tarefa de Engenharia da Internet):

- EF (Expedited Forwarding - Encaminhamento Expresso) - Com o grupo PHB EF é possível construir um serviço fim-a-fim através de domínios DS com baixa perda, baixa latência, baixa variação de atraso (jitter) e largura de banda assegurada (Jacobsen, Nichols e Poduri, 1999).
- AF (Assured Forwarding - Encaminhamento Assegurado) - O grupo PHB AF oferece 4 diferentes classes de encaminhamento, cada qual com três níveis de precedência de descarte. Desta forma, os recursos de encaminhamento, como memória para armazenamento temporário (buffers) e largura de banda, podem ser alocados para cada classe (Heinanen et all., 1999).
- CS (Class Selector – Seletor de Classe) – Este grupo foi concebido para manter compatibilidade com o uso previamente concebido para o campo IP de precedência sem limitar a flexibilidade futura (Nichols et al., 1998).

A Tabela 2, emprestada de Balliache, 2003, mostra os possíveis valores as classes AF1 a AF4, além da classe EF, onde:

- DP - Drop Precedence (Precedência de Descarte);
- b-DSCP – valor DSCP em binário (dois zeros adicionados à esquerda do valor do DSCP);
- x-DSCP – valor DSCP em hexadecimal;
- b-DS – valor do campo DS em binário (dois zeros adicionados à esquerda do valor do DSCP);
- x-DS – valor do campo DS em hexadecimal.

Tabela 2 – Classes AF e EF e seus valores DSCP e DS

Classe	DP	DSCP	b-DSCP	x-DSCP	b-DS	x-DS
AF1	1	001010	0000-1010	0xa	0010-1000	0x28
	2	001100	0000-1100	0xc	0011-0000	0x30
	3	001110	0000-1110	0xe	0011-1000	0x38
AF2	1	010010	0001-0010	0x12	0100-1000	0x48
	2	010100	0001-0100	0x14	0101-0000	0x50
	3	010110	0001-0110	0x16	0101-1000	0x58
AF3	1	011010	0001-1010	0x1a	0110-1000	0x68
	2	011100	0001-1100	0x1c	0111-0000	0x70
	3	011110	0001-1110	0x1e	0111-1000	0x78
AF4	1	100010	0010-0010	0x22	1000-1000	0x88
	2	100100	0010-0100	0x24	1001-0000	0x90
	3	100110	0010-0110	0x26	1001-1000	0x98
EF		101110	0010-1110	0x2e	1011-1000	0xb8

SEGURANÇA

Alguns aspectos relacionados à segurança na implementação de DiffServ devem ser considerados:

- Um atacante poderia modificar o DSCP de seus fluxos de forma a “roubar” serviço como se fosse um serviço avançado. Dependendo da proporção deste “roubo” o atacante pode provocar uma condição de negação de serviço.
- Conforme a RFC 2475 a “combinação do condicionamento de tráfego nos limites de nós DS aliado à segurança e integridade da infra-estrutura de rede dentro de um domínio DS” deve ser utilizada como defesa contra o “roubo” ou a negação de serviço.

4.3 IMPLEMENTAÇÃO DE DIFFSERV EM LINUX

Para o desenvolvimento do protótipo proposto neste trabalho, foram utilizadas requisições de QoS baseadas em DiffServ, o que pode ser implementado em Linux utilizando uma mistura de filtragem e roteamento de pacotes.

A marcação dos pacotes é feita no campo DS do cabeçalho IP conforme especificado na RFC2474 (Nichols *et al.*, 1998), anteriormente

denominado campo de Tipo de Serviço (TOS). O tratamento do tráfego é diferenciado de acordo com o valor especificado neste campo.

As duas ferramentas básicas para manipulação de pacotes, atreladas ao *Kernel* (núcleo) do Linux (Linux Kernel) são o iptables e o tc (*traffic control* – controle de tráfego).

O iptables, em conjunto com o netfilter, são componentes do framework de filtragem de pacotes, tradução de endereçamento de rede e porta (NAT) e outras manipulações de pacotes do Linux 2.4.

A associação de pacotes a uma determinada classe DiffServ pode ser feita em Linux utilizando-se do iptables para atribuir valores ao campo DSCP. Também pode ser utilizado para aplicar regras específicas de encaminhamento ou bloqueio baseadas no conteúdo do DSCP.

A ferramenta ***tc*** é parte do pacote **iproute** que acompanha as distribuições Linux. Ela tem por objetivo exibir e manipular configurações de controle de tráfego. O processamento deste controle de tráfego é feito no *kernel* do sistema onde é controlado por três tipos de objetos: disciplinas ou critérios de enfileiramento (qdiscs), classes e filtros. A qdisc DSMARK é uma disciplina de enfileiramento especificamente criada para manipular requisições DiffServ.

A qdisc GRED (*Generalized Random Early Detection* – Detecção Aleatória Adiantada Generalizada) foi implementada com objetivo de tratar múltiplas prioridades de descarte, o que é necessário para tratar o grupo PHB AF. Com esta qdisc é possível utilizar quatro prioridades de atraso.

DSMARK

A Figura 26 mostra o esquema de manipulação do conteúdo do DSCP de cada pacote recebido utilizando a qdisc DSMARK, a qual é uma disciplina de enfileiramento especificamente criada para manipular requisições DiffServ.

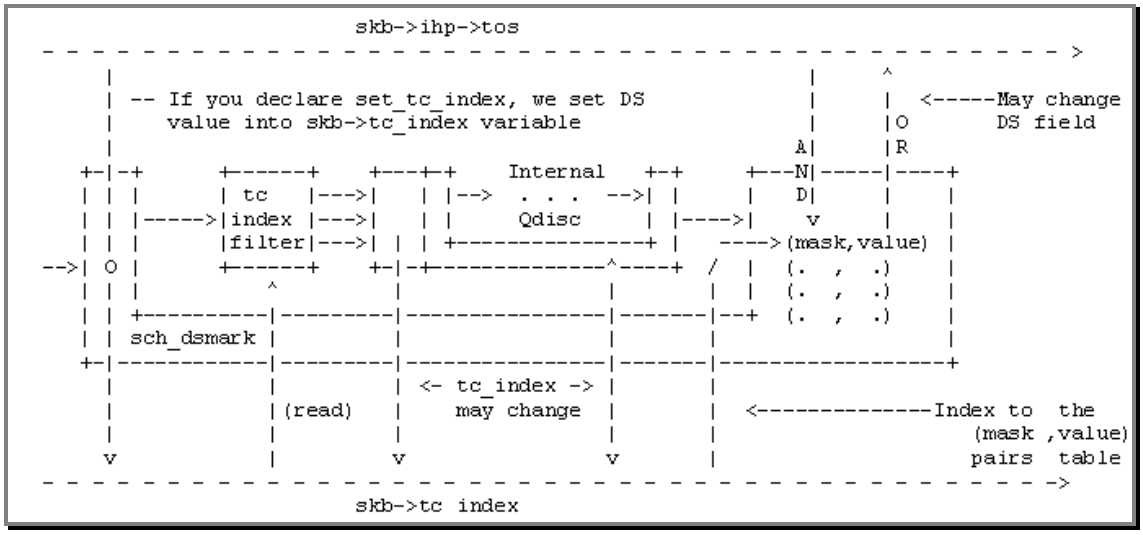

Figura 26 - Como SCH_DSMARK funciona

(Hubert et al., 2003)

O comando abaixo cria uma nova *qdisc* DSMARK com identificador 1:0 atrelada à interface eth0, onde o tamanho 64 é definido para uma tabela de armazenamento dos pares (máscara, valor). O parâmetro final do comando, *set_tc_index*, instrui a disciplina DSMARK a armazenar o conteúdo do campo DS no campo skb->tc_index da estrutura sk_buff implementada em Linux.

```
# tc qdisc add dev eth0 handle 1:0 root dsmark indices 64 \
set_tc_index
```

O campo skb->tc_index será então utilizado para selecionar a classes dentro desta disciplina de enfileiramento e também poderá receber novos valores de acordo com os critérios de classificação aplicados.

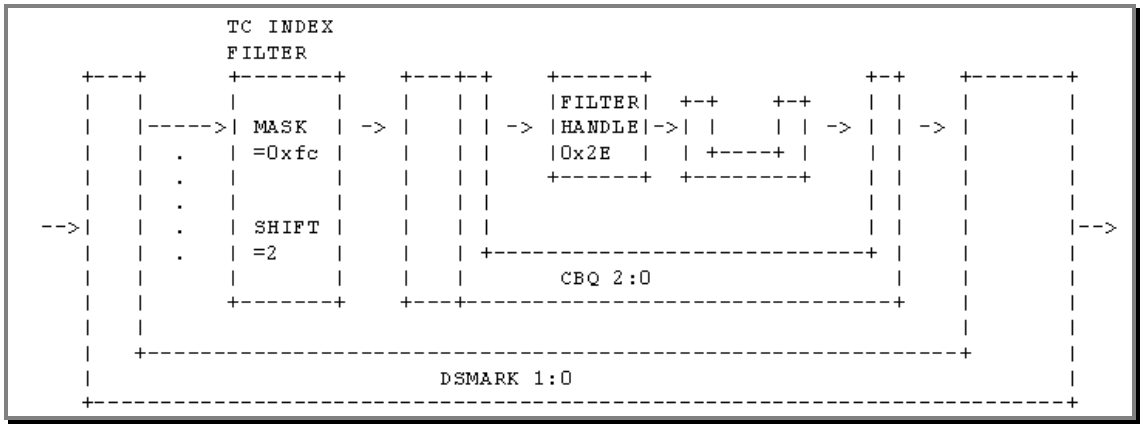

Figura 27 - Filtro TC_INDEX

(Hubert et al., 2003)

Esta qdisc também pode ser utilizada para modificar o conteúdo do campo DSCP dos pacotes. Um exemplo de sua utilização para esta finalidade seria:

```
# tc class change dev eth0 classid 1:1 dsmark mask 0x3 value 0xb8
```

Neste exemplo os pacotes referentes à classe 1:1 associada à interface eth0 seriam modificados. No caso de um pacote com DSCP 0x0 (melhor esforço), seu valor após modificado pela qdisc dsmark seria 0xb8.

GRED

A qdisc GRED (*Generalized Random Early Detection* – Detecção Aleatória Adiantada Generalizada) foi implementada com objetivo de tratar múltiplas prioridades de descarte, o que é necessário para tratar o grupo PHB AF. Os bits menos significativos de skb->tc_index são utilizados para selecionar a classe de descarte e então a série correspondente de parâmetros RED (Almesberger, Salim & Kuznetsov, 1999).

Com esta qdisc é possível utilizar quatro prioridades de atraso.

A Tabela 3 lista os parâmetros de configuração disponíveis para esta qdisc.

Tabela 3 – Parâmetros de configuração da qdisc GRED

Limit	define o limite “físico” da fila virtual em bytes
Min	define o valor limite mínimo em bytes
Max	define o valor limite máximo em bytes
Avpkt	é o tamanho médio de pacote em bytes
Bandwidth	é a taxa de transferência nominal da interface
Burst	é o número de pacotes de tamanho dentro da média autorizados a serem disparados
Probability	define a probabilidade de descarte no intervalo
DP	identifica a fila virtual atribuída a estes parâmetros
Prio	identifica a prioridade da fila virtual se o parâmetro grio foi definido nos parâmetros gerais

4.4 DISPOSITIVO INTERMEDIÁRIO DE ENFILEIRAMENTO

O IMQ não é considerado uma *qdisc* mas sua utilização é muito próxima a isso. Ele é utilizado em Linux como uma interface virtual para onde todo o tráfego é direcionado, baseado em regras pré-definidas, e onde *qdisc* podem ser associadas. Desta forma, regras de condicionamento de tráfego podem ser especificadas, inclusive, para tráfego de entrada.

Para redirecionar tráfego de uma interface física para uma interface IMQ utiliza-se o iptables (Netfilter, 2003).

Este não é um recurso disponível no *kernel* oficial do Linux e para ser possível utilizá-lo deve-se aplicar alterações a este *kernel* e ao iptables. Atualmente estas alterações (*patch*) podem ser obtidas no sítio http://www.linuximq.net.

4.5 CARACTERÍSTICAS DO PROTÓTIPO

O protótipo apresentado, baseado no modelo proposto por Meylan (2003), foi construído com ferramentas de código aberto e facilmente obtidas na Internet.

O referido modelo tem por finalidade manter a disponibilidade seletiva de serviços em situações de ataque DDoS a um determinado sítio, para clientes que acessem tal sítio a partir de provedores “confiáveis”. Para tal avaliação foram utilizadas requisições com características de QoS para diferenciar o tráfego de rede que ocorrera entre o provedor de acesso e um roteador localizado no sítio da vítima, o qual encarregou-se de redirecionar o tráfego aos servidores reais. O comportamento do protótipo foi avaliado em situações simuladas de ataques DDoS.

Os principais recursos de software utilizados para configurar o ambiente foram:

- O sistema operacional Linux, distribuição Fedora Core 1 (Linux, 2003; Fedora Project, 2003);
- O subsistema de *firewall* do Linux v.2.4 (Netfilter, 2003), composto pelo *netfilter* e *iptables*;
- Os utilitários iproute e tc para Linux (Hubert *et al.*, 2003);
- LVS (*Linux Virtual Server* – Servidor Virtual Linux) (Zhang, 2000) para implementação do balanceamento de carga.

O componente “Distribuidor de Conexões” foi implementado e os seguintes componentes foram configurados:

- Roteador;
- Condicionador de tráfego;
- Balanceador de carga;
- Marcador de pacotes diferenciados;
- Cliente com tráfego diferenciado;
- Cliente comum ou genérico;
- Atacante;
- Servidor *web* virtual;
- Servidor *web* destinado a tráfego diferenciado;
- Servidor *web* destinado a tráfego genérico;

Um programa escrito em *shell script* e baseado em exemplo que acompanha o pacote iproute2, foi utilizado no componente “Condicionador de tráfego” para inicializar e parar as disciplinas de enfileiramento, as classes e os filtros associados ao dispositivo virtual “imq0”. Ainda era possível utilizar este programa para mostrar estatísticas associadas às classes e disciplinas de enfileiramento. Foram definidas com o utilitário **tc** quatro classes AF e uma classe para melhor esforço (BE – *Best Effort* ou Melhor Esforço). A qdisc GRED foi utilizada como definida em Almesberger, Salim & Kuznetsov (1999) com 3 prioridades de atraso dentro de cada classe AF.

No componente “Marcador de pacotes diferenciados“ um programa escrito em *shell script* foi utilizado para inicializar e parar a marcação de pacotes com DSCP especificando a classe AF esperada pelo “Servidor *web* destinado a tráfego diferenciado”. Regras de encaminhamento com marcação de pacotes foram implementadas neste programa com **iptables**.

Adicionalmente aos componentes configurados foi desenvolvido um programa em s*hell script* para controlar o número total de conexões entrantes no “Balanceador de carga”. Este programa também foi utilizado para inclusão ou remoção de regras de encaminhamento de tráfego construídas com **iptables**, inclusão e remoção de servidor *web* no esquema de balanceamento de carga, além do acionamento ou desativação do “Distribuidor de conexões” e do “Condicionador de tráfego” de acordo com os limites de conexões estipulados.

A lógica desenvolvida para ativação e desativação do “Distribuidor de conexões” é apresentada a seguir:

```
swdtd = Servidor web destinado a tráfego diferenciado
bc    = Balanceador de carga
dc    = Distribuidor de conexões
ct    = Condicionador de Tráfego
lmce  = Limite máximo de conexões entrantes
lddc  = Limite para desativação do distribuidor de conexões

Ativar bc entre os servidores web
Enquanto verdadeiro
      sce = somatório das conexões entrantes
      Se sce >= lmce e dc(desativado)
            Excluir swdtd do bc
            Ativar ct
            Ativar dc
      Se sce <= lddc e dc(ativado)
            Desativar ct
            Desativar dc
            Incluir swdtd no bc
```

4.6 METODOLOGIA DE TESTES

No ambiente de testes foram representadas condições comumente encontradas atualmente na Internet. Um sítio de comércio eletrônico foi contemplado neste ambiente, bem como o atacante e os clientes *web* que acessam os recursos compartilhados deste sítio, o qual também implementa balanceamento de carga.

O tempo de duração de um ataque de DDoS varia, mas a grande maioria dos ataques observados na Internet atualmente duram entre 3 e 20 minutos, mas alguns ataques duram menos do que um minuto (Moore, Voelker & Savage). Para os testes realizados no ambiente apresentado foram simulados ataques com duração aproximada de um minuto.

Nos testes foram utilizados 6 microcomputadores PCs em uma rede FastEthernet de 100 Mbps que conectava seus componentes. A Figura 28 mostra a topologia utilizada em laboratório e a Tabela 4 descreve algumas características dos componentes utilizados.

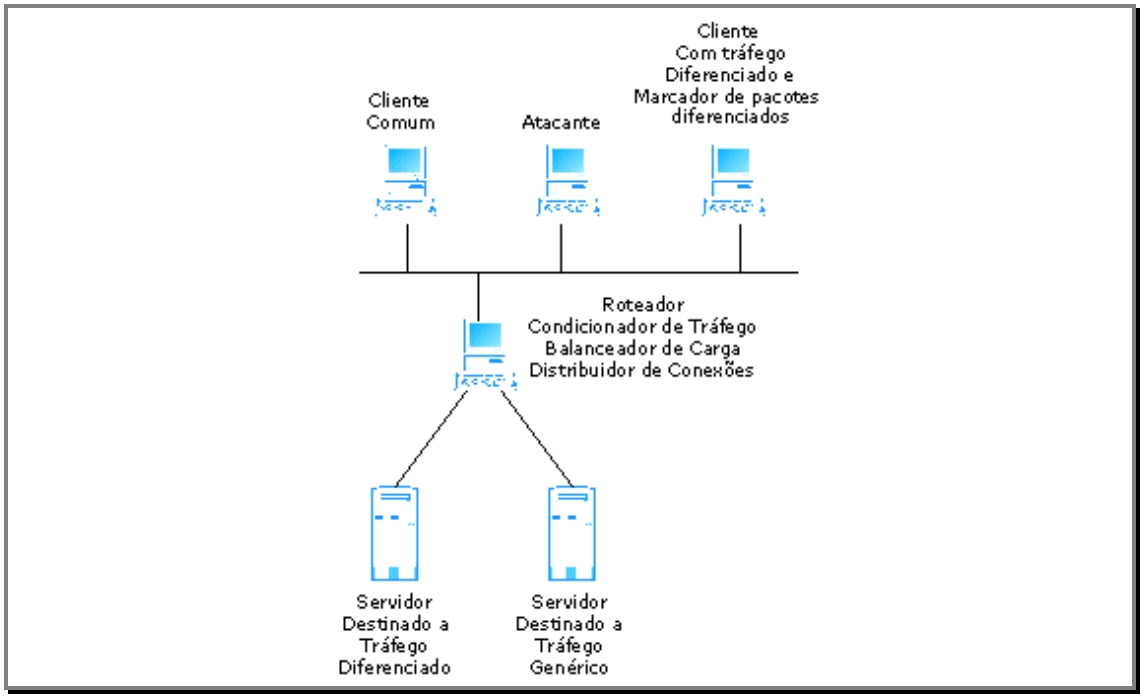

Figura 28 - Laboratório do Protótipo Experimental

O *kernel* Linux do roteador, bem como o iptables, foram recompilados com o *patch* (componente de atualização) IMQ (McHardy, 2003) para possibilitar a implementação do enfileiramento de pacotes recebidos e o seu tratamento diferenciado de acordo com o campo DSCP de cada pacote.

Tabela 4 – Equipamentos utilizados na implementação do protótipo

Entidade	**Equipamento**	**Sistema Operacional**
Cliente Comum	Pentium IV 2.0 Mhz 256 MB RAM	Windows XP
Atacante	Pentium IV 2.0 Mhz 256 MB RAM	Linux Fedora Core 1
Cliente com Tráfego Diferenciado	Pentium IV 2.0 Mhz 256 MB RAM	Linux Fedora Core 1
Roteador	Pentium IV 2.0 Mhz 256 MB RAM	Linux Fedora Core 1
Servidor Destinado a Tráfego Diferenciado	Pentium IV 2.0 Mhz 1 GB RAM	Linux Fedora Core 1
Servidor Destinado a Tráfego Genérico	Pentium IV 2.0 Mhz 1 GB RAM	Linux Fedora Core 1

O link simulado foi dividido entre as 5 classes definidas (quatro classes AF e uma classe de melhor esforço).

O valor de *tcp_max_syn_backlog* foi mantido no padrão do sistema operacional que era 1024 em todos os componentes servidores. Este parâmetro especifica o número máximo de requisições de conexão, as quais ainda não foram reconhecidas pelo cliente e que devem ser lembradas. Em um ataque de inundação SYN este valor pode ser atingido facilmente.

O LVS somado a regras de NAT criaram a representação do “Servidor web virtual” na entidade “Roteador”, a qual também comportou os componentes “Condicionador de Tráfego”, “Balanceador de Carga” e “Distribuidor de conexões”.

Durante o ataque simulado em um ambiente com a participação do “Cliente com tráfego diferenciado”, enquanto o “Atacante” simulava um ataque de DDoS gerando tráfego com synful (synful.c) que totalizou um milhão de pacotes direcionados ao endereço do “Servidor *web* virtual” e à sua porta TCP 80 (HTTP), o “Cliente comum” ou “Genérico” tentava acessar o sitio hospedado no “Servidor *web* virtual” (isto é, um sítio replicado em ambos os servidores *web*) gerando tráfego genérico a partir da ferramenta MS *Web Application Stress Tool* (ferramenta de teste de carga de aplicação w*eb*) (Microsoft *Corporation*) mas não era capaz de atingir seu objetivo. Ao mesmo tempo o “Cliente com tráfego diferenciado” gerava tráfego com Flood (The Apache Software Foundation) e o “Marcador de pacotes diferenciados” aplicava a marcação necessária de DSCP a este tráfego. Este “Cliente com tráfego diferenciado” conseguia acessar o sitio.

Simulações de outros tipos de ataques foram executadas mas não provocaram negação de serviço nos servidores pois tais ataques eram tratados no nível de sistema operacional, o qual descartava rapidamente o tráfego inadequado gerado pelas ferramentas utilizadas.

4.7 RESULTADOS

A Figura 29 mostra um comparativo entre os padrões de tráfego de um ataque simulado sem a utilização do mecanismo desenvolvido (a) e outro ataque simulado com a utilização do mecanismo (b).

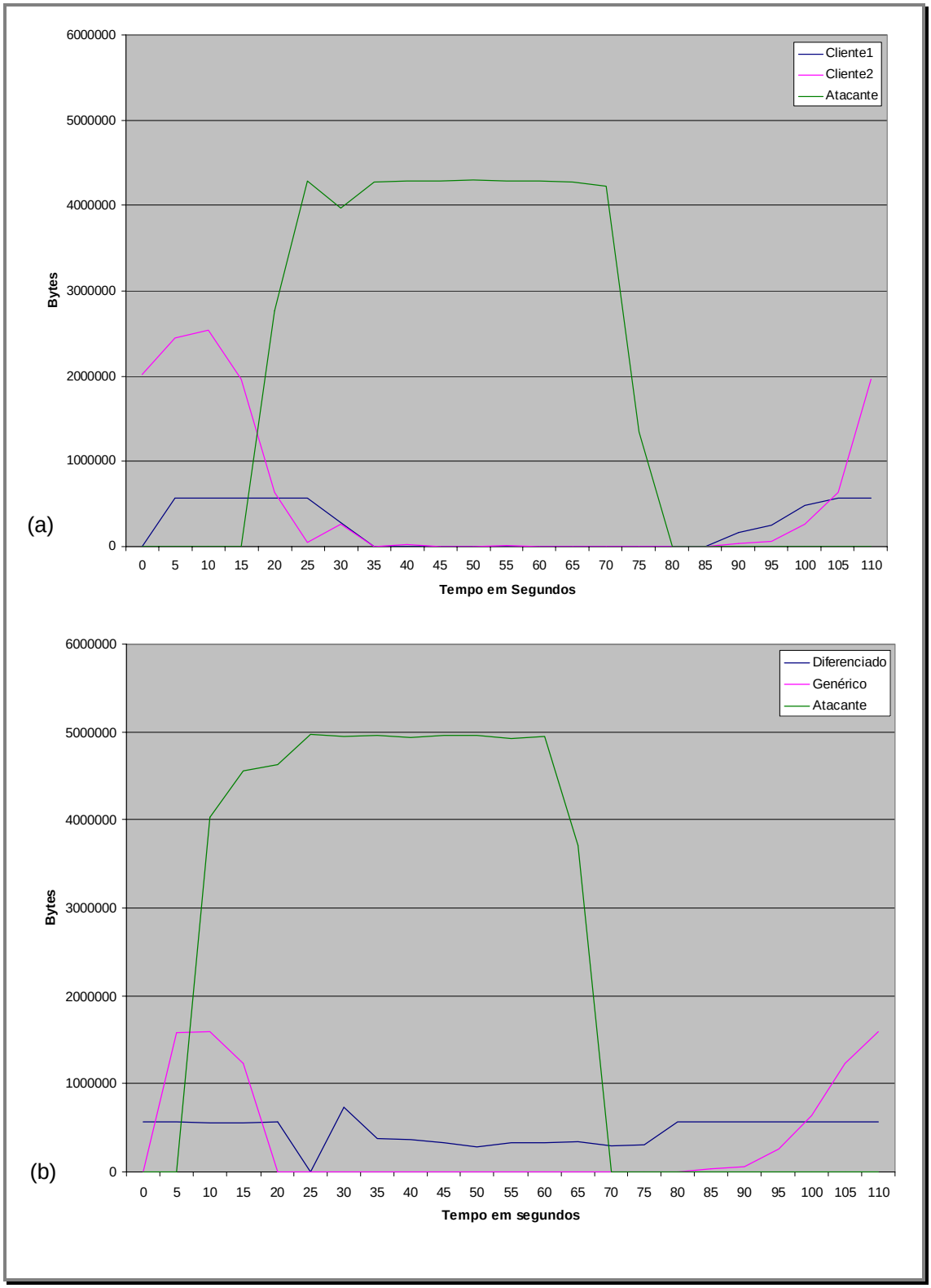

Figura 29 - Tráfego em *bytes* no "Distribuidor de Conexões"

O gráfico da Figura 29(a) representa o tráfego de três fluxos distintos, dois deles simulando “Clientes comuns” ou “Genéricos” do sítio de comércio eletrônico representado e outro simulando um “Atacante”. Observa-se neste gráfico:

- O atacante atinge o limite de consumo dos recursos disponíveis e consegue provocar negação de serviço aos “Clientes comuns” até o momento em que o ataque é finalizado;
- Os “Clientes comuns” demoram um certo tempo (aproximadamente 20 segundos) após a finalização do ataque até recuperar o seu padrão de tráfego e isto acontece em função do tempo necessário para expiração das requisições SYN nos servidores *web* utilizados pelo “Balanceador de Carga”.

Outro gráfico na Figura 29(b) representa o tráfego de fluxos “Cliente com tráfego diferenciado”, o “Cliente comum” ou “Genérico” e o “Atacante”. Percebe-se neste gráfico:

- No momento da ativação do mecanismo, momento este que ocorre próximo ao ponto em que o atacante atinge o limite do consumo dos recursos disponíveis, ocorre diminuição significativa do tráfego gerado pelo “Cliente com tráfego diferenciado”. Provavelmente isso é provocado pela instabilidade causada na ativação do mecanismo, que remove do “Balanceador de carga” o “Servidor *web* destinado a tráfego diferenciado” e introduz as classes de serviços necessárias.
- Este comportamento não é notado quando o mecanismo é desativado automaticamente em função do tráfego não ser mais considerado como um ataque;
- Ocorre negação de serviço ao “Cliente comum” enquanto o fluxo do tráfego do “Atacante” não é finalizado;
- Após a finalização do ataque o “Cliente comum” demora um certo tempo até iniciar a recuperação do seu padrão de tráfego em função do tempo necessário para expiração das requisições de conexão SYN no “Servidor *web* destinado a tráfego genérico”;
- O “Cliente com tráfego diferenciado” enfrenta uma pequena variação no seu padrão de tráfego após o término do ataque até a desativação do “Distribuidor de conexões”, mas este padrão é rapidamente restabelecido uma vez que o “Servidor *web* destinado a tráfego diferenciado” é adicionado novamente o

"Balanceador de carga" e este o escolhe como melhor opção para direcionamento do tráfego deste cliente;

- Com a utilização do mecanismo foi possível manter 70,3% do tráfego originado no "Cliente com tráfego diferenciado" durante o período do ataque.

Testes realizados com a ferramenta Netperf (Hewlett-Packard Company, 1995) para avaliar a vazão (*throughput*) entre os componentes apresentaram os resultados mostrados na Tabela 5.

Tabela 5 – Vazão média estimada no ambiente

Componentes		**Situação**	**Vazão Média TCP em Mbps**
Origem	**Destino**		
Cliente comum	Roteador	Sem tráfego	85,09
Cliente comum	Roteador	Com tráfego de fundo[7]	47,50
Cliente comum	Roteador	Com tráfego de clientes, sob ataque e sem mecanismo	0,29
Cliente com tráfego diferenciado	Roteador	Com tráfego de clientes, sob ataque e com mecanismo	1,26

A implementação do protótipo experimental comprovou a efetividade do modelo proposto por Meylan (2003) em um cenário onde provedores "confiáveis" gerem tráfego com os requisitos necessários de QoS utilizando DiffServ para que o sítio de comércio eletrônico possa diferenciá-lo em casos de ataques de negação de serviço. Também demonstrou a viabilidade de uma solução simples que provê uma disponibilidade seletiva de sítios de Internet utilizando apenas protocolos já existentes.

[7] Tráfego que representa outros fluxos através do "Roteador".

5 CONCLUSÕES, LIMITAÇÕES E TRABALHOS FUTUROS

O objetivo deste capítulo é apresentar as conclusões alcançadas durante o desenvolvimento deste trabalho. Serão relatadas as limitações identificadas nas avaliações do problema de DDoS, nas avaliações dos mecanismos de prevenção e combate e, na implementação do protótipo experimental. O capítulo é finalizado com proposições para trabalhos futuros relativos aos temas abordados.

5.1 CONCLUSÕES

Muitas organizações, as quais avaliam a Internet como um canal estratégico para seus negócios, vêm investindo em infra-estruturas resilientes, baseadas em replicação e distribuição, mas tudo isso não tem sido completamente efetivo. A resiliência buscada só poderá ser obtida com a integração e cooperação entre diversos mecanismos proteção, pois as ameaças são diversas e as estratégias de proteção e combate também.

O problema de DDoS não é simples e as estratégias de prevenção e combate também são diversas e algumas delas bastante complexas. O estudo e desenvolvimento destas estratégias deve ser acompanhado de implementações experimentais ou, simulações baseadas na topologia e modelagem de tráfego que representem o padrão encontrado atualmente na Internet

Por outro lado, para desenvolver aplicativos para a Internet não basta levar em consideração apenas requisitos de negócio. Também não basta alocar recursos de alta disponibilidade se as aplicações suportadas por esta infra-estrutura não estiverem preparadas para tratar eventos que podem resultar em negação de serviço.

Somente um esforço conjunto e cooperativo parece ser a melhor estratégia para combater estes problemas. E este esforço deve envolver a caracterização e assunção de responsabilidades por parte de todos os interessados, quais sejam, as organizações que mantém os serviços de infra-estrutura básica da Internet (raiz DNS, atribuição de endereçamento IP, registros de nomes de domínios, etc.), hospedeiros de sítios, provedores de serviços de Internet, fabricantes, governos, entre outros.

O problema abordado em certo nível, poderia ter então sua magnitude reduzida se responsabilidades fossem assumidas. Mas, de qualquer forma, os mecanismos abordados representam esforços no sentido de combatê-lo tecnologicamente.

Mesmo lançando mão de recursos tecnológicos também é necessária a assunção de responsabilidades e neste sentido o mecanismo DWARD (Mirkovic, 2003) é um ótimo exemplo, por considerar que a origem de ataques DDoS deve se responsabilizar em combatê-lo na raiz. Infelizmente, ainda não existem leis em todos os países para exigir a adoção de mecanismos deste tipo por todos os administradores de redes conectadas à Internet. Sendo assim, as potenciais vítimas e seus parceiros poderiam adotar um mecanismo como o *pushback* (Mahajan, 2001) para reduzir os efeitos de possíveis ataques de DDoS, se dispostos a adotar os mesmos critérios para controle de agrupamentos.

A abordagem proposta por Brustoloni (2002) encontra resultados semelhantes com o mecanismo desenvolvido e apresentado nesta dissertação, mas a avaliação destes resultados é feita com outro ferramental e com outra metodologia. Além disso, os parâmetros de classes de serviço utilizados são diferentes, pois Brustoloni (2002) utiliza apenas duas classes, enquanto neste trabalho foram implementadas quatro classes de QoS.

Em comparação com a perda de vazão relatada por Brustoloni (2002), a qual foi de 23,6%, a perda média de 29,7% identificada nos resultados deste trabalho pode ser percebida como pior, mas este resultado deve-se às diferenças já relatadas entre as implementações. Estes resultados podem ser considerados aceitáveis quanto ao objetivo de manutenção da disponibilidade seletiva do sítio de comércio eletrônico.

Portanto, o modelo proposto por Meylan (2003) e analisado nesta dissertação em uma implementação experimental, pode ser uma alternativa para aquelas potenciais vítimas de ataques de DDoS que busquem maiores garantias de disponibilidade dos serviços para seus clientes que provêem de provedores “confiáveis”.

5.2 LIMITAÇÕES IDENTIFICADAS

Alguns mecanismos de prevenção e combate avaliados poderiam ter sido mais bem entendidos e comparados se houvesse a disponibilidade de protótipos ou código experimental. O único mecanismo que se obteve acesso foi o DWARD (Mirkovic, 2003), mas acompanhado de pouca documentação para viabilizar sua implementação e teste.

O acesso a laboratórios constituídos por uma grande quantidade de nós poderia ter possibilitado a simulação de ataques de DDoS de larga escala construídos com ferramentas automatizadas para melhor avaliar o comportamento do protótipo implementado e avaliado.

Outras tecnologias de QoS, como o Intserv (*Integrated Services* – Serviços Integrados) (Grupo de Trabalho IETF, 2000) por exemplo poderiam ter sido avaliadas na implementação do modelo proposto por Meylan (2003), o que exigiria que fossem contemplados componentes e recursos adicionais.

As avaliações poderiam ter considerado outros serviços e protocolos também bastante importantes e utilizados na Internet, tais como DNS, SMTP (*Simple Mail Transfer Protocol*), FTP, entre outros.

As instituições financeiras e alguns órgãos governamentais não costumam divulgar os incidentes de segurança que por ventura tenham afetado seus sítios de Internet. Essa prática é adotada para não atrair atenção aos seus sítios que por ventura estejam vulneráveis e também para não prejudicar a imagem destas organizações. Isso dificulta sobremaneira as avaliações mais realistas quanto ao panorama dos ataques de DDoS. Este trabalho baseou-se nos acontecimentos mais notórios para evidenciar a problemática atual de ataques de DDoS, mas é possível que ela seja muito mais preocupante.

5.3 TRABALHOS FUTUROS

Neste trabalho foi implementado o modelo proposto por Meylan (2003) utilizando um componente “Distribuidor de conexões” baseado em iptables. Uma outra possibilidade seria implementar a distribuição de conexões baseada em diferenciação de tráfego em um balanceador de carga, o que poderia ser viável adicionando características de tratamento de pacotes com marcação DSCP no LVS (Zhang, 2000). A integração com um IDS baseado em assinaturas de ataque como

o Snort (Roesch & Green, 2003) também seria possível, mas o mais adequado seria a utilização de sistemas que detectem anomalias de tráfego.

Para garantir que o mecanismo não seja explorado indevidamente por atacantes, por exemplo, os quais gerem tráfego com endereço de origem forjado (como se fosse originado em um provedor “confiável”) e com o PHB AF esperado pelo(s) servidor(es) destinado(s) a tráfego diferenciado, mecanismos de autenticação ou validação do tráfego recebido com tais características devem ser desenvolvidos.

Neste sentido, um recurso bastante simples para validação poderia ser agregado, como o proposto por Jin, Wang & Shin (2003), o qual utiliza a contagem de saltos entre origem e destino para validar o tráfego. Este seria um recurso transparente para o usuário e poderia, de maneira simples, evitar explorações do mecanismo.

Espera-se ser possível avaliar em ambiente de larga escala o modelo proposto por Meylan (2003) trabalhando cooperativamente com o mecanismo DWARD (Mirkovic, 2003).

REFERÊNCIAS

ALMESBERGER, W.; SALIM, J.; KUZNETSOV, A.; **Differentiated Services on Linux**, 2001. Disponível em: http://diffserv.sourceforge.net/. [24/10/2004].

AKAMAI. **Press Release - Akamai Unveils EdgeSuite for Business Continuity**, dez. 2001. Disponível em: http://www.akamai.com/en/html/about/press/press319.html. [24/10/2004].

AUERBACH, K. **Protecting the Internet's Domain Name System**, out. 2001. Disponível em: http://www.cavebear.com/rw/steps-to-protect-dns.htm. [24/10/2004].

BALLIACHE, L. **Differentiated Service on Linux HOWTO.** Agosto, 2003. Disponível em: http://opalsoft.net/qos/DS.htm. [24/10/2004].

BARANOWSKI, S. **How Secure are the Root DNS Servers?** Global Information Assurance Certification, Security Essentials Practical. SANS Institute. Março, 2003. Disponível em: http://www.giac.org/practical/GSEC/Susan_Baranowski_GSEC.pdf. [24/10/2004].

BERNSTEIN, D. **SYN Cookies** Disponível em: http://cr.yp.to/syncookies.html [24/10/2004].

BLAKE, S.; *et al.* An Architecture for Differentiated Services, RFC 2475, dez. 1998. Disponível em: http://www.ietf.org/rfc/rfc2475.txt. [24/10/2004].

BLAZE, M.; *et al.* The KeyNote Trust-Management System Version 2, RFC 2704, set. 1999. Disponível em: http://www.ietf.org/rfc/rfc2704.txt. [24/10/2004].

BRUSTOLONI, J. Protecting Electronic Commerce From Distributed Denial-of-Service Attacks In: INTERNATIONAL WORLD WIDE WEB CONFERENCE (WWW2002), 11., 2002, Honolulu. **Proceedings...,** Association for Computing Machinery, p. 553-561, maio, 2002. Disponível em: http://www2002.org/CDROM/refereed/528/. [24/10/2004].

CERT Coordination Center. CERT Advisory CA-1996-01 UDP Port Denial-of-Service Attack. 8 fev. 1996. Disponível em: http://www.cert.org/advisories/CA-1996-01.html. [24/10/2004].

CERT Coordination Center. CERT Advisory CA-1996-21 TCP SYN Flooding and IP Spoofing Attacks. 19 set. 1996. Disponível em: http://www.cert.org/advisories/CA-1996-21.html. [24/10/2004].

CERT Coordination Center. CERT Advisory CA-1997-28 IP Denial-of-Service Attacks. 16 dez. 1997. Disponível em: http://www.cert.org/advisories/CA-1997-28.html. [24/10/2004].

CERT Coordination Center. CERT Incident Note IN-99-07. Distributed Denial of Service Tools. 18 nov. 1999. Disponível em: http://www.cert.org/incident_notes/IN-99-07.html. [24/10/2004].

CERT Coordination Center. CERT Advisory CA-1999-17 Denial-of-Service Tools. 28 dez. 1999. Disponível em: http://www.cert.org/advisories/CA-1999-17.html. [24/10/2004].

CERT Coordination Center. Results of the Distributed-Systems Intruder Tools Workshop. Pittsburgh, Pennsylvania USA. 2-4 nov. 1999. Disponível em: http://www.cert.org/reports/dsit_workshop-final.html. [24/10/2004].

CERT Coordination Center. CERT Incident Note IN-2000-05 "mstream" Distributed Denial of Service Tool. 2 maio, 2000. Disponível em: http://www.cert.org/incident_notes/IN-2000-05.html. [24/10/2004].

CHEN, Y.; *et al.* Quantifying Network Denial of Service: A Location Service Case Study, agosto, 2001. **Lecture Notes in Computer Science**. v. 2229. ftp://sunsite.berkeley.edu/pub/techreps/CSD-01-1150.html. [24/10/2004].

CISCO SYSTEMS **Configuring TCP Intercept (Prevent Denial-of-Service Attacks)** Disponível em: http://www.cisco.com/univercd/cc/td/doc/product/software/ios113ed/113ed_cr/secur_c/scprt3/scdenial.htm#xtocid2548110. [24/10/2004].

COOK, D.; *et al.* In: IEEE INTERNATIONAL CONFERENCE ON NETWORK, 11., 2003, Sydney. **Proceedings...**, pp. 450-460, set./out. 2003. Disponível em: http://nsl.cs.columbia.edu/projects/sos/papers/websos-icon.pdf. [24/10/2004].

CORSAIRE; VRIES, S. Application of Denial of Service Attacks. **informIT** 30, jul. 2004. Disponível em: http://www.informit.com/articles/printerfriendly.asp?p=175930. [24/10/2004].

DALNET IRC NETWORK. **Just What Is a Botnet?** Jan. 2003. Disponível em: http://zine.dal.net/previousissues/issue22/botnet.php. [24/10/2004].

DITTRICH, D. **The DoS Project's "trinoo" Distributed Denial of Service Attack Tool**. University of Washington. 21 out. 1999. Disponível em: http://staff.washington.edu/dittrich/misc/trinoo.analysis. [24/10/2004].

DITTRICH, D. **The "Tribe Flood Network" Distributed Denial of Service Attack Tool**. University of Washington. 21 out. 1999. Disponível em: http://staff.washington.edu/dittrich/misc/tfn.analysis. [24/10/2004].

DITTRICH, D. **The "Stacheldraht" Distributed Denial of Service Attack Tool**. University of Washington. 31 dez. 1999. Disponível em: http://staff.washington.edu/dittrich/misc/stacheldraht.analysis. [24/10/2004].

Fedora Project. Disponível em: http://fedora.redhat.com. [24/10/2004].

FERGUSON, P.; SENIE, D. Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing, RFC 2827, Maio, 2000. Disponível em: http://www.ietf.org/rfc/rfc2827.txt. [24/10/2004].

GRUPO DE TRABALHO IETF Integrated Services (intserv) Charter. set. 2000. Disponível em: http://www.ietf.org/html.charters/intserv-charter.html. [24/10/2004].

HAINING, W.; ZHANG D.; SHIN, K. Detecting SYN Flooding Attacks. In: Annual Joint Conference of the IEEE Computer and Communications Societies, 21., 2002, New York. **Proceedings...**, jun. 2002. Disponível em: http://www.cs.wm.edu/~hnw/paper/attack.pdf. [24/10/2004].

HEINANEN, J.; BAKER, F.; WEISS, W. E WROCLAWSKI, J. Assured Forwarding PHB Group, RFC 2597. jun. 1999. Disponível em: http://www.ietf.org/rfc/rfc2597.txt. [24/10/2004].

HEWLETT-PACKARD COMPANY; **A Network Performance Benchmark.** Information Networks Division, Hewlett-Packard Company. 15 fev. 1995. Disponível em: http://www.netperf.org/netperf/training/Netperf.html. [24/10/2004].

HILL, K. Microsoft Attack, Mimail Worm Leave Industry Waiting for Other Shoe. **www.NewsFactor.com.** 4, agosto, 2003. Disponível em: http://www.newsfactor.com/story.xhtml?story_id=22025. [24/10/2004].

HOGGAN, D. **The Internet Book: Introdution and Reference**. 2000. Disponível em: http://www.camtp.uni-mb.si/books/Internet-Book [24/10/2004].

HUBERT, B. *et al.* **Linux Advanced Routing & Traffic Control HOWTO**, Disponível em: http://lartc.org/howto/. [24/10/2004].

HUOVINEN, L.; HURSTI, J. **Denial of Service Attacks: Teardrop and Land.** 1998. Disponível em: http://www.hut.fi/~lhuovine/study/hacker98/dos.html. [24/10/2004].

HUSSAIN, A.; HEIDEMANN, J.; PAPADOPOULUS, C. A Framework for Classifying Denial of Service Attacks. In: ACM SIGCOMM 2003 Conference on computer communications, 2003, Karlruhe. **Proceedings...**, agosto, 2003. Disponível em: http://www.isi.edu/~hussain/pubs/Hussain03b.pdf. [24/10/2004].

IOANNIDIS, J.; BELLOVIN, S. Implementing Pushback: Router-Based Defense Against {DDoS} Attacks. In: Network and Distributed System Security Symposium, Catamaran Resort Hotel, San Diego, California. **Proceedings...**, 6-8 fev. 2002. Disponível em: http://citeseer.nj.nec.com/ioannidis02implementing.html. [24/10/2004].

JACOBSON, V; NICHOLS, K.; PODURI, K. **An Expedited Forwarding PHB** – RFC 2598, Junho 1999. Disponível em: http://www.ietf.org/rfc/rfc2598.txt. [24/10/2004].

JIN, C.; WANG, H. e SHIN, K. Hop-Count Filtering: An Effective Defense Against Spoofed DdoS Traffic. In: Conference on Computer and Communications Security. 10., 2003, Washington. **Proceedings...**, New York: ACM Press, 2003. p. 30-41, out. 2003. Disponível em: http://portal.acm.org/citation.cfm?id=948116. [24/10/2004].

KAMIENSKI, C. A. e SADOK, D. **Qualidade de Serviço na Internet**. In: JORNADA DE ATUALIZAÇÃO EM INFORMÁTICA. SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. CONGRESSO NACIONAL 2000, XIX, 2000, Curitiba. **Anais...**, Curitiba: Editora Universitária Champagnat, 2000. v. 2, p. 123-162.

KENNEY, M. **It's the Ping o' Death Page!: How to crash your operating system!** 1997. Disponível em: http://www.ovb.ch/?http://www.ovb.ch/Ping/pod.html. [24/10/2004].

KEROMYTIS, A., MISRA, V. e RUBENSTEIN, D. **Secure Overlay Services** In: 2002 Conference on Applications, technologies, architectures, and protocols for computer communications. 2002. Pittsburgh, Pennsylvania. **Proceedings...**, New York: ACM Press, 2002. vol. 32, no. 4, pp. 61 - 72, out. 2002. Disponível em: http://www.cs.columbia.edu/~angelos/Papers/sos.pdf. [24/10/2004].

KNOWLES, D. **Symantec Security Response W32.SQLExp.Worm** 4 fev. 2003. Disponível em:

http://securityresponse.symantec.com/avcenter/venc/data/pf/w32.sqlexp.worm.html. [24/10/2004].

KONG, J. *et al.* Random Flow Network Modeling and Simulations for DDoS Attack Mitigation. In: IEEE International Conference on Communications(ICC´03), 38., 2003, Anchorage. **Proceedings...,** p. 487-491, 2003. Disponível em: http://www.cs.ucla.edu/NRL/wireless/uploads/ICC03-jkong.pdf.pdf. [24/10/2004].

LEMOS, R. New vírus infects PCs, whacks SCO. **CNET News.com,** 26 jan. 2004. Disponível em: http://news.com.com/2100-7349-5147605.html. [24/10/2004].

LI, J.; *et al.* **SAVE: Source Address Validity Enforcement Protocol.** UCLA Technical Report, 2001. Disponível em: http://citeseer.nj.nec.com/li01save.html. [24/10/2004].

Linux Kernel. Disponível em: http://www.kernel.org. [24/10/2004].

MAHAJAN, R.; *et al.* **Controlling High Bandwidth Aggregates in the Network**. Fev. 2001. Disponível em: http://citeseer.ist.psu.edu/546004.html. [24/10/2004].

MARSAN, C. DDoS Attack Highlights ´Net problems **NetorkWorldFusion** Network World, 28, out. 2002. Disponível em: http://www.nwfusion.com/news/2002/1028ddos.html. [24/10/2004].

MCAFEE Trojan name: IRC/Flood. 28, dez. 2000. Disponível em: http://vil.nai.com/vil/content/v_98936.htm. [24/10/2004].

MCCLURE, S.; SCAMBRAY, J.; KURTZ, G. **Hacking Exposed:** Network Security Secrets and Solutions. Berkeley: Osborne/McGraw-Hill, 1999.

MCGUIRE, D.; KREBS, B. Attack On Internet Called Largest Ever **washingtonpost.com** Whashington, 22, out. 2002. http://www.washingtonpost.com/ac2/wp-dyn?pagename=article&contentId=A828-2002Oct22&notFound=true. [24/10/2004].

MCGUIRE, D.; KREBS, B. More Than One Internet Attack Occurred Monday **washingtonpost.com** Whashington, 23, out. 2002. Disponível em: http://www.washingtonpost.com/ac2/wp-dyn?pagename=article&contentId=A6894-2002Oct23&notFound=true. [24/10/2004].

MCHARDY, P. **Linux IMQ - Intermediate queueing device.** Disponível em: http://www.linuximq.net. [24/10/2004].

MEYLAN; F. **CRIPTOQoS: Uma Plataforma de Gerenciamento e Desenvolvimento de Aplicações Distribuídas com Suporte Integrado à QoS e Segurança**, 2003. Cap. 4, p. 70-78, Tese (Doutorado em Engenharia), Escola Politécnica, Universidade de São Paulo, São Paulo, 2003.

MICROSOFT CORPORATION. Microsoft Help and Support. Article ID: 238329 **Fragmented IGMP Packet May Promote "Denial of Service" Attack**. 16 de Julho, 2004. Disponível em: http://support.microsoft.com/default.aspx?scid=kb;en-us;238329. [24/10/2004].

MICROSOFT CORPORATION. Microsoft TechNet Archive. **MS Web Application Stress Tool**. Disponível em: http://www.microsoft.com/technet/archive/itsolutions/intranet/downloads/webstres.mspx. [24/10/2004].

MIRKOVIC, J.; PRIER, G.; REIHER, P. Attacking DDoS at the Source. In: IEEE INTERNATIONAL CONFERENCE ON NETWORK PROTOCOLS, 10., 2002, Paris. **Proceedings...**, pp. 312-321, nov. 2002. Disponível em: http://www.lasr.cs.ucla.edu/ddos/404_mirkovic_j.pdf. [24/10/2004].

MIRKOVIC, J. **D-WARD: Source-End Defense Against Distributed Denial-of-Service Attacks** 2003. 367 p. Tese (Doctor of Philophy in Computer Science), Computer Science Department, University of California, Los Angeles Disponível em: http://www.lasr.cs.ucla.edu/ddos/dward-thesis.pdf. [24/10/2004].

MITRE CYBER RESOURCE CENTER TEAM. **Help Defeat Denial of Service Attacks: Step-by-step.** 2000. Disponível em: http://www.mitre.org/tech/cyber/DDOS/. [24/10/2004].

MOORE, D.; VOELKER, G.; SAVAGE, S. Inferring Internet Denial-of-Service Activity In: Usenix Security Symposium 10., 2001, Washington, D.C. **Proceedings...,** agosto, 2001. Disponível em: http://www.caida.org/outreach/papers/2001/BackScatter/usenixsecurity01.pdf. [24/10/2004].

NAKAMURA, E. T. **Um Modelo de Segurança de Redes para Ambientes Cooperativos**. 2000. 286 p. Dissertação (Mestrado em Ciência da Computação), Instituto de Computação, UNICAMP.

NARAINE, R. Massive DDoS Attack Hit DNS Root Servers. **Internetnews.com** 23 out. 2002. Disponível em: http://www.internetnews.com/dev-news/article.php/1486981. [24/10/2004].

NETCRAFT. DDoS Attack on DoubleClick Slows Many Sites. 28 jul. 2004. Disponível em: news.netcraft.com/archives/2004/07/28/ddos_attack_on_doubleclick_slows_many_sites.html. [24/10/2004].

NETCRAFT. Akamai Attack Highlights Threat From Bot Networks. 16 jun. 2004. Disponível em: news.netcraft.com/archives/2004/06/16/akamai_attack_highlights_threat_from_bot_networks.html [24/10/2004].

NICHOLS, K.; *et al.* Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, RFC 2474, dez. 1998. Disponível em: http://www.ietf.org/rfc/rfc2474.txt. [24/10/2004].

NISCC National Infrastructure Security Co-Ordination Centre. NISCC Vulnerability Advisory 236929 20 abr. 2004. Disponível em: http://www.uniras.gov.uk/vuls/2004/236929/index.htm. [24/10/2004].

NUTTALL, C. Sci/Tech Virtual 'Nucked' on Net. **BBC News,** 26 jan. 1999. Disponível em: http://news.bbc.co.uk/1/hi/sci/tech/263169.stm. [24/10/2004].

OIKARINEN, J.; REED, D. Internet Relay Chat Protocol, RFC-1459. Maio, 1993. Disponível em: http://www.ietf.org/rfc/rfc1459.txt. [24/10/2004].

PAPPALARDO, D. ISPs take on DDoS attacks. **Computerworld.** 19 nov. 2003. Disponível em: http://www.computerworld.com/securitytopics/security/story/0,10801,87343,00.html. [24/10/2004].

PAXSON, V. **An Analysis of Using Reflectors for Distributed Denial-of-Service Attacks** ACM Computer Communications Review (CCR). jul. 2001 v. 31. Disponível em: http://citeseer.nj.nec.com/paxson01analysis.html [24/10/2004].

PELLINE, J. MyDoom downs SCO site. **CNET News.com,** 2 fev. 2004. Disponível em: http://news.com.com/2100-7349-5151572.html. [24/10/2004].

PFLEEGER, S.; PFLEEGER, C., Program Security. **InformIT**, 30 maio, 2003. Disponível em: http://www.informit.com/articles/article.asp?p=31782. [24/10/2004].

POULEN, K. FBI busts alleged DDoS Mafia. **SecurityFocus.** 26 agosto, 2004. Disponível em: http://www.securityfocus.com/news/9411. [24/10/2004].

PTACEK, T.; NEWSHAM, T. **Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection,** jan. 1998. Disponível em: http://securityfocus.com/data/library/ids.ps. [24/10/2004].

RIVERHEAD NETWORKS. Disponível em: http://www.riverhead.com. [24/10/2004].

ROESCH, M.; GREEN, C. **Snort Users Manual – Snort Release: 2.2.0**. 2003. Disponível em: http://www.snort.org/docs/snort_manual/. [24/10/2004].

RUSSEL, R.; WELTE, H. **Linux Netfilter Hacking HOWTO.** 2 jul. 2002. Disponível em: http://www.netfilter.org. [24/10/2004].

SENIE, D. Changing the Default for Directed Broadcasts in Routers, RFC-2644, agosto, 1999. Disponível em: http://www.ietf.org/rfc/rfc2644.txt. [24/10/2004].

SIMCOCK, T. **Distributed Denial of Service Attacks: Threats, Motivation & Management**. 5 nov. 2002. SANS Institute Disponível em: http://www.giac.org/practical/GSEC/Tom_Simcock_GSEC.pdf. [24/10/2004].

SINGER, A.; **Eight Things that ISP's and Network Managers Can Do to Help Mitigate Distributed Denial of Service Attacks** San Diego Supercomputer Center. fev. 2000. Disponível em: http://security.sdsc.edu/publications/ddos.shtml. [24/10/2004].

SUNG, M.; XU, J. IP Traceback-based Intelligent Packet Filtering: A Novel Technique for Defending Against Internet DDoS Attacks. In: IEEE INTERNATIONAL CONFERENCE ON NETWORK PROTOCOLS, 10., 2002, Paris. **Proceedings...**, nov. 2002. Disponível em: http://www.ieee-icnp.org/2002/papers/2002-27.pdf. [24/10/2004].

SWARTZ, J. Online betting sites fight cyberextotion. **USA TODAY,** 9 mar. 2004. Disponível em: http://www.usatoday.com/money/industries/technology/2004-03-09-cyberextort_x.htm. [24/10/2004].

Synful.c, Disponível em: http://www.buha.info/files/user/html/id_Sources_synful.c.html. [24/10/2004].

TECHWEB NEWS. AT&T Adds Weapon Against Denial Of Service Attacks. **TechWeb,** 2 jun. 2004. Disponível em: http://www.techweb.com/wire/story/TWB20040602S0004. [24/10/2004].

THE APACHE SOFTWARE FOUNDATION. **Flood - a profile-driven HTTP load tester.** Disponível em: http://httpd.apache.org/test/flood/. [24/10/2004].

VIRUSLIST.COM. **25th August 2004: Who knows what tomorrow will bring?** 25 agosto, 2004. Disponível em: http://www.viruslist.com/eng/index.html?tnews=461517&id=2100900. [24/10/2004].

VOLOBUEV, Y. **ARP and ICMP redirection games.** 19, set. 1997. Disponível em: http://www.insecure.org/sploits/arp.games.html. [24/10/2004].

WATSON, P. **Slipping In The Window: TCP Reset Attacks.** CanSecWest 2004. Disponível em: http://cansecwest.com/csw04/csw04-Watson.doc. [24/10/2004].

WEARDEN, G. **Denial-of-service extortion threat for online bookies.** ZDnet UK 24 fev. 2004. Disponível em: http://www.silicon.com/networks/webwatch/0,39024667,39118605,00.htm. [24/10/2004].

WHEELER, D. **Secure Programming for Linux and Unix HOWTO** v3.010, 3 mar. 2003. Disponível em: http://www.dwheeler.com/secure-programs/Secure-Programs-HOWTO.pdf. [24/10/2004].

ZEUGE, K.; ROLLO, T.; MESANDER, B. **The Client-To-Client Protocol (CTCP).** agosto, 1994. Disponível em: http://www.irchelp.org/irchelp/rfc/ctcpspec.html. [24/10/2004].

ZHANG, W. Linux Virtual Server for Scalable Network Services. In: OTTAWA LINUX SYMPOSIUM. 2., 2000. Ottawa. **Proceedings...** Disponível em: http://www.linuxvirtualserver.org/ols/lvs.ps.gz. [24/10/2004].

ZWICKY, E.; COOPER, S.; CHAPMAN, D. **Building Internet Firewalls** Sebastopol: O´Reilly & Associates, Inc., 2000.