PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

CARLOS RAFAEL GUERBER

ARVDoS: UMA ARQUITETURA REATIVA A ATAQUES DoS PARA REDES VIRTUAIS

CURITIBA
2010

CARLOS RAFAEL GUERBER

ARVDoS: UMA ARQUITETURA REATIVA A ATAQUES DoS PARA REDES VIRTUAIS

Dissertação apresentada ao Programa de Pós-Graduação em Informática Aplicada da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Informática Aplicada.

Área de Concentração: *Redes de Computadores e de Telecomunicações.*

Orientador: Prof. Dr. Mauro Sérgio Pereira Fonseca.

CURITIBA

2010

Dados da Catalogação na Publicação
Pontifícia Universidade Católica do Paraná
Sistema Integrado de Bibliotecas – SIBI/PUCPR
Biblioteca Central

G929a
2010

Guerber, Carlos Rafael
ARVDoS : uma arquitetura reativa a ataques DoS para redes virtuais / Carlos Rafael Guerber ; orientador, Mauro Sérgio Pereira Fonseca. – 2010.
106 f. : il. ; 30 cm

Dissertação (mestrado) – Pontifícia Universidade Católica do Paraná, Curitiba, 2010
Bibliografia: f. 101-106

1. Arquitetura de rede de computador. 2. Sistemas de computador. 3. Redes de computação. I. Fonseca, Mauro Sérgio Pereira. II. Pontifícia Universidade Católica do Paraná. Programa de Pós-Graduação em Informática aplicada. III. Título.

CDD 22. ed. – 004.678

Pontifícia Universidade Católica do Paraná

ATA DE DEFESA DE DISSERTAÇÃO DE MESTRADO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

DEFESA DE DISSERTAÇÃO Nº 11/2010

Aos 04 dias do mês de Novembro de 2010 realizou-se a sessão pública de Defesa da Dissertação **"ARVDoS: Uma Arquitetura Reativa a Ataques DoS para Redes Virtuais."** apresentada pela aluno **Carlos Rafael Guerber** como requisito parcial para a obtenção do título de Mestre em Informática, perante uma Banca Examinadora composta pelos seguintes membros:

Prof. Dr. Mauro Sérgio Pereira Fonseca PUCPR (Orientador)	(assinatura)	Aprovado (aprov/reprov.)
Prof. Dr. Manoel Camillo de Oliveira Penna Neto PUCPR		APROVADO
Prof. Dr. Marcelo Pellenz PUCPR		APROVADO
Profa. Dra. Michele Nogueira UFPR	Michele N. Lima	APROV.

Conforme as normas regimentais do PPGIa e da PUCPR, o trabalho apresentado foi considerado Aprovado (*aprovado/reprovado*), segundo avaliação da maioria dos membros desta Banca Examinadora. Este resultado está condicionado ao cumprimento integral das solicitações da Banca Examinadora registradas no Livro de Defesas do programa.

Prof. Dr. Mauro Sérgio Pereira Fonseca
Diretor do Programa de Pós-Graduação em Informática

Programa de Pós-Graduação em Informática
PPGIa
PUCPR

AGRADECIMENTOS

Ao professor orientador Dr. Mauro Sérgio Pereira Fonseca pela orientação, especialmente pela paciência durante todos os processos de mestrado, mais especificamente à orientação e desenvolvimento da dissertação. Meu muito obrigado.

Ao amigo e administrador de redes Robert Wagner pelo apoio técnico e incentivo para conclusão deste trabalho.

À Universidade do Contestado – UnC/Mafra-SC que viabilizou o curso de mestrado e apoiou meu trabalho até aqui, assim como pela paciência em todo este processo.

Ao carinho e compreensão dos familiares, mãe e irmãs pelo constante apoio e incentivo nos momentos difíceis de angústia e ansiedade que envolveram este trabalho. Obrigado por nunca me deixarem desistir.

A minha namorada Alesandra que acima de tudo acredita em mim, muitas vezes mais que eu mesmo e me faz seguir em frente. Obrigado.

À Força.

"May the force be with you!"

Master Yoda.

RESUMO

A virtualização de redes é proposta como um paradigma que permite a múltiplas redes virtuais (VNs) coexistirem sobre um recurso físico compartilhado. Assim como no modelo físico, estas redes apresentam problemas de segurança e gerência de recursos. Este trabalho propõe uma arquitetura escalar com atividades reativas que possibilite mitigar ataques DoS em ambientes virtuais com a utilização de múltiplos métodos de detecção sem dependência destes métodos ou de tecnologia de virtualização. No trabalho são descritos os agentes que compõe a arquitetura, suas características, atividades e implementações. Da mesma forma um intervalo de confiança é proposto a fim de verificar o desempenho e a regularidade da arquitetura em diferentes cenários.

Palavras chave: Arquitetura, DoS, Virtualização de redes, Defesa reativa.

ABSTRACT

Network virtualization is proposed as a paradigm which allows multiples Virtualized Networks (VNs) coexist over a same shared physical resource. As in the physical models, these networks have management and security issues to discuss. This paper proposes a scalar architecture with reactive features that enables to mitigate DoS attacks in virtual environments with multiple detection methods with no dependency on these methods or virtualization technologies. The work describes the architecture composition agents as its activities implementations and characteristics. Likewise a confidence interval is proposed to verify the architecture performance and regularity in different scenarios.

Key words: DoS, Network virtualization, Reactive defense.

LISTA DE FIGURAS

Figura 2.1 Recurso de virtualização a partir do perímetro da VN. 29
Figura 2.2 Rede virtual – NV (Adaptado) 32
Figura 2.3 Ambiente de virtualização de redes (NVE). 34
Figura 5.1. Pré-condições de implementação para ARVDoS 52
Figura 5.2. Relacionamento entre as bases de parametrização da ARVDoS..... 53
Figura 5.3. Atividades dos agentes para IDoS..... 58
Figura 5.4. Atividades dos agentes para CDoS 59
Figura 5.5. Atividades dos agentes para NDoS 59
Figura 5.6. Classificação das Reações da ARVDos 60
Figura 5.7. Escalabilidade da ARVDoS 61
Figura 5.8. Visão geral da arquitetura ARVDoS..... 62
Figura 6.1. Cenário I de simulação InP e SP da rede virtual..... 66
Figura 6.2. Base de políticas para o Cenário I..... 67
Figura 6.3. Topologia após a reação da ARVDoS para IDoS para o Cenário I..... 70
Figura 6.4. Comportamento do enlace na topologia para o Cenário I..... 72
Figura 6.5. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário I 73
Figura 6.6. Base de políticas para o Cenário II 74
Figura 6.7. Cenário II de simulação InP e SP da rede virtual..... 75
Figura 6.8. Topologia após a reação da ARVDoS para IDoS no Cenário IIa..... 76
Figura 6.9. Comportamento do enlace na topologia para o Cenário IIa..... 78
Figura 6.10. Comportamento do enlace na topologia para o Cenário IIb..... 81
Figura 6.11. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário IIa 82
Figura 6.12. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário IIb 83
Figura 6.13. Cenário III de simulação InP e SP da rede virtual..... 84
Figura 6.14. Base de políticas para o Cenário III 85
Figura 6.15. Topologia após a reação da ARVDoS para IDoS para Cenário III..... 88
Figura 6.16. Comportamento do enlace na topologia para o Cenário III..... 91
Figura 6.17. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário III 92

Figura 6.18. Base de políticas para o Cenário IV 93
Figura 6.19. Topologia após a reação da ARVDoS para CDoS para o Cenário IV... 94
Figura 6.20. Comportamento do enlace na topologia para o Cenário IV 96
Figura 6.21. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário IV 97

LISTA DE SIGLAS

ADS	*Attack Detection Strategy*, Estratégia de Detecção de Ataque.
AP	*Access Provider,* Provedor de Acesso.
ARP	*Address Resolution Protocol,* Protocolo de Resolução de Endereço.
ARS	*Attack Response Strategy*, Estratégia de Resposta ao Ataque.
CEP	Controle Estatístico de Processos.
CDoS	Certeza de DoS
CSUM	*Cumulative Sum*, Soma Cumulativa.
DDOS	*Distributed Denial of Service*, Negação de Serviço Distribuída.
DLPI	*Data Enlace Provider Interface,* Inteface Provedora de Enlace de Dados.
DMA	*Direct Memory Access*, Acesso Direto a Memória.
DNS	*Domain Name System*, Sistema de Nomes e Domínios.
DoS	*Denial of Service*, Negação de Serviço.
e.g.	*Exempli Gratia*, Por Exemplo.
et. al.	*Et Alii*, Entre Outros.
HTTP	*Hyper Text Transfer Protocol*, Protocolo de Transferência de Hipertexto.
ICMP	*Internet Control Message Protocol*, Protocolo de Controle de Mensagens de Internet.
IEEE	*Institute of Electrical and Electronics Engineers,* Instituto de Engenheiros Eletricistas e Eletrônicos.
InP	*Infrastructure Provider*, Provedor de Infraestrutura.
IP	*Internet Protocol*, Protocolo de Internet.
IPSec	*IP Security Protocol,* Protocolo de Segurança IP.
ISP	*Internet Server Provider*, Provedor de Serviço de Internet.
LAN	*Local Area Network*, Rede de Abrangência Local.
MAC	*Medium Access Control*, Controle de Acesso ao Meio.
MAN	*Metropolitan Area Network*, Rede de Abrangência Metropolitana.
MTU	*Maximum Transfer Unit,* Unidade Máxima de Transferência.
NDoS	Não DoS.
NVE	*Network Virtualized Environment*, Ambiente Virtualizado de Rede.

PPVPN	*Provider Provisioned Virtual Private Network*, Provedor de Serviço de Redes Privadas Virtuais.
QoS	*Quality of Service*, Qualidade de Serviço.
RVA	Roteador Virtual da Arquitetura
SNMP	*Simple Network Management Protocol*, Protocolo de Simples Gerenciamento de Rede.
SP	*Service Provider*, Provedor de Serviço.
SYN	*Syncronism Bit*, Bit de Sincronismo.
TCP	*Transfer Control Protocol*, Protocolo de Controle de Transferência.
UDP	*User Datagram Protocol*, Protocolo de Datagrama de Usuário.
IDoS	Incerteza de DoS.
VLAN	*Virtual Local Area Network*, LAN Virtual.
VLAN ID	*Virtual Local Area Network Identificator*, Identificador de Rede de Abrangência Local Virtual.
VN	*Virtual Network*, Rede Virtual.
VNIC	*Virtual Network Interface Card*, Interface de Rede Virtual.
VPN	*Virtual Private Network*, Rede Provada Virtual.
WAN	*Wide Area Network*, Rede de Ampla Abrangência.

LISTA DE TERMOS

ACK	Bit de sincronismo que compõe o protocolo de transporte TCP, aplicado no estabelecimento de conexões entre dois computadores em uma rede.
AGENTES	São computadores que efetivamente realizam o ataque de DoS comandados por uma máquina atacante (mestre).
BIT	*Binary Digit, m*enor unidade de dado que um sistema computacional pode tratar, reconhece os valores 0 (zero) e 1 (um).
CABEÇALHO	Em um protocolo de rede é a parte que contém as informações suplementares colocados no começo de um bloco de dados que está sendo armazenado ou transmitido.
CONFORMES	Em concordância ou de acordo com os critérios estabelecidos para verificação de status em um instante de tempo.
DEFAULT GATEWAY	Um Gateway padrão é o nó na rede de computadores que é escolhida quando o endereço IP não corresponde a nenhuma outra rota na tabela de roteamento ou repasse.
DESVIO PADRÃO	Unidade Estatística da medida da dispersão em torno da média aritmética de um conjunto de dados.
DOMÍNIO DE BROADCAST	Segmentação lógica de uma rede no qual as estações recebem mensagens de broadcast.
ETHERNET	Padrão IEEE 802.3 que define o protocolo e o método de sinalização para uma LAN.
FILA DE CONEXÃO	Conexão estabelecida entre um par de computadores em uma rede de dados na qual forma-se uma fila para transmissão de dados com verificação por bits de reconhecimento e sincronismo que determina a sequência da transmissão de dados.
FIM A FIM	Comunicação estabelecida entre dois computadores em uma rede e garantida pela camada de transporte.
FIREWALLS	Combinação de hardware e software cujo papel é o de filtrar o trânsito de informações entre redes fechadas. Usa sistemas de

	monitoração que analisa tudo o que entra e sai do servidor e outros protocolos de segurança.
FLAG	Mecanismo lógico utilizado para indicar o estado de um objeto em um algoritmo ou protocolo a fim de determinar se tal objeto está ligado ou desligado.
FLOODING	Inundação de dados em uma rede de computadores. Utilizado para designar tipos de ataques por protocolo.
FORÇA BRUTA	Ou, busca exaustiva, que consiste em buscar todas as possíveis soluções a fim de satisfazer uma situação. Sempre encontrará uma solução se esta existir, porém a um elevado custo computacional.
GATEWAY	É uma máquina intermediária geralmente destinada a interligar redes, separar domínios de broadcast, ou mesmo traduzir protocolos.
HIPERMÍDIA	Programa que contém ligação dinâmica com outras mídias, como rádio, vídeo e arquivos gráficos, entre outros interligados por meio da web.
HOP	É o trajeto de um pacote de dados partindo de um roteador ou ponto intermediário para outro roteador ou ponto de rede.
ICMP ECHO	Requisição realizada pelo protocolo ICMP por uma máquina origem para um verificar conectividade entre dois computadores em uma rede IP.
ICMP ECHO REPLY	Resposta realizada pelo protocolo ICMP por uma máquina destino como resposta a requisição de conectividade entre computadores em uma rede IP.
INUNDAÇÃO UDP	Tipo de ataque de negação de serviço através da semântica do protocolo UDP com a realização de inundações de dados realizados no transporte.
IPSEC	Conjunto de serviços e protocolos de proteção baseados em criptografia padrão do setor.
IP SPOOFING	Técnica de subversão em sistemas de computadores que consiste em mascarar pacotes IP utilizando endereços de remetentes falsificados.

LOOP	Termo utilizado para indicar a repetição de um grupo de comandos em um algoritmo ou programa.
MÁQUINAS ZUMBIS	São computadores que efetivamente realizam o ataque de DoS comandados por uma máquina atacante (mestre).
MÉDIA	O valor médio de uma distribuição, determinado segundo uma regra.
MULTI HOP	Múltiplos saltos. Condição que pode ocorrer no encaminhamento de pacotes da origem para o destino entre dois computadores. Ver *Salto*.
MULTICAST	Uma forma de broadcast no qual um pacote é entregue a um grupo pré-definido de destinos de todos os destinos possíveis.
NÃO-CONFORMES	Em discordância ou não de acordo com os critérios estabelecidos para verificação de status em um instante de tempo.
NUVEM IP	Conjunto de nós com endereçamento IP formando uma rede de grande abrangência em que há diversos pontos de roteamento.
PAYLOAD	Em protocolos de comunicação refere-se ao dado real sendo transmitido acompanhado de um cabeçalho de um protocolo com informações de origem e destino.
PLOTADA	Mapeada, diagramada. Realiza a conexão de pontos a valores coordenados.
PROCESSADOR	Circuito eletrônico que executa as instruções de processamento aritmético, lógico e movimentação de dados definida por um programa (software).
PROCESSO	No contexto da informática trata-se de um programa em execução.
PROCESSO (2)	É um modo de proceder, uma sequência de atos que visam produzir um resultado.
PROTOCOLO	É uma convenção ou padrão que controla e possibilita uma conexão, comunicação ou transferência de dados entre dois sistemas computacionais.
PULSING ATTACK	Tipo de ataque de negação de serviço, o qual envia uma sequência de pulsos de ataque a fim de reduzir o *throughput*.

QUALIDADE DE SERVIÇO	Garantia de entrega de serviço de rede de acordo com as exigências do usuário.
QUEUES	O mesmo que fila. Relacionado à teoria das filas em redes de computadores.
RECURSIVA	Recursivo, processo que para atingir um objetivo, chama a si mesmo, várias vezes, e a cada iteração fornece um resultado parcial que alimentará o turno seguinte do processo.
REDE OVERLAY	Tipo de rede de computador construída sobre a infraestrutura de outra rede.
REFLECTOR ATTACKS	Tipo de ataque DDoS em grande escala.
REQUISIÇÕES DNS	Solicitação de acesso ao serviço de tradução de nomes e endereços em um servidor DNS.
REQUISIÇÕES HTTP	Solicitação de acesso ao serviço de transferência de endereços entre dois computadores em uma rede.
ROTEAMENTO	Trata-se do encaminhamento de pacotes. Designa o processo de reencaminhamento de pacotes entre redes distintas, que se baseia no endereço IP e suas máscaras de rede.
Rx/Tx	Mede a largura de banda de transferência de dados entre o computador e vários servidores.
SEMÂNTICA	Entendimento, compreensão do funcionamento e das características de um objeto para aplicação de técnicas que o permitam manipular de acordo com uma sintaxe.
SERVIDOR WEB	Programa de computador responsável por aceitar pedidos HTTP de clientes, geralmente os navegadores, e servi-los com respostas HTTP.
STREAM	Fluxo de dados em um sistema computacional.
SMURF ATTACK	Tipo de ataque de negação de serviço que gera tráfego de rede significante em uma rede atacada com envio de grande quantidade de ICMP *echo request* em *broadcast* a uma rede IP.
SWITCHES	Dispositivo utilizado em redes de computadores para reencaminhar frames entre os diversos nós e possibilita a redução do domínio de *broadcast* em uma rede.
SYN	Bit de sincronismo que compõe o protocolo de transporte TCP,

	aplicado no estabelecimento de conexões entre dois computadores em uma rede.
TABELA DE ROTEAMENTO	São registros de endereços de destino, ID de rede e máscara, associados ao número de saltos.
TCP/IP	Conjunto de protocolos de comunicação entre computadores em rede.
TCP SYN COOKIES	Técnica utilizada para proteção contra ataques de DoS SYN *Flood*.
TEMPO REAL	Tipo de processamento de resultado imediato, quase simultâneo às ações que o acionam.
THROUGHPUT	Desempenho da transmissão de dados, sendo medido pela quantidade de bits transmitidos ou recebidos durante certo intervalo de tempo. É a taxa real de transmissão de dados em um instante de tempo.
TRACEBACK	Método confiável para a determinação da origem de um pacote em redes de computadores.
VARIÂNCIA	Medida da dispersão estatística que indica quão longe os seus valores se encontram do valor esperado (média).
VARIÁVEL	Atributo mensurável que tipicamente apresenta comportamento variável de valores entre interações de objetos.
VÍTIMA	Em ataques de negação de serviço trata-se da rede, computador ou serviço que esta sobre ataque.

LISTA DE TABELAS

Tabela 1. Métodos de detecção DoS e suas avaliações 49
Tabela 2. Configurações de endereçamento de rede do InP e da VN 66
Tabela 3. Tabela de roteamento do roteador virtual CWB 66
Tabela 4. Tabela de roteamento do roteador virtual SPO 67
Tabela 5. Mensagem de configuração para do *mIndentifier* para *mRemaker* para o Cenário I 68
Tabela 6. Configuração dos VNICs da VN reativa 68
Tabela 7. Tabela de roteamento do RVA 68
Tabela 8. Rotas adicionadas ao RV CWB após reação 68
Tabela 9. Rotas adicionadas ao RV SPO após reação 69
Tabela 10. Valores de configuração de enlace entre CWB e RVA 69
Tabela 11. Valores de configuração de enlace entre RVA e SPO 69
Tabela 12. Comportamento do enlace na topologia para o Cenário I 70
Tabela 13. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário I 72
Tabela 14. Avaliação do tempo de reação em diferentes tecnologias para o Cenário I 73
Tabela 15. Mensagem de configuração para do *mIndentifier* para *mRemaker* para o Cenário II 75
Tabela 16. Valores de configuração de enlace entre CWB e RVA para o Cenário IIa 76
Tabela 17. Valores de configuração de enlace entre RVA e SPO para o Cenário IIb 76
Tabela 18. Comportamento do enlace na topologia para o Cenário IIa 77
Tabela 19. Comportamento do enlace na topologia para o Cenário IIb 79
Tabela 20. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário IIa 82
Tabela 21. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário IIb 82
Tabela 22. Tempo de reação da arquitetura para IDoS para o Cenário IIa e Cenário IIb 83
Tabela 23. Configurações de endereçamento de rede do InP e da VN 84

Tabela 24. Tabela de roteamento do roteador virtual CWB 84
Tabela 25. Tabela de roteamento do roteador virtual MFA 85
Tabela 26. Tabela de roteamento do roteador virtual SPO 85
Tabela 27. Mensagem de configuração para do *mIndentifier* para *mRemaker* para o Cenário III 86
Tabela 28. Configuração dos VNICs da VN reativa 86
Tabela 29. Tabela de roteamento do RVA 87
Tabela 30. Rotas adicionadas ao RV CWB após reação 87
Tabela 31. Rotas adicionadas ao RV MFA após reação 87
Tabela 32. Valores de configuração de enlace entre CWB e RVA 88
Tabela 33. Valores de configuração de enlace entre RVA e MFA 88
Tabela 34. Comportamento do enlace na topologia para o Cenário III 89
Tabela 35. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário III 91
Tabela 36. Tempo de reação da arquitetura para IDoS para o Cenário III 92
Tabela 37. Mensagem de configuração para do mIndentifier para mRemaker para o Cenário IV 94
Tabela 38. Valores de configuração de enlace entre CWB e RVA 94
Tabela 39. Comportamento do enlace na topologia para o Cenário IV 95
Tabela 40. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário IV 96

SUMÁRIO

1. INTRODUÇÃO 22

1.1. MOTIVAÇÃO 23

1.1.1. Dinamismo em um NVE 24

1.1.2. Escalabilidade 25

1.2. ESCOPO 25

1.3. OBJETIVO GERAL 26

1.3.1. Objetivos Específicos 26

1.4. ESTRUTURA DA DISSERTAÇÃO 26

2. VIRTUALIZAÇÃO DE REDES 27

2.1. PERSPECTIVA TECNOLÓGICA 27

2.1.1. Redes Locais Virtuais 27

2.1.2. Redes Privadas Virtuais 27

2.1.3. Redes Overlay 28

2.2. VIRTUALIZAÇÃO DE REDES 29

2.3. ELEMENTOS BÁSICOS DA VIRTUALIZAÇÃO DE REDES 32

2.4. AMBIENTE DE VIRTUALIZAÇÃO DE REDES 33

2.5. REQUISITOS DAS REDES VIRTUALIZADAS 35

3. NEGAÇÃO DE SERVIÇO 37

3.1. ATAQUES DE NEGAÇÃO DE SERVIÇO 37

3.2. TIPOS DE ATAQUES DE NEGAÇÃO DE SERVIÇO 37

3.2.1. Ataques Baseados em Semântica de Protocolos 38

3.2.2. Ataques Baseados em Inundação de Pacotes 38

3.2.3. Ataques Baseados em Possibilidade de Caracterização 39

3.3. MECANISMOS DE DEFESA 41

3.3.1. Preventivo 41

3.3.2. Reativo 42

4. MÉTODOS DE DETECÇÃO 45

4.1. DETECÇÃO DE NEGAÇÃO DE SERVIÇO 45

4.2. TÉCNICAS DE DETECÇÃO 46

4.2.1. Perfil de Atividade 47

4.2.2. Detecção por mudança sequencial de pontos 47

4.2.3 Análise de Wavelets 48

4.2.4 Métodos e Avaliações 49

5. ARVDoS: ARQUITETURA PROPOSTA 51
5.1. DOMÍNIO DE APLICAÇÃO 51
5.2. PRÉ-REQUISITOS 51
5.3. ATIVIDADES 53
5.4 CLASSIFICAÇÃO DAS REAÇÕES DA ARVDoS 60
6. TESTES 64
6.1. VIRTUALIZAÇÃO DE REDE 64
6.1.1. Cenário I 65
6.1.2. Cenário II 73
6.1.3. Cenário III 83
6.1.4. Cenário IV 93
7. CONCLUSÃO E TRABALHOS FUTUROS 98
REFERENCIAS BIBLIOGRÁFICAS 101

1. INTRODUÇÃO

O conceito de virtualização de redes tem recentemente atraído atenção na discussão de como modelar o paradigma da próxima geração de redes Internet para substituir o modelo de Internet atual. A virtualização de redes é vista como uma ferramenta para avaliação de novas arquiteturas, assim como atributo fundamental para as arquiteturas de nova geração (Anderson, et. al, 2005).

A virtualização de redes aplica uma idéia similar à utilizada no atual paradigma de internet para introduzir flexibilidade e separação de políticas (Turner e Taylor 2005) (Feamster, et.al. 2007) dividindo as regras dos tradicionais *Internet Service Providers*, ISPs, ou provedores de serviço de Internet, em duas: *i)* provedores de infraestrutura, *Infrastructure Providers* (InPs) relacionados às redes físicas e seus recursos; *ii)* provedores de serviços, *Service Providers* (SPs) que distribuem as redes virtualizadas, *Virtualized Networks,* VNs customizadas agregando recursos de múltiplos InPs e fornecendo serviços fim a fim aos usuários finais.

Além disso, a virtualização de redes permite a cada uma destas redes físicas e virtuais implementar o controle e o gerenciamento de protocolos de forma heterogênea. Porém tal flexibilidade é acompanhada de alguns ônus em relação ao potencial heterogêneo das redes, em que problemas de desempenho e riscos de segurança na comunicação fim a fim estão presentes nos ambientes virtualizados de rede, *Network Virtualized Environments* (NVE) (Mosharaf, et.al., 2008).

Assim, a implementação de ações de segurança e gerência nestes ambientes se aplica da mesma forma que em ambientes não virtualizados. Os ataques de Negação de Serviço, *Denial of Service,* (DoS), são pertinentes a redes virtuais.

Ataques de negação de serviço apresentam uma ameaça às redes e à Internet. Estes ataques caracterizam-se pelo envio indiscriminado de pacotes e requisições a um alvo, visando degradar a qualidade ou cessar os serviços oferecidos pela vítima.

A prevenção e o rastreamento dos ataques DoS constituem operações de dificuldade considerável. Isso se deve ao grande número de máquinas atacantes envolvidas, ao uso de técnicas para forjar endereços IP, *IP Spoofing*, que escondem a origem verdadeira dos pacotes, e também à similaridade entre o tráfego legítimo e o tráfego de ataque.

A construção de ferramentas efetivas contra ataques DoS representa um desafio aos administradores de redes e na criação de métodos de combate.

Visto que os ataques de negação de serviço visam tornar os recursos de um sistema indisponíveis, seja pela sobrecarga ou pela obstrução do canal de comunicação do mesmo, é importante a detecção destes ataques com o máximo grau de precisão com pouco consumo de tempo, evitando desta forma, prejudicar usuários legítimos. Estas condições podem ser satisfeitas se houver a possibilidade de detectar rapidamente alterações no uso dos recursos do sistema.

Assim, é importante estudar e propor ações de gerência e segurança nestes ambientes no que se refere a ataques de *Denial of Service* (DoS). Desta forma, uma vez que seja possível sob uma base de políticas, identificar e separar os tráfegos pela instanciação em paralelo de diferentes métodos detecção e estabelecer uma regra para eliminação, redirecionamento ou isolamento deste tráfego, modificando a topologia e agindo de forma reativa ao ataque identificado sendo possível diminuir o impacto de uma ação de negação de serviço.

Neste trabalho é proposta uma arquitetura de defesa reativa para ambientes de redes virtuais contra ataques de negação de serviço (DoS). A arquitetura é implementada com atividades de três agentes e permite acessar uma base de políticas e instanciar métodos a partir de uma base de métodos para identificação do tráfego DoS na VN e aplicação de ações reativas de alteração de topologia, rota e banda contratadas.

Partindo deste princípio, uma vez que seja possível identificar o comportamento e separar o tráfego e estabelecer uma política para eliminação ou redirecionamento do mesmo, modificando a topologia e agindo de forma reativa ao ataque identificado pode-se diminuir o impacto de um ataque DoS.

1.1. MOTIVAÇÃO

A virtualização é a abstração de um dispositivo físico ou de recursos. O dispositivo físico poderia ser um computador, um processador, uma placa de rede, um chip de memória ou de armazenamento. A abstração permite que o dispositivo virtual possa ser utilizado como o dispositivo físico real.

Desse modo, a complexidade técnica do dispositivo físico torna-se escondida e uma interface mais simples é fornecida. A abstração permite a multiplexação em que um único dispositivo físico é compartilhado entre usuários ou pedidos, ou ainda demultiplexação, na qual um único pedido é dividido em múltiplos dispositivos físicos.

A virtualização de rede refere-se à emulação de uma rede e pode ser composta por hardware e software, combina a plataforma de virtualização e a virtualização de recursos, desta forma o consumo de recursos de forma inesperada ou talvez abusiva como em situações de negação de serviço.

Para entender o problema de ataques DoS em um NVE se faz necessário considerar os seguintes cenários: O que fazer se não for conhecida a quantidade de banda que uma aplicação irá usar? E se surgir uma inesperada fonte de tráfego de alguma aplicação ou alguma fonte de uma sub-rede com origem em uma VN tendo como alvo interromper múltiplas aplicações? Isto implica em um cenário de negação de serviços. Neste caso há necessidade de proteger os recursos do sistema primeiramente detectando a fonte e o tipo de ataque, e então criando fluxos que representem o tráfego de ataque e finalmente restringir os recursos para os fluxos do ataque DoS em um nível que possa ser administrado pelo sistema sem prejudicar o desempenho do mesmo nem tampouco provocar a cessão de algum serviço temporariamente.

Algumas peculiaridades a caracterizar, portanto, existentes em NVEs, são importantes para que seja possível tratar as questões descritas anteriormente:

1.1.1. Dinamismo em um NVE

A virtualização de rede apresenta um ambiente dinâmico para todo tipo de rede, a qual inicia com *usuários finais* individuais ou elementos de rede e continua até o nível de uma VN completa. Tal dinamismo pode ser caracterizado em duas classes (Mosharaf, et.al., 2008):

Nível Macro: VNs fornecem serviços básicos ou VNs com interesses comuns podem ser dinamicamente agregadas para juntas criar VNs compostas. No entanto

o nível de dinamismo esperado é muito pequeno e a complexidade em adicionar ou remover uma VN, pode apresentar um alto grau de complexidade.

Nível Micro: Este é nível de maior influência e requer mais atenção. O comportamento dinâmico em nível micro pode ser atribuído a dois amplos conjuntos de atividades (Wang, *et.al.*, 2008):

- A inserção, exclusão e mobilidade dinâmica de usuários finais dentro e entre VNs;
- Dinamismo ocasionado pela migração de roteadores virtuais por diferentes motivos.

1.1.2. Escalabilidade

A cada dia o número de usuários aumenta rapidamente e isto é esperado que seja continuo. Qualquer nova topologia proposta deve ser considerada escalável o suficiente para interagir e acomodar os fluxos de ataques provenientes desta escalabilidade ascendente.

1.2. ESCOPO

Este trabalho apresenta uma arquitetura de defesa reativa para ambientes de virtualização de redes através de uma estratégia de resposta a ataques de negação de serviço (DoS). A arquitetura não se preocupa com os tipos de ataques que a rede possa estar sofrendo, mas sim em reagir, quando uma incerteza ou certeza de ataque sejam identificadas.

O mecanismo define ações executadas por três agentes onde se realiza a configuração de equipamentos virtuais de rede com base em percentuais de certeza de classificação DoS com o intuito de separar incertezas de ataques legítimos com a limitação dos recursos de banda no caso de uma incerteza e a eliminação do tráfego no caso de certezas.

1.3. OBJETIVO GERAL

Desenvolver uma arquitetura de defesa contra ataques de negação de serviço chamada ARVDoS que atue com características reativas em ambientes virtualizados de rede para gerenciar e reconfigurar os enlaces e as tabelas de roteamento a fim de evitar a cessão de serviços ou a percepção pelo cliente.

1.3.1. Objetivos Específicos

Os objetivos específicos deste trabalho são:

1. Possibilitar a utilização de diferentes métodos de detecção propostos na literatura assim como sua integração.
2. Estabelecer um modelo para gerenciamento de tráfego e políticas de segurança em ambientes virtualizados, de baixo custo gerencial, transparente ao administrador de rede e ao usuário.
3. Ponderar as diferentes classificações de ataques DoS.
4. Propor uma estratégia de aplicação de redução do impacto de ataques de negação de serviço com baixo consumo de recursos de hardware.
5. Utilizar tecnologias de virtualização, a fim de apresentar um parecer sobre a arquitetura ARVDoS.

1.4. ESTRUTURA DA DISSERTAÇÃO

Este documento está organizado da seguinte maneira:

Capítulo 1: Introdução, motivação, objetivos e o escopo deste trabalho.

Capítulo 2: Uma revisão da literatura sobre virtualização de redes e ambiente virtualizado de rede.

Capítulo 3: Uma revisão da literatura sobre ataques DoS.

Capítulo 4: Uma fundamentação dos métodos de detecção para ataques DoS.

Capítulo 5: Apresentação da arquitetura ARVDoS e suas atividades.

Capítulo 6: Apresentação dos testes de avaliação da ARVDoS e resultados obtidos.

Capítulo 7: Conclusão da dissertação e trabalhos futuros.

2. VIRTUALIZAÇÃO DE REDES

Este capítulo apresenta uma perspectiva tecnológica da virtualização de recursos de rede e descreve os princípios fundamentais e os elementos básicos de uma rede virtualizada.

2.1. PERSPECTIVA TECNOLÓGICA

Na abordagem da perspectiva tecnológica é apresentado um mapeamento de tecnologias referentes à virtualização de redes. Os assuntos tratados são Redes Locais Virtuais, Redes Privadas Virtuais e Redes Overlay.

2.1.1. Redes Locais Virtuais

Uma Rede Local Virtual ou *Virtual Area Network,* é um grupo de *hosts* logicamente ligados em rede pertencentes a um mesmo domínio de *broadcast* sem levar em conta sua conectividade física. Todos os quadros em uma VLAN assumem um identificador, VLAN ID, no cabeçalho do protocolo de acesso ao meio ou *Medium Access Control, MAC,* desta forma uma VLAN configurada sobre *switches* utiliza para encaminhamento dos quadros o endereço MAC de destino e o VLAN ID. Visto que as VLANs são baseadas em conexões lógicas ao invés de físicas, a administração de rede, o gerenciamento, a reconfiguração das VLANs são mais simples que os equivalentes físicos (IEEE, 2006).

2.1.2. Redes Privadas Virtuais

Uma Rede Privada Virtual ou *Virtual Private Network, VPN,* (Ferguson e Huston, 1998) (Rosen e Rekhter, 1999) (Rosen e Rekhter, 2006) é uma rede dedicada que conecta múltiplos pontos utilizando túneis seguros e privados sobre redes de comunicação públicas ou compartilhadas como a Internet. Na maioria dos casos, VPNs realizam conexão de pontos geograficamente distribuídos de uma mesma empresa ou corporação.

Cada extremidade de uma VPN contém um ou mais clientes de borda, equipamentos que são ligados a um ou mais roteadores de borda, que são os provedores. Tipicamente uma VPN é gerenciada e mantida por provedor de serviço de VPN, o *provider-provisioned* VPN, PPVPN (Andersson e Madsen, 2005).

2.1.3. Redes Overlay

Uma rede *overlay* é uma rede lógica construída por cima de uma ou mais redes físicas existentes. A Internet iniciou como overlay por cima das redes de telecomunicações. As redes overlay no formato atual da Internet são tipicamente implementadas na camada de aplicação, entretanto, existem diversas implementações em camadas inferiores.

Estas redes não requerem, ou não causam nenhuma mudança às redes que formam sua base. Como consequência, as redes overlay têm sido utilizadas como meio relativamente fácil e de baixo custo para dispor novas características e aplicação de correções na Internet.

Uma grande quantidade aplicações em camadas têm sido propostas às redes overlay na abordagem de diversos aspectos, nos quais se inclui: assegurar desempenho (Savage *et. al.*, 1999) e disponibilidade (Andersen *et. al.*, 2001) de roteamento na Internet, permitindo transmissões *multicast* (Eriksson, 1994), (Janotti *et. al.*, 2000), (Chu *et. al.*, 2001), de forma a prover garantia de Qualidade de Serviço, QoS (Subramanian, *et. al.*, 2004), protegendo de ataques de negação de serviço, Denial of Service, DoS (Keromytis *et. al.*, 2002) (Andersen, 2003) e para distribuição de conteúdo (Krishnamurthy *et. al.*, 2001), compartilhamento de arquivos (Lua, *et. al.*, 2005) e em sistemas de armazenamento (Dabek, *et. al.*, 2001). As redes overlay têm sido utilizadas para testes (e.g. *PlanetLab* (Planetlab, 2009)) de projetos e avaliações de novas arquiteturas.

Entretanto, Anderson *et al.* (2005), apontam que as redes overlay deixam a desejar como um caminho de distribuição na inovação de arquiteturas em no mínimo duas formas. i) Overlay tem sido de forma abrangente vistas como um meio de dispor correções limitadas a problemas específicos sem uma visão holística das interações entre diferentes overlay. ii) A maioria das redes overlay têm sido projetadas na camada de aplicação em cima do protocolo de Internet, *Internet*

Protocol, IP, e por consequência não são capazes de suportar conceitos radicalmente diferentes.

2.2. VIRTUALIZAÇÃO DE REDES

A virtualização de redes é uma abordagem arquitetural para fornecer um ambiente de rede logicamente separado para cada grupo de usuários dentro de um grupo em uma organização. Estes ambientes lógicos são criados sobre uma única infraestrutura de rede compartilhada. Cada rede lógica prove ao grupo de usuários correspondente serviços de rede completos similares aqueles fornecidos em uma rede tradicional não virtualizada (Mosharaf, *et. al.*, 2008) (Moreno e Reddy, 2006).

Na perspectiva do usuário final as redes virtualizadas propiciam acesso a uma rede dedicada com todos os recursos que o usuário requisitar e políticas de segurança independentes.

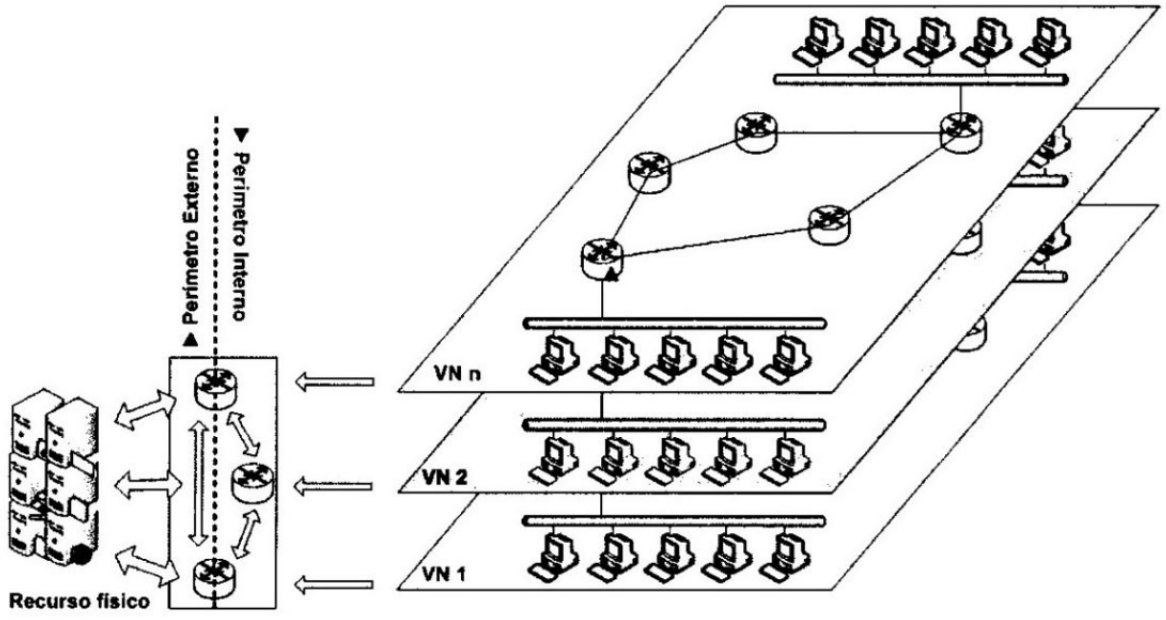


Figura 2.1 Recurso de virtualização a partir do perímetro da VN.
FONTE: Moreno e Reddy, 2006.

A perspectiva do administrador de rede é de poder facilmente criar e modificar ambientes virtuais de trabalho para os diferentes grupos de usuários e adaptar à evolução das demandas de necessidades. Esta última decorre da capacidade de criar zonas de segurança que são reguladas por políticas executadas centralizadamente. Uma vez que as políticas sejam executadas de forma

centralizada, adicionando ou removendo usuários e serviços a partir de ou para uma VN sem exigir qualquer reconfiguração de políticas. Contudo, as novas políticas que afetam um grupo inteiro podem ser implantadas centralizadamente no perímetro da VN (Mosharaf, et.al., 2008) (Moreno e Reddy, 2006).

Para *virtualizar* uma rede, as bases funcionais devem fornecer as seguintes funcionalidades (Moreno e Reddy, 2006):

- Dinamicamente autenticar e autorizar os usuários em grupos;
- Isolar conectividade entre os grupos para garantir a privacidade;
- Aplicar políticas de segurança independentes para cada grupo no perímetro;
- Fornecer serviços básicos de rede para os diferentes grupos quer estes sejam compartilhados ou dedicados;
- Fornecer domínios independentes de roteamento e espaços de endereçamento para cada grupo.

A partir de uma perspectiva de arquitetura, os pré-requisitos podem ser abordados pela segmentação da rede em VNs e centralizando a aplicação das políticas de rede no perímetro de cada VN.

Devido ao tráfego interno de uma zona ser confiável, as políticas são necessárias apenas no perímetro para controlar o acesso a recursos externos que poderiam, em muitos casos, ser compartilhados. A Figura 2.1 ilustra este conceito.

Isso torna os usuários móveis e assegura que independentemente da sua localização sempre serão sujeitos às mesmas políticas. Para garantir que os utilizadores estão sempre conectados à VN correta, mecanismos dinâmicos de autenticação e autorização são obrigatórios. Isto permite a identificação dos dispositivos, usuários e até mesmo aplicações, no entanto devem ser autorizados no segmento virtual correto para assim herdar as políticas do segmento (Mosharaf, et.al., 2008) (Moreno e Reddy, 2006)..

Desta forma, a virtualização de redes envolve a segmentação lógica do transporte, dispositivos e sua interligação e serviços de redes direcionando a duas áreas de foco (Mosharaf, et.al., 2008) (Moreno e Reddy, 2006):

- **Virtualização de caminho:** Refere-se à virtualização de interconexão entre dispositivos. Este poderia ser uma interconexão com um salto ou multi salto.

(e.g. uma ligação entre dois switches Ethernet proporciona um único salto). Um exemplo de uma interligação multi salto seria fornecido por uma nuvem IP entre dois dispositivos. Esta interligação pode ser virtualizada através da utilização de múltiplos túneis entre os dois dispositivos.

- **Virtualização de Dispositivo:** Refere-se à virtualização de um dispositivo de rede ou a criação de dispositivos lógicos dentro do dispositivo físico. Isto inclui a virtualização de todos os processos, bases de dados, tabelas e interfaces em um dispositivo físico.

Por sua vez, em cada dispositivo de rede, há pelo menos duas características para virtualizar:

- **Características de controle:** Referem-se a todos os protocolos, bases de dados e tabelas necessárias para fazer encaminhamento e manter uma topologia de rede livre de *loops* ou falhas involuntárias. Esse plano pode ser usado para obter uma perspectiva clara da topologia da rede para o dispositivo. Um dispositivo virtualizado deve possuir uma única função atribuída em cada VN a ser gerenciada.

- **Características de encaminhamento:** Refere-se a todos os processos e as tabelas utilizadas para o encaminhamento do tráfego. Constrói as tabelas de encaminhamento a partir das informações obtidas das características de controle.

O administrador/projetista de rede deve observar de que forma os blocos funcionais de uma VN estão interligados e preocupar-se que a virtualização de rede não onera em custos de desempenho a rede física.

É importante lembrar que ao virtualizar uma rede, nem tudo deve ser migrado para a VN criada. As tecnologias de virtualização de redes são sobrepostas aos InPs operacionais existentes. Por isso, a rede física continua a funcionar da mesma forma que antes da virtualização de seus recursos, mas com VNs sobrepostas em seus recursos (Mosharaf, *et. al.*, 2008).

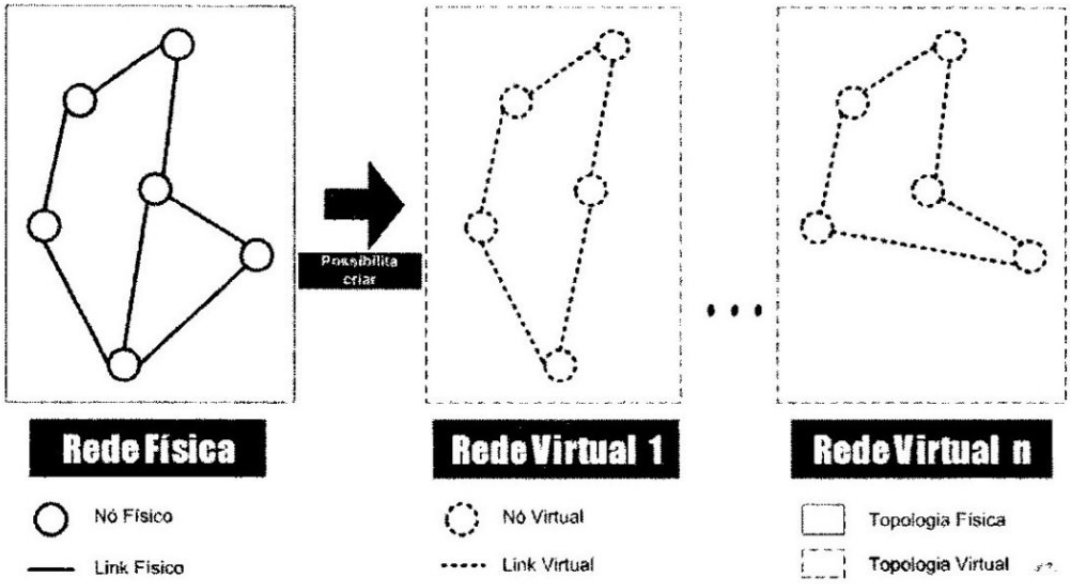


Figura 2.2 Rede virtual – NV (Adaptado)
FONTE: Mosharaf, *et. al.*, 2008

Por causa das diversas redes lógicas que compartilham uma mesma infraestrutura e um mesmo conjunto de serviço de aparelhos e servidores, muitas vezes centralizado, os grupos de usuários podem colaborar na melhoria da flexibilidade e do gerenciamento.

Assim, uma rede virtual é um conjunto de nós virtuais e enlaces virtuais que formam uma topologia virtual, conforme ilustra a Figura 2.2.

2.3. ELEMENTOS BÁSICOS DA VIRTUALIZAÇÃO DE REDES

Quando uma rede é virtualizada, ela deve satisfazer critérios de segurança, escalabilidade e níveis de demanda. Para fornecer o nível apropriado de conectividade, uma rede virtualizada deve seguir os mesmos princípios dos projetos que têm fornecido, segurança e escalabilidade nas redes locais, LANs, redes metropolitanas, MANs, e redes de grande abrangência, WANs (Mosharaf, et.al., 2008) (Moreno e Reddy, 2006)..

As redes permitem que os usuários tenham acesso a serviços e recursos distribuídos. Alguns desses serviços e recursos são públicos, aqueles acessados pela Internet, e outros que são privados, acessados na infraestrutura interna. Cada rede possui níveis de políticas únicos para segurança e para os serviços, que gerenciam a conectividade, quer sejam os diferentes serviços públicos ou privados.

Um elemento importante de uma política de segurança para a virtualização de redes é a definição de um perímetro de rede. Em geral, o nível de confiança dentro e fora do perímetro de rede se difere, estações no interior do perímetro geralmente consideradas confiáveis e qualquer acesso de fora do perímetro são considerados não confiáveis (Moreno e Reddy, 2006).

A comunicação entre o interior e o exterior do perímetro deverá acontecer através de uma verificação. Na verificação, *firewalls* e outros dispositivos de segurança devem assegurar que todo o tráfego que entra ou sai da empresa seja rigorosamente controlado. Por isso, refere-se que o ponto de entrada/saída, *gateway* da rede como o seu perímetro.

O perímetro da rede define uma camada de segurança que devem ser complementados com outros mecanismos de segurança. É crítico incorporar mecanismos para proteger a rede contra ataques iniciados no interior do perímetro. Esta funcionalidade é geralmente fornecida na rede de acesso e não é impactado pela virtualização da rede.

Quando uma única rede fornece serviços a grupos diferentes, é necessário criar redes virtualizadas, de modo a satisfazer cada grupo especificamente nos seguintes critérios (Moreno e Reddy, 2006):

- Conectividade privada sobre uma infraestrutura compartilhada;
- Um perímetro dedicado, no qual políticas independentes podem ser executadas por grupo;
- Acesso adequado à rede virtual independentemente da localização do usuário.

Uma VN pode ser vista como uma zona de segurança, qualquer comunicação com outras zonas de segurança, ou outras redes, precisa acontecer de forma controlada, garantida pela autenticação. Assim, uma rede virtualizada irá conter simultaneamente muitas zonas de segurança, e os seus perímetros dedicados, sobre uma infraestrutura compartilhada.

2.4. AMBIENTE DE VIRTUALIZAÇÃO DE REDES

Trata-se de um ambiente de rede que permite múltiplos provedores de serviços, formarem dinamicamente múltiplas redes virtuais que coexistem em isolamento uma das outras e permitem distribuir serviços fim a fim, assim como gerenciá-los em suas redes virtuais para usuários finais utilizando e compartilhando os recursos base sustentados por múltiplos provedores de infraestrutura (Mosharaf, et.al., 2008). A Figura 2.3 ilustra este ambiente.

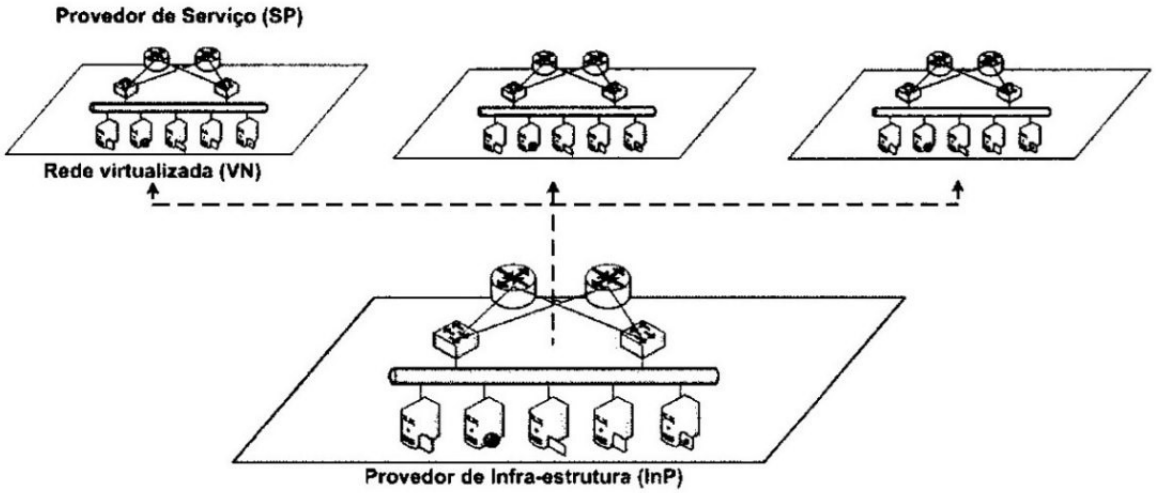


Figura 2.3 Ambiente de virtualização de redes (NVE).
FONTE: Moreno e Reddy, 2006.

Nas redes virtualizadas os InPs distribuem e também gerenciam os recursos físicos que formam a base do NVE. Estes fornecem seus recursos com a utilização de interfaces programáveis para diferentes SP. Os InPs se comunicam e colaboram entre si, baseados em acordos específicos que completam a rede base. Alguns InPs oferecem conectividade aos SP por diferentes tecnologias de rede e são conhecidos por *facilities providers,* FP. Por outro lado os InPs que conectam usuários finais e equipamentos ao núcleo da rede são conhecidos por provedores de acesso, ou *Acccess Provider,* AP, a Figura 2.3 apresenta o modelo hierárquico de um ambiente virtualizado de redes (Moreno e Reddy, 2006).

Os SPs por sua vez locam recursos virtuais de múltiplos InPs em ordem de sintetizar as VNs. Distribuem protocolos customizados pela programação dos recursos virtuais alocados para oferecer serviços fim a fim aos usuários finais.

Os usuários finais, clientes, em um modelo de virtualização de redes são similares aos usuários finais no paradigma atual de Internet, exceto que pela existência de múltiplas VNs de SPs concorrentes possibilitam aos usuários finais

selecionar uma grande gama de serviços. Qualquer usuário final pode simultaneamente conectar a múltiplas redes virtuais de diversos SPs distintos para obter diferentes serviços (Mosharaf, et.al., 2008).

2.5. REQUISITOS DAS REDES VIRTUALIZADAS

Em um NVE, a entidade básica é a VN, que é uma coleção de nós virtuais conectados por um conjunto de enlaces virtuais formando uma topologia virtual. Enquanto cada VN é composta e gerenciada por um único SP, esta pode cruzar por múltiplas redes físicas. Uma vez criado um SP, uma VN tem a aparência de uma rede física propriamente dita.

A meta principal em permitir a coexistência de múltiplas redes virtuais em conjunto compartilhando uma mesma infraestrutura física pode ser dividida em alguns objetivos a fim de viabilizar a virtualização de redes. Para que isso se realize cada um destes objetivos precisa ser cumprido. Essas metas também fornecem orientações para a concepção de protocolos e algoritmos para redes virtuais (Feamster, *et. al.*, 2005), (Boucadair *et. al.*, 2007), (Zhu e Ammar, 2006), (Turner e Taylor, 2005) (Mosharaf, et.al., 2008) (Moreno e Reddy, 2006).

- **Flexibilidade:** A virtualização de rede deve proporcionar a flexibilidade em todos os aspectos do trabalho em rede. Todos os SP devem ser capazes de usar arbitrariamente a topologia da rede para realizar funções de roteamento ou encaminhando, independente da rede física e outras VNs coexistentes. (e.g. em um ambiente virtualizado o proprietário de uma VN deve ser capaz de oferecer uma fonte de roteamento sem precisar se coordenar com quaisquer outras partes).

- **Capacidade de ser gerenciável:** Ao separar SPs dos InPs, a virtualização de redes modulariza as atividades de gerenciamento de rede. Os InPs permitem controle total da gerência e das operações sobre as entidades físicas na rede, assim como fornecem acesso aos recursos físicos. Os SPs, por outro lado, utilizam recursos de diferentes InPs para criar redes virtuais em cima dos recursos atribuídos, seguindo políticas específicas e prestam efetivamente

serviços aos usuários finais. Esta separação de responsabilidade irá fornecer controle fim a fim completo das VNs.

- **Escalabilidade:** A coexistência de múltiplas redes é uma das bases da virtualização de redes. A coexistência é permitida enquanto os recursos físicos disponibilizados pelos InPs suportarem a criação de diversas VNs. A escalabilidade se apresenta como um importante requisito nesta relação. Os InPs deveriam maximizar o número de VNs coexistentes sem afetarem seu desempenho. Isto irá aumentar o consumo de recursos e diminuir o consumo de capital e operacional de VNs individuais.

- **Isolamento:** A virtualização de redes deve assegurar o completo isolamento físico e lógico entre as VNs coexistentes para melhorar a tolerância a falhas, segurança e privacidade. Os protocolos de rede são frequentemente mal configurados e sujeitos a erros de implementação. A virtualização deve assegurar que erros de configuração em uma VN são pertinentes ao seu domínio e não devem afetar outras VNs coexistentes.

- **Estabilidade e Convergência:** O isolamento garante que defeitos em uma VN não afetam outras VNs coexistentes, mas erros e configurações incorretas na rede física podem desestabilizar o NVE. Além disso, a instabilidade no InP (e.g. o oscilação no roteamento) pode levar à instabilidade de todas as VNs hospedadas. A virtualização deve assegurar a estabilidade do NVE, e, no caso de uma eventual instabilidade, as VNs afetadas devem ser capazes de convergir para um estado estável.

- **Heterogeneidade:** No contexto da virtualização de redes a heterogeneidade vem principalmente a partir de duas frentes: i) a heterogeneidade das tecnologias da rede física (e.g. óptico, sem fios, sensores, entre outros); ii) cada VN fim a fim, criado sobre a combinação de redes físicas heterogêneas, também devem ser heterogêneas. Os InPs devem ser capazes de suportar protocolos e algoritmos heterogêneos implementados por diferentes SPs. Além disso, a heterogeneidade dos dispositivos do usuário final também deve ser levada em consideração.

3. NEGAÇÃO DE SERVIÇO

Negação de Serviço, ou, *Denial of Service,* DoS, é um ataque que permite tornar um sistema inutilizável ou consideravelmente lento para os usuários legítimos através do consumo de seus recursos. Neste capítulo são apresentadas as características, os tipos de ataques mais comuns e os mecanismos de defesa para os ataques DoS.

3.1. ATAQUES DE NEGAÇÃO DE SERVIÇO

Um ataque de negação de serviço é caracterizado por uma tentativa explícita de impedir o uso legítimo de um serviço. As técnicas visam sobrecarregar o tráfego de uma rede ao ponto de não permitir que seus usuários verdadeiros a usem (CERT, 2009).

Desta forma podem cessar a conexão entre grupo de *hosts*, realizar um número elevado de requisições de conexão a um determinado servidor *web* até que o acesso a este não seja mais possível, não permitir que determinados usuários conectem a um serviço ou até negar a autenticação em um sistema qualquer com arquitetura baseada em redes (e.g. aplicações cliente servidor).

Uma forma de realizar um ataque DoS pode ser configurada quando o atacante envia fluxos de pacotes a uma vítima, este fluxo consome recursos essenciais de um sistema, assim, os torna inacessível aos usuários legítimos. Outra abordagem comum a ataques de negação de serviço ocorre quando o atacante envia pacotes mal formados, que ocasionam confusão para alguma aplicação ou um protocolo na máquina atacada, desta forma força o travamento ou reinicialização do sistema.

3.2. TIPOS DE ATAQUES DE NEGAÇÃO DE SERVIÇO

Os ataques DoS exploram diferentes vulnerabilidades a fim de negar algum serviço à vítima. Nesta seção são apresentados alguns tipos de ataques que se referenciam a vulnerabilidade e *flooding*. Nesta abordagem são citados os ataques baseados em vulnerabilidade, ou problema de semântica, ataque de força bruta, taxa dinâmica de ataque e possibilidade de caracterização (Giampaolo, 2008)

(Cisco, 2004) (Silva *et. al.*, 2006) (Luo *et. al.*, 2009) (Mirkovic e Reiher, 2004) (Mirkovic *et. al,* 2005) (Mirkovic, 2003) (Mirkovic *et. al.*, 2002) (Patcha e Park, 2007).

3.2.1. Ataques Baseados em Semântica de Protocolos

Os ataques exploram uma característica específica ou erro de implementação de algum protocolo ou aplicação instalada na vítima a fim de consumir em excesso seus recursos. Como exemplo, no ataque conhecido por TCP SYN, a falha explorada é a alocação substancial de espaço em uma fila de conexões, imediatamente, uma vez que acuse o recebimento de uma requisição TCP SYN. O atacante inicia múltiplas conexões que nunca serão completadas, assim ocupando a fila de conexão.

Este ataque inicia e estabelece numerosas conexões TCP que consomem a fila de conexão na vítima. Este ataque contorna a pilha do protocolo TCP na máquina agente, e não mantém o estado para as conexões que originou. Ao invés disto, participa na conexão, inferindo atributos de respostas dos pacotes recebidos. Desta forma até mesmo um agente que produza um ataque sem muita força pode facilmente cessar os recursos de uma vítima bem protegida.

3.2.2. Ataques Baseados em Inundação de Pacotes

Durante o ataque, cada agente participante envia uma sequência de pacotes à rede atacada. De acordo com a dinâmica da taxa de ataque de um agente de ataque, é possível diferenciar em três classificações: i) taxa constante de ataque, ii) aumento da taxa e iii) taxa flutuante por (Bernstein, 2009) (Ferguson e Senie, 2000) (Bellovin, 2008) (Mirkovic *et. al.*, 2002) (CISCO, 2004) (Mirkovic, 2003) (Luo *et. al.*, 2009) (Mirkovic *et. al.*, 2005) (Mirkovic e Reiher, 2004) (Patcha e Park, 2007).

- **Taxa constante de ataque:** A maioria dos ataques conhecidos implementa mecanismos com taxa constante de ataque. Assim que o início for ordenado, as máquinas agentes geram pacotes de ataque a um taxa constante, usualmente tantos quanto seus recursos permitam. A inundação repentina de pacotes interrompe os serviços na vítima rapidamente. Esta abordagem

proporciona o melhor custo-eficácia para o atacante, pois este pode implantar um número mínimo de agentes para propiciar o dano. Por outro lado, o grande e contínuo fluxo de tráfego pode ser detectado como anômalo e despertar suspeitas na rede atacada, facilitando assim a descoberta do ataque.

- **Aumento da Taxa**: Ataques que possuem um gradual aumento de taxa direcionam para uma exaustão lenta dos recursos das vítimas. Os serviços se degradam lentamente durante um longo período de tempo, desta forma, substancialmente retardando a detecção do ataque. Em adição, estes ataques podem manipular mecanismos de defesa que treinam seus modelos de detecção.

- **Taxa Flutuante:** Ataques que apresentam taxa flutuante ajustam a taxa de ataque com base no comportamento da vítima ou por tempo pré programado, ocasionalmente aliviando o efeito para evitar a detecção. Por exemplo, durante um ataque do tipo *pulsing attack* (Luo *et. al.*, 2009), os agentes periodicamente interrompem o ataque e o reiniciam algum tempo depois. Se este comportamento for simultâneo para todos os agentes, a vítima terá seus serviços periodicamente interrompidos. No entanto, se os agentes são divididos em grupos que se coordenam de modo que um grupo está sempre ativo, a vítima sofre contínua negação de serviço, enquanto a máquina agente na rede hospedeira pode não notar qualquer anomalia prolongada.

3.2.3. Ataques Baseados em Possibilidade de Caracterização

Observando-se os campos de conteúdo e cabeçalho dos pacotes de ataque, é possível algumas vezes caracterizar o ataque. A caracterização pode levar à elaboração de regras de filtragem. Baseado na possibilidade de caracterização é possível diferenciar os ataques em: i) caracterizável, ii) filtrável, iii) não filtrável e iv) não caracterizável.

- **Caracterizável:** Os ataques caracterizáveis são aqueles que têm por alvo protocolos e aplicações específicas e podem ser identificadas pela combinação dos valores do cabeçalho IP e protocolo de transporte, ou talvez pelo conteúdo do pacote. Alguns exemplos incluem o ataque TCP SYN, onde apenas pacotes com o bit SYN ajustado no cabeçalho TCP pode potencialmente ser parte de um ataque, ataque ICMP ECHO, ataques de requisições DNS, entre outros.

- **Filtrável:** Os ataques filtráveis são aqueles que utilizam pacotes mal formados ou pacotes de serviços não críticos. Estes podem ser filtrados por um *firewall*. Como exemplos destes ataques é possível citar: ataques de inundação UDP ou um ataque de inundação ICMP ECHO em um servidor Web. Desde que o servidor Web somente necessite tráfego TCP e algum tráfego DNS, pode facilmente bloquear qualquer outro tráfego de entrada UDP e todo tráfego ICMP, e ainda operar corretamente.

- **Não Filtrável:** Os ataques não filtráveis utilizam pacotes bem formados que requisitam serviços legítimos e críticos ao sistema da vítima. Portanto, a filtragem de todos os pacotes que são semelhantes à caracterização de ataque levaria a uma imediata negação de serviço a um serviço específico em ambos atacante e clientes legítimos. Exemplos são encontrados em inundação de requisições HTTP em um servidor Web ou em inundações de requisições DNS apontando para um servidor conhecido. No caso de ataques não filtráveis, o conteúdo de um pacote de ataque é impossível de ser distinguido do conteúdo de pacotes originados de usuários legítimos.

- **Não Caracterizável:** Os ataques não caracterizáveis procuram consumir a banda da rede utilizando uma variedade de pacotes que envolvem diferentes aplicações e protocolos. Algumas vezes os pacotes serão gerados randomicamente usando números reservados de protocolos. É possível observar que a classificação como ataque caracterizável ou não caracterizável, depende dos recursos que podem ser dedicados a caracterização e ao nível de caracterização. Por exemplo, um ataque utilizando um merge de pacotes TCP SYN, TCP ACK, ICMP ECHO, ICMP

ECHO REPLY e UDP poderia provavelmente ser caracterizável, mas somente após considerável esforço e tempo, e somente se um obtiver acesso a uma sofisticada ferramenta de caracterização. Além disso, um ataque utilizando uma mistura de pacotes TCP com várias combinações dos campos do cabeçalho TCP pode ser caracterizado como um ataque, mas uma caracterização mais apurada seria provavelmente de alto custo.

3.3. MECANISMOS DE DEFESA

A seriedade do problema de ataque de negação de serviço e a crescente frequência, sofisticação e força dos ataques tem direcionado a diversas propostas de mecanismos de defesas. Com base no nível de atividade dos mecanismos de defesa contra DoS, é possível classificá-los como preventivos e reativos.

3.3.1. Preventivo

Os mecanismos preventivos tentam eliminar a possibilidade de ataques DoS completamente, ou também possibilitar que vítimas em potencial resistam ao ataque sem que ocorra a negação de algum serviço aos usuários legítimos.

Estes mecanismos, de acordo com o objetivo de prevenção de negação de serviço, podem ser divididos em dois tipos: i) Prevenção de ataque e; ii) Prevenção de negação de serviço, e descritos por (Axelsson, 2000) (Bernstein, 2009) (Ferguson e Senie, 2000) (Lau *et. al.*, 2000) (Bellovin, 2008) (Ioannidis e Bellovin, 2002) (Mirkovic *et. al.*, 2002) (Thomas *et. al.*, 2003) (Keromitys *et. al.*, 2004) (Garg e Reddy, 2002) (Cisco, 2004) (Mirkovic, 2003) (Luo *et. al.*, 2009) (Mirkovic *et. al.*, 2005) (Patcha e Park, 2007).

- **Prevenção de ataque:** Estes mecanismos modificam sistemas e protocolos na Internet para eliminar a possibilidade de execução de um taque DoS. O histórico da segurança de computadores sugere que uma abordagem de segurança nunca pode ser cem por cento efetiva, desde que a aplicação em um cenário global não pode ser garantida. Entretanto, a execução de técnicas eficientes aplicadas de forma local, certamente irá decrementar a freqüência e

a força dos ataques de negação de serviço. A aplicação de mecanismos de prevenção pode tornar um *host* hábil a reconhecer ataques de protocolo. Também, estas abordagens são compatíveis e complementares com qualquer outra técnica de defesa.

- **Prevenção de negação de serviço:** Estes mecanismos permitem às vítimas resistir às tentativas de ataque sem negar algum serviço para usuários legítimos. Isto é feito ao reforçar as políticas para consumo de recursos ou pela garantia de que recursos abundantes existem de forma que usuários legítimos não serão afetados pelo ataque.

3.3.2. Reativo

Os mecanismos reativos esforçam-se para aliviar o impacto de um ataque sobre uma vítima. Para alcançar este objetivo se faz necessário detectar este ataque e responder ao mesmo. A meta principal é detectar todas as tentativas de ataque DoS o mais rapidamente possível e ao mesmo tempo obter o menor número de incertezas. Algumas ações devem ser tomadas para caracterizar os pacotes de ataque e fornecer estas caracterizações para um mecanismo de defesa.

Os mecanismos de defesa reativos são classificados em duas estratégias de: i) Detecção de ataque (*Attack Detection Strategy – ADS)* e; ii) Resposta ao ataque (*Attack Response Strategy - ARS)*. O primeiro pode ser dividido em: Detecção por padrões, Detecção por anomalias e Detecção por mensagens externas. O segundo pode ser dividido em: Identificação por agentes, Limitação de taxa, Filtragem e Reconfiguração.

As classificações do ADS são descritas por (Mahajan *et. al.*, 2002) (ISI, 2009) (Gil e Poletto, 2001) (Cs3, 2009) (Arbor, 2009) (Mirkovic, 2003) (Savage *et. al.*, 2000) (Mirkovic e Reiher, 2004):

- **Detecção por padrões:** Este mecanismo armazena as assinaturas de um ataque conhecido em um banco de dados e monitora cada comunicação a procura da presença de algum padrão no tráfego atual. Uma desvantagem é apontada em que somente ataques conhecidos podem ser detectados,

enquanto novos ataques ou mesmo variações imperceptíveis de ataques conhecidos não serão reconhecidas. Por outro lado, ataques conhecidos são facilmente reconhecidos com confiabilidade e sem a presença de incertezas. O Sourcefire (2009) apresenta um exemplo de um sistema de defesa contra DoS que usa a detecção por padrões. Uma abordagem semelhante tem sido utilizada de forma satisfatória no controle de viroses de computador. Como nos programas antivírus, a base de dados de assinaturas destes mecanismos precisa ser regularmente atualizada para proteger-se de novos ataques.

- **Detecção por anomalias:** Mecanismos que aplicam este tipo de detecção possuem o modelo de um sistema normal de comportamento, assim como tráfegos dinâmicos normais ou desempenho esperado do sistema. O estado atual do sistema é periodicamente comparado com modelos para detectar anomalias no tráfego. A vantagem desta técnica em relação à detecção por modelos é que ataques ainda não conhecidos podem ser descobertos, porém a habilidade de detectar todos os ataques faz com que esta técnica apresente uma tendência em identificar incorretamente tráfego normal como de ataque.

O objetivo da ARS é aliviar o impacto do ataque sobre as vítimas enquanto impõe mínimo efeito colateral aos clientes legítimos. A descrição de sua classificação é dada por (Mahajan *et. al.*, 2002) (Cs3, 2009) (Mircovic *at. al.*, 2002) (Mirkovic, 2003) (Arbor, 2009) (Darmohray e Oliver, 2000) (ISI, 2009) (Mirkovic e Reiher, 2004):

- **Identificação por agentes:** Este mecanismo fornece à vítima informações sobre a identidade das máquinas que estão executando o ataque. Esta informação pode ser usada por outras abordagens para aliviar o impacto do ataque. Como exemplo de identificação por agentes incluem-se técnicas de *traceback*.
- **Limitação de taxa:** Este mecanismo impõe um limite de taxa sobre um conjunto de pacotes que tenham sido caracterizados como maliciosos por um mecanismo de detecção. Trata-se de uma técnica de resposta que é usualmente aplicada quando o mecanismo de detecção caracteriza muitas incertezas no tráfego ou não consegue efetivamente identificar DoS. A

desvantagem é que a limitação da taxa permitirá a passagem de algum tráfego de ataque, desta forma, ataques em grande escala serão efetivos, mesmo que todo o fluxo tenha a banda limitada.

- **Filtragem:** Estes mecanismos usam as características fornecidas por mecanismos de detecção para filtrar completamente um tráfego. Alguns exemplos incluem *firewalls* dinamicamente configurados, e alguns sistemas comerciais. Ao menos que a caracterização seja muito precisa, mecanismos de filtragem correm o risco de acidentalmente negar serviço a tráfego legítimo. Em outra situação, ataques inteligentes, podem utilizar esta técnica como ferramenta para DoS.

- **Reconfiguração:** Estes mecanismos mudam a topologia da rede atacada ou da rede intermediária, para adicionar mais recursos ou isolar as máquinas atacantes. Como exemplos incluem-se as redes overlay reconfiguráveis, serviço de replicação de recurso, estratégias de isolamento de ataque, entre outros.

4. MÉTODOS DE DETECÇÃO

Neste capítulo são apresentados e descritos alguns métodos de detecção DoS os quais são importantes na aplicação da arquitetura para obtenção do percentual de certeza.

4.1. DETECÇÃO DE NEGAÇÃO DE SERVIÇO

O principal objetivo de um detector DoS é identificar e distinguir pacotes maliciosos de pacotes legítimos no tráfego de rede (Mirkovic e Reiher, 2004) (Mirkovic, 2005). Se, por exemplo, muitos clientes solicitam um serviço web e ao mesmo tempo ocorre uma inundação DoS em várias sessões web, como o servidor web irá discriminar entre os pedidos? Claramente, a atividade do usuário legítimo pode ser facilmente confundida com um ataque de negação de serviço.

Quando uma grande quantidade de tráfego, esperado ou inesperado, a partir de clientes legítimos repentinamente chegar a um sistema, há a necessidade de verificar estes eventos. Uma forma de prever esses acontecimentos e assim distingui-los de ataques DoS representa para servidores que: adicionar novos conteúdos podem desencadear grandes volumes de pedido. Atividades imprevisíveis e legítimas na web também são possíveis, no entanto por não existir um mecanismo inato na Internet para efetuar diferenciação de tráfego malicioso, as alternativas que se propõe são instalar detectores de ataque para controlar o tráfego ou consumo físico de recursos (e.g. tempo de uso de processador ou processos consumindo elevada quantidade de memória) em tempo real, ao invés de basear-se em previsões estatísticas de carga na rede.

As abordagens para detecção de ataque DoS podem ser configuradas de forma local, protegendo assim uma possível vítima, ou remotamente, a fim de detectar a propagação ataques em um ambiente específico. Embora detectar a propagação de ataques propagação seja desejável, em sua prática os departamentos de Tecnologia da Informação, TI, tem por opção a proteção de suas próprias redes, portanto, utilizam abordagens locais de detecção. Nesse caso, instanciam-se detectores em vítimas em potencial (e.g. um roteador ou firewall) de uma sub-rede. Partindo deste pressuposto, limita-se o escopo de tratamento de negação de serviço (Carl e Kesidis, 2006).

Qualquer método de detecção define um ataque como um desvio anormal e notório a cerca da estatística controlada do tráfego de rede. Nas seções seguintes deste capítulo são descritos métodos de detecção.

4.2. TÉCNICAS DE DETECÇÃO

As cargas de ataque de vulnerabilidades usam atributos comuns para explorar as fraquezas do software. Um ataque TCP SYN, por exemplo, requer o uso repetitivo de campos *flag* TCP específicos. Os atacantes endereçam o ataque TCP SYN utilizando *syn cache*, *syn cookies* e mecanismos *syn kill*, por exemplo.

Embora as vulnerabilidades referentes a falhas de protocolo ou aplicação sejam corrigidas por seus desenvolvedores, estes tipos de ataques mantêm-se problemáticos. Se o seu volume é suficiente para causar o esgotamento de recursos e posterior degradação de desempenho, eles podem ser reclassificados como ataques por inundações, *flooding attacks*. Por essa razão, ataques de inundação são especialmente difíceis de controlar, porque mesmo o melhor sistema de manutenção pode tornar-se congestionado, negando serviço a usuários legítimos.

A meta principal de um método de detecção é distinguir o tráfego de pacotes maliciosos do tráfego de pacotes legítimos, por exemplo, se muitos clientes requisitam um serviço web e da mesma forma um ataque malicioso DoS gera inundações para sessões web, como pode o servidor discriminar entre os pedidos? Claramente, a atividade do usuário legítimo pode ser facilmente confundida com um ataque de inundação e vice-versa.

Quando grandes quantidades de tráfego, esperados ou inesperados, de clientes legítimos chegam de repente a um sistema são chamados de eventos *flash*, ou rajadas. Como uma maneira de prever tais eventos e assim distingui-los dos ataques DoS, a priori, significa para os provedores de serviços que a adição de novos conteúdos podem provocar um grande volume de pedidos. Desta forma atividades imprevisíveis e legítimas também são possíveis (e.g. uma notícia nova publicada em um servidor *web* que irá gerar um grande volume de acessos). Por não existir um mecanismo inato para executar a discriminação de tráfego malicioso, considera-se a melhor alternativa instalar detectores de ataque para monitorar em tempo real o tráfego e evitar confiar nas previsões de tráfego.

As abordagens de detecção de ataques DoS podem ser instaladas localmente, protegendo assim uma possível vítima, ou remotamente, para detectar - ataques propagados. Embora a detecção de ataques de propagação seja desejável, os departamentos de TIC geralmente têm como foco a proteção de suas próprias redes e, portanto, optam por métodos locais de detecção. Neste caso, são instalados detectores nos recursos que são vítimas em potencial (e.g. roteador ou firewall) dentro sub-rede.

Partindo deste pressuposto, delimitam-se o domínio de detecção de ataques de negação de serviço, que exclui vários outros métodos de detecção tais como o código-fonte com base DWARD6, rastreamento, identificação do caminho, entre outros.

Nas seções seguintes são descritos alguns métodos de detecção de negação de serviço.

4.2.1. Perfil de Atividade

Neste método de detecção é possível utilizar grupos individuais com fluxos de características semelhantes, assim o monitoramento das informações dos cabeçalhos dos pacotes de rede oferece um perfil de atividade do tráfego. Esta atividade, em uma breve definição, é o perfil da taxa média dos pacotes para um fluxo de rede, que consiste em pacotes consecutivos com campos semelhantes (e.g. endereço, porta ou protocolo), os grupos de atividades.

O tempo decorrido entre pacotes consecutivos determina o fluxo médio da taxa de pacotes ou o nível de atividade. É possível medir a atividade total de rede como a soma da média ao longo de todas as taxas de pacotes de entrada e saída de fluxos (Carl e Kesidis, 2006) (Feinstein *et. al.*, 2003). O aumento da atividade entre os níveis agrupados pode indicar o aumento da geração das taxas de ataque;

4.2.2. Detecção por mudança sequencial de pontos

Os métodos de detecção por mudança sequencial de pontos (Blazek *et. al.*, 2001) (Brooks, 2005) (Carl e Kesidis, 2006) isola as estatísticas da mudança de um tráfego causada por ataques de negação de serviços. Inicialmente esta abordagem

filtra os dados do tráfego por endereço, porta ou protocolo e armazena o fluxo resultante como uma série temporal.

A série temporal (Alves, 2003) pode ser considerada um tempo de representação de atividade no domínio de um grupo. Se um ataque DoS por inundação começa no momento τ , a série de tempo irá mostrar uma mudança estatística em um tempo maior do que τ .

Uma classe de algoritmos de detecção de mudança de ponto opera continuamente na amostra de dados e exige o consumo de pequenas quantidades computacionais.

Para identificar e localizar um ataque de negação de serviço, o algoritmo de soma cumulativa identifica desvios no próprio local em comparação com a média esperada no tempo de tráfego (Alves, 2003) (Montgomery, 2005). Se a diferença exceder o limite máximo, a estatística da soma cumulativa recursiva aumenta para a série de tempo da amostra. Durante intervalo de tempo da amostra que contêm apenas intervalos normais de tráfego, a diferença é inferior ao desejado, e a estatística da soma cumulativa diminui até chegar a zero.

Usando um limite adequado em comparação a estatística da soma cumulativa, o algoritmo identifica uma tendência crescente na série de tempo para os dados analisados, o que pode indicar o aparecimento de um ataque DoS (Alves, 2003) (Montgomery, 2005). Através das configurações do limiar e do limite superior, o algoritmo pode descartar a detecção de atrasos e incertezas na análise de um tráfego de rede (Jung *et. al*, 2004).

4.2.3 Análise de Wavelets

Wavelet é uma função capaz de decompor e descrever outras funções no domínio da freqüência, de forma a analisar estas funções em diferentes escalas de freqüência e de tempo (Brooks, 2005).

A decomposição de uma função com o uso de *wavelets* é conhecida como transformada de *wavelet* e tem suas variantes: contínua e discreta. Sua capacidade de decompor as funções tanto no domínio da frequência quanto no domínio do tempo, torna as funções *wavelet* ferramentas para a análise de sinais e compressão de dados. Em geral a transformada contínua de wavelet é usada na análise de

sinais, enquanto a sua versão discreta é usada na compressão de dados (Brooks, 2005) (Barford et. al., 2002) (Lu e Ghorbani, 2009).

As *wavelets* fornecem uma descrição para tempo e frequência concorrentes e assim pode determinar o momento em que certos componentes de frequência estão presentes. Para aplicações de detecção, as wavelets separam sinais de ruído de tráfego anômalo, ambos verificados em um sinal de entrada. De forma geral, o tráfego e o sinal de ruído serão separados em diferentes janelas espectrais em um determinado intervalo de tempo para análise de anomalia. A análise de cada janela espectral determina a presença de anomalias. (Brooks, 2005) (Carl e Kesidis, 2006) (Lu e Ghorbani, 2009).

4.2.4 Métodos e Avaliações

As proposições de formas de detecção de anomalias em redes propostas em diferentes métodos, apresentam características de aplicação que podem ser descritas de acordo com o tipo de ataque DoS, tipo de método e técnica implementado.

Para cada tipo de método de acordo com o ataque é obtida uma avaliação de desempenho que caracteriza a capacidade destas técnicas em detectar DoS, e obter como resultado valores que indicam o percentual de certeza que o método e a técnica proporcionam, assim como o tempo médio de detecção em segundos para identificação do DoS.

A Tabela 1 é utilizada para apresentar estas relações, os métodos de detecção e suas avaliações propostas por (Brooks, 2005) (Carl e Kesidis, 2006) (Lu e Ghorbani, 2009) (Wang, Zhang e Shin, 2002) (Jung, Krishnamurthy e Rabinovich, 2001) (Moore, Voelker e Savage, 2001).

Tabela 1. Métodos de detecção DoS e suas avaliações

#	Tipo de Ataque	Método	Técnica	C_i	Tempo Detecção (segundos)
1	TCP SYN e Inundação TCP	Perfil de Atividade	Análises de Backscatter	0,9	1-5
2	Inundação ICMP, UDP e TCP SYN	Perfil de Atividade	Chi-Quadrado Detector de	1,0	1-5

			Entropia		
3	Inundação ICMP, UDP e TCP SYN	Perfil de Atividade	Detector de Entropia	1,0	1-5
4	TCP, UDP, e ICMP inundação pelo aumento linear abrupto	Mudança Sequencial de Pontos	Soma Cumulativa	0,94	De 1 a 36
5	TCP SYN com taxa constante de ataque	Mudança Sequencial de Pontos	Soma Cumulativa	0,7	12
6	Base de 39 anomalias + Inundação DoS	Análise de Wavelets	Transformada de wavelets	0,47	25
7	119 inundações DoS com intensidade de 4x, 7x e 10x.	Mudança Sequencial de Pontos	Soma Cumulativa	0,15	De 1 a 36
8	119 inundações DoS com intensidade de 4x, 7x e 10x.	Análise de Wavelets	Transformada de Wavelets + Soma Cumulativa	0,40	De 1 a 36

FONTE: (Brooks, 2005) (Carl e Kesidis, 2006) (Lu e Ghorbani, 2009) (Wang, Zhang e Shin, 2002) (Jung, Krishnamurthy e Rabinovich, 2001) (Moore, Voelker e Savage, 2001).

5. ARVDoS: ARQUITETURA PROPOSTA

Neste capítulo é apresentada a arquitetura proposta para o tratamento de DoS em redes virtualizadas.

5.1. DOMÍNIO DE APLICAÇÃO

O escopo da arquitetura reativa contra ataques DoS ARVDoS visa reduzir o impacto destes ataques sobre um alvo pela implementação de uma estratégia de resposta ao ataque, *Attack Response Strategy* (ASR) (Mirkovic e Reiher, 2004).

As metodologias ASR apresentam como principal característica reagir aos ataques pela detecção, com a melhor classificação de certeza possível, no menor tempo possível. Para que a arquitetura possa realizar suas atividades, algumas pré-condições devem ser atendidas.

5.2. PRÉ-REQUISITOS

O ambiente virtualizado de redes fornece os pré-requisitos para que a ARVDoS efetue suas atividades.

Estas pré-condições são compostas:

Pelo tráfego propriamente dito em que serão gerados o tráfego da rede e monitorados os fluxos DoS;

Pelo InP o qual fornece a infra estrutura física para configuração da rede virtual e

Pelo SP o qual executa atividades diretamente utilizadas pelos agentes na arquitetura. Estas atividades atribuídas ao Provedor de Serviço são utilizadas como entradas para as ações executadas pela arquitetura.

A Figura 5.1 demonstra a interação de atividades para o estabelecimento das pré-condições.

Na composição dos pré-requisitos a sequência de atividades que estabelece a interação entre seus elementos, inicia com as características topológicas e de recursos de rede físicos oferecidos pelo Provedor de Infraestrutura, InP, (e.g. equipamentos, enlaces, NICs). Estes recursos fornecem ao Provedor de Serviços, SP, as condições de criação e configuração da rede virtual, que da mesma forma

terá componentes como roteadores virtuais, VNICs e SLA de usuário. Desta forma é estabelecida a topologia virtual.

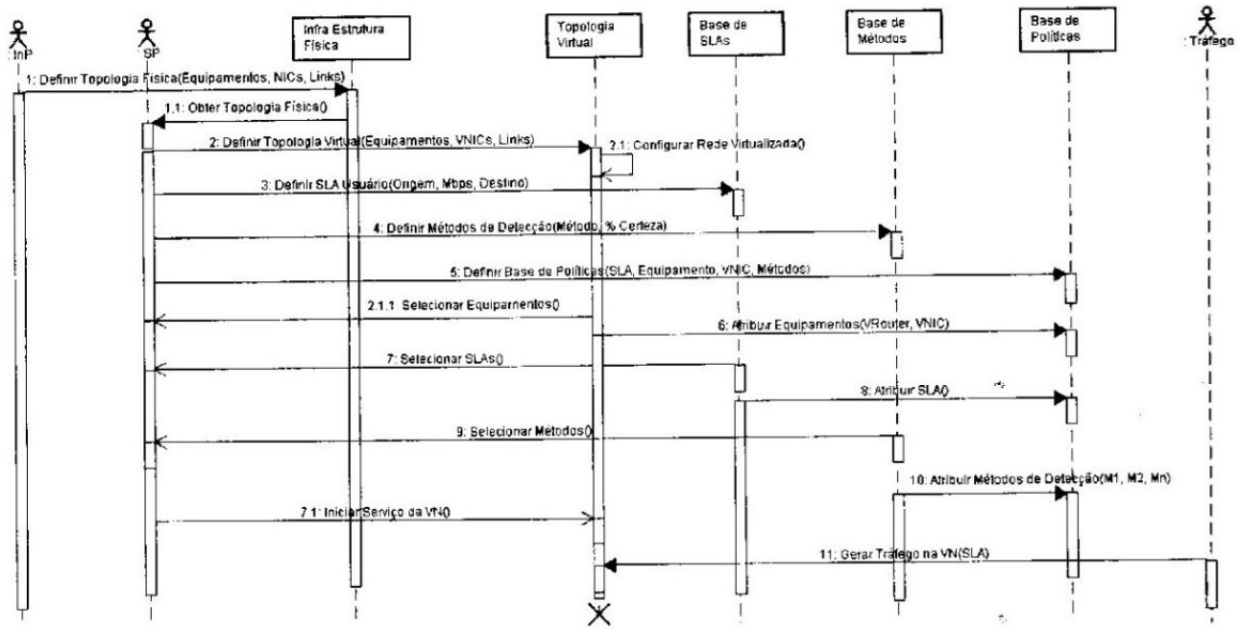


Figura 5.1. Pré-condições de implementação para ARVDoS

O SP permite:

i. Definir a topologia virtual;
ii. Definir o acordo de nível de serviço (SLA) de usuário;
iii. Definir a base de políticas e
iv. Atribuir métodos de detecção.

A topologia virtual é definida em acordo com a necessidade de nós para comunicação entre os clientes e o serviço solicitado. Nesta topologia serão instalados e configurados roteadores, adaptadores de redes, switches e enlaces virtualizados.

Sobre esta topologia virtual é definida e armazenada, em uma base, as SLAs de usuário de forma a considerar um identificador (ID), a origem ou nó de origem (ORIGEM), a banda passante nominal contratada e medida em megabits por segundo (Mbps) (BANDA) e o destino ou nó de destino (DESTINO) para uma SLA de usuário.

Da mesma forma uma base de métodos de detecção é necessária. É importante ressaltar que a arquitetura é independente de qualquer método de

detecção, permite sim que sejam atribuídos diversos métodos, estes armazenados em uma base de métodos em que possam ser instanciados a fim de conhecer o percentual de certeza de um ataque de negação de serviço por uma ponderação entre os valores de retorno dos métodos escolhidos para uma política a ser utilizada pela ARVDoS. A definição dos métodos é dada por um identificador (ID), o nome do método (MÉTODO) e o percentual de certeza de detecção DoS (CERTEZA) fornecido pela implementação de um método.

Para que a arquitetura possa utilizar as tuplas armazenadas na base de SLAs e na base de métodos é estabelecida uma Base de Políticas, na qual seus valores relacionais serão consultados pelo agente *mIdentifier*. Para obter as entradas de dados necessárias as atividades deste agente as linhas da base de políticas são definidas pelas colunas identificador (ID), identificador da SLA na Base de SLAS (SLA), equipamento a ser configurada a SLA (EQUIPAMENTO), interface de rede a qual será atribuída a SLA (INTERFACE) e o identificador da Base de Métodos em que serão instanciados para detecção (MÉTODOS).

A Figura 5.2 apresenta o relacionamento entre a Base de Políticas com a Base de SLAs e a Base de Métodos, estes relacionamentos permitem a parametrização de políticas da ARVDoS.

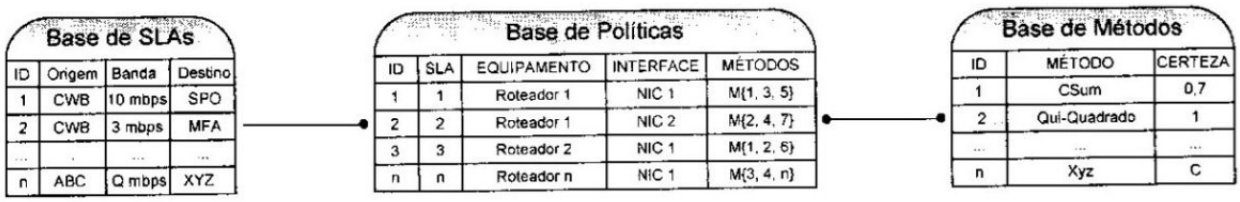


Figura 5.2. Relacionamento entre as bases de parametrização da ARVDoS

5.3. ATIVIDADES

A fim de distinguir e determinar qual tipo de atividade reativa a arquitetura deve acionar classificou-se o DoS em três classes de reação de acordo com o percentual de certeza (PC) obtido pela ponderação dos valores de retorno de cada método instanciado:

i) NDoS: não há DoS, esta situação ocorre quando $PC = 0$;

ii) IDoS: há incerteza de DoS, esta situação ocorre quando $0 < PC < 1$ e

iii) CDoS: há DoS, esta situação ocorre quando $PC = 1$.

A obtenção do PC é dado segundo a equação 1, onde β_i é a ponderação do resultado da instanciação de um método pelo agente *mIdentifier*, i determina os métodos a partir da base de políticas e n é a quantidade de métodos instanciados.

$$PC = \sum_{i=1}^{n} \beta_i, \forall i \geq 1 \quad (1)$$

O cálculo de β_i é dado segundo a equação 2, onde M é o resultado do método *i* obtido durante o monitoramento do tráfego quando é instanciado pelo *mIdentifier*, e C_i é uma constante que representa o grau de confiança do método *i*.

$$\beta_i = \frac{(M_i * C_i)}{\sum_{j=1}^{n} C_j} \quad (2)$$

O Quadro 5.1 apresenta um fragmento do pseudocódigo para implementação do agente *mIdentifier*.

```
calcular beta()
inicio
      para j <- 0 ate n faca
            ler C[j];
            somaC <- somaC + C[j];
      fim para

      //obter o valor de beta
      para i <- 0 ate n faca
            ler M[i];
            ler C[i];
            beta[i] <- (M[i]*C[i])/somaC;
      fim para
fim
```

Quadro 5.1. Pseudocódigo para calcular β (*mIdentifier*).

Para responder ao ataque DoS a arquitetura propõe uma nova configuração para a topologia da rede virtualizada, separando o fluxo DoS da rede através da criação de um caminho alternativo a fim de isolar o tráfego de acordo com a SLA do usuário e o PC estabelecido

A ARVDoS é independente de tecnologia de virtualização ou de método de detecção, permite sim sua aplicação em diferentes tecnologias com a utilização de

um ou mais métodos em sua estrutura, desta forma sua composição conta com três agentes, *mIdentifier, mRemaker e mInstaller.*

mIdentifier: executa atividades para acionar a análise do tráfego pela consulta à Base de Políticas, assim como é responsável por efetuar o cálculo de β. O resultado deste valor será atualizado, conforme parametrização neste agente de acordo com o método de detecção instanciado.

O Quadro 5.2 apresenta um fragmento do pseudocódigo para implementação do agente *mIdentifier.*

```
mIdentifier
inicio
      ler politicas(id,sla, equipamento, interface, metodos[i]);
      enquanto i <= n
            ler metodos(id, metodo, certeza);

      ler slas(id, origem, banda, destino);

      instanciar metodos[i];
            retornar percentual M[i];

      calcular beta();

      ler endereco_origem;
      ler endereco_destino;
fim
```

Quadro 5.2. Pseudocódigo agente *mIdentifier*

mRemaker – executa atividades de acionar o tratamento quando $0 < PC < 1$ ou $PC > 0$ do tráfego assim como cancelar o tratamento quando o percentual de certeza retorna a zero, $PC = 0$. O Quadro 5.3 apresenta um fragmento do pseudocódigo para implementação do agente *mRemaker.*

```
//obter o valor de PC
calcular PC()
inicio
      para i <- 0 ate n faca
            PC <- PC + beta[i];
      se PC = 1
            redirecionar null();

      se PC > 0 e PC < 1
            calcular enlace();
            adicionar rota();

      se PC = 0
            calcular enlace();
            excluir rota();
fim
```

Quadro 5.3. Pseudocódigo agente *mRemaker* para calculo de PC

Assim que a condição de acionar tratamento for ativada o *mRemaker* tem a função de configurar a tabela de roteamento a fim de executar duas ações específicas de reação: redirecionar o tráfego para o buraco negro sem interferir na SLA contratada pelo usuário ou redirecionar para o caminho alternativo criado pela ARVDoS.

Quando o tráfego é redirecionado para o caminho alternativo o *mRemaker* executa a reconfiguração da banda passante em dois domínios, no enlace alternativo e no enlace da SLA.

Toda vez que $0 < PC < 1$ será realizada a redução da SLA a partir do PC em duas variáveis que denominamos: Enlace reativo (LR) e Enlace real (LS). A obtenção de LR é dada pela equação 3, onde BW é o valor em megabits por segundo (Mbps) de um fluxo identificado como IDoS ou CDoS na SLA do usuário contratada para a rede virtualizada. A obtenção de LS desta forma é dada pela equação 4, onde a SLA é o valor total em Mbps no enlace contratado.

$$LR = BW * PC \quad (3)$$

$$LS = SLA - LR \quad (4)$$

Quando a condição de cancelar tratamento for ativada, $PC = 0$, o *mRemaker* deve zerar o valor de LR e reintegrar o valor de LS, desta forma $LS = BW$. O Quadro 5.4 apresenta um fragmento do pseudocódigo para implementação do agente *mRemaker* no que se refere ao cálculo de LS e LR.

```
calcular enlace()
inicio
      receber PC;
      se PC > 0 e PC < 1
            receber BW;
            receber PC;

            LR <- BW * PC;
            alterar_LR(LR);

            LS <- SLA - LR;
            alterar_LS(LS);

      senao
            se PC = 0
                  LS <- BW;
                  LR <- { };
      fim se
fim
```

Quadro 5.4. Pseudocódigo para o cálculo de LS e LR (*mRemaker*)

mInstaller – a execução das atividades deste agente depende de uma mensagem enviada pelo agente *mRemaker* referente às condições de acionar tratamento ou cancelar tratamento.

O *mInstaller* irá instalar e configurar os equipamentos virtuais (VNICs e Roteadores), suas tabelas de roteamento e enlace reativo de acordo com o valor obtido de LR enviado pelo agente *mRemaker*. A instalação dos equipamentos e sua configuração são dependentes da tecnologia de virtualização utilizada.

Os Quadros 5.5, 5.6, 5.7, 5.8 e 5.9 apresentam fragmento do pseudocódigo para implementação do agente *mInstaller* no que se refere às configurações de manipulação de rota, enlace reativos.

```
adicionar rota()
      inicio
            criar tabela_de_rota<n>;
                  adicionar ()
                        definir endereco_origem;
                        definir endereco_destino;
                        definir proximo_salto;

fim
```

Quadro 5.5. Pseudocódigo para adicionar rota (*mInstaller)*

```
excluir rota()
inicio
      definir tabela_de rota<n>;
            excluir()
                  definir endereco_origem;
fim
```

Quadro 5.6. Pseudocódigo para excluir rota (*mInstaller)*

```
redirecionar null()
inicio
      definir tabela_de_rota<n>;
            adicionar();
                  definir endereco_null;
                        definir endereco_origem;
fim
```

Quadro 5.7. Pseudocódigo para direcionar tráfego para /null (*mInstaller)*

```
alterar LS();
inicio
      criar regra_principal;
      definir interface;
      definir filtro <- protocolo;
      reduzir enlace_sla()
            definir taxa <- LS;
            definir limite_fila <- taxa;
fim
```

Quadro 5.8. Pseudocódigo alterar enlace com valor de LS (*mInstaller)*

```
alterar LR();
inicio
      criar regra_principal
      definir interface
      definir filtro <- protocolo;
      reduzir enlace_sla()
            definir taxa <- LR;
            definir limite_fila <- taxa;
fim
```

Quadro 5.9. Pseudocódigo alterar enlace com valor de LR (*mInstaller)*

Na Figura 5.3 é possível observar as atividades dos agentes *mIdentifier, mRemaker e mInstaller* representadas pela sequência de ações apresentadas quando IDoS é detectado.

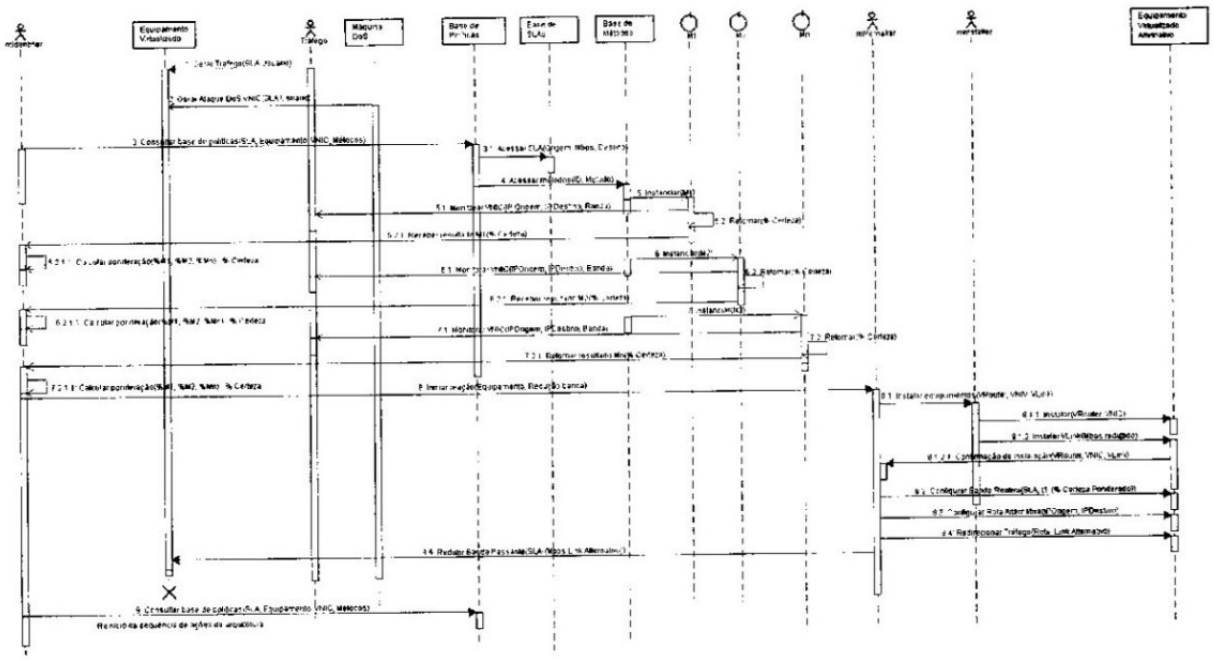


Figura 5.3. Atividades dos agentes para IDoS

Na Figura 5.4 é possível observar as atividades dos agentes *mIdentifier, mRemaker e mInstaller* representadas pela sequência de ações apresentadas quando CDoS é detectado.

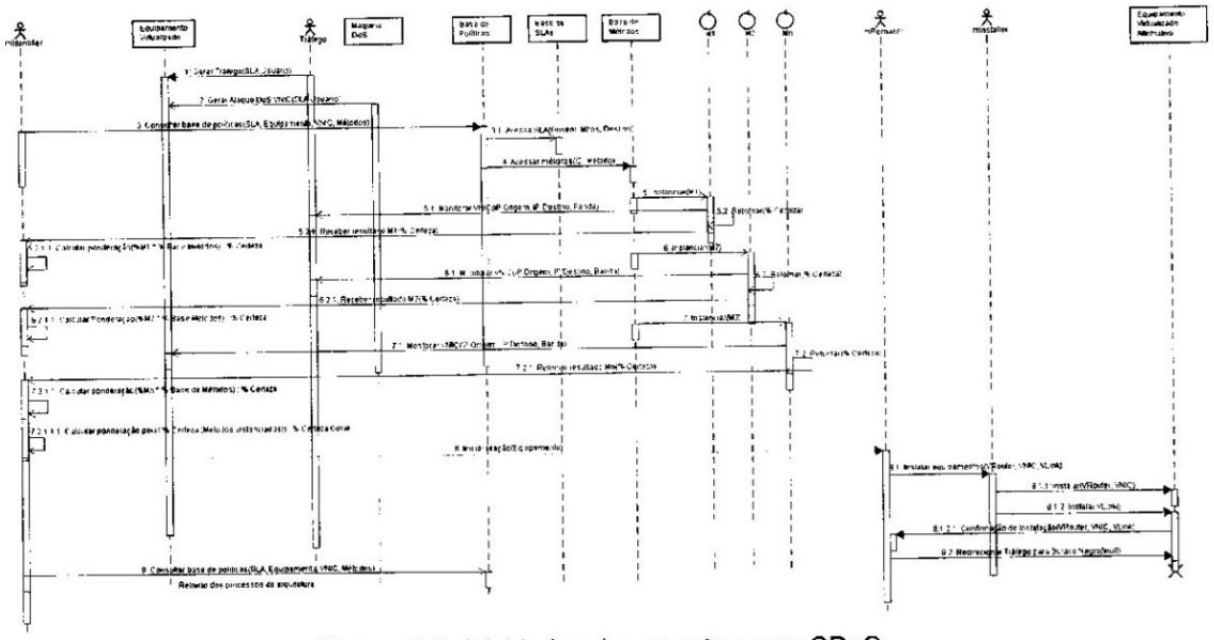


Figura 5.4. Atividades dos agentes para CDoS

Na Figura 5.5 é possível observar as atividades dos agentes *mIdentifier, mRemaker e mInstaller* representadas pela sequência de ações apresentadas quando NDoS é detectado.

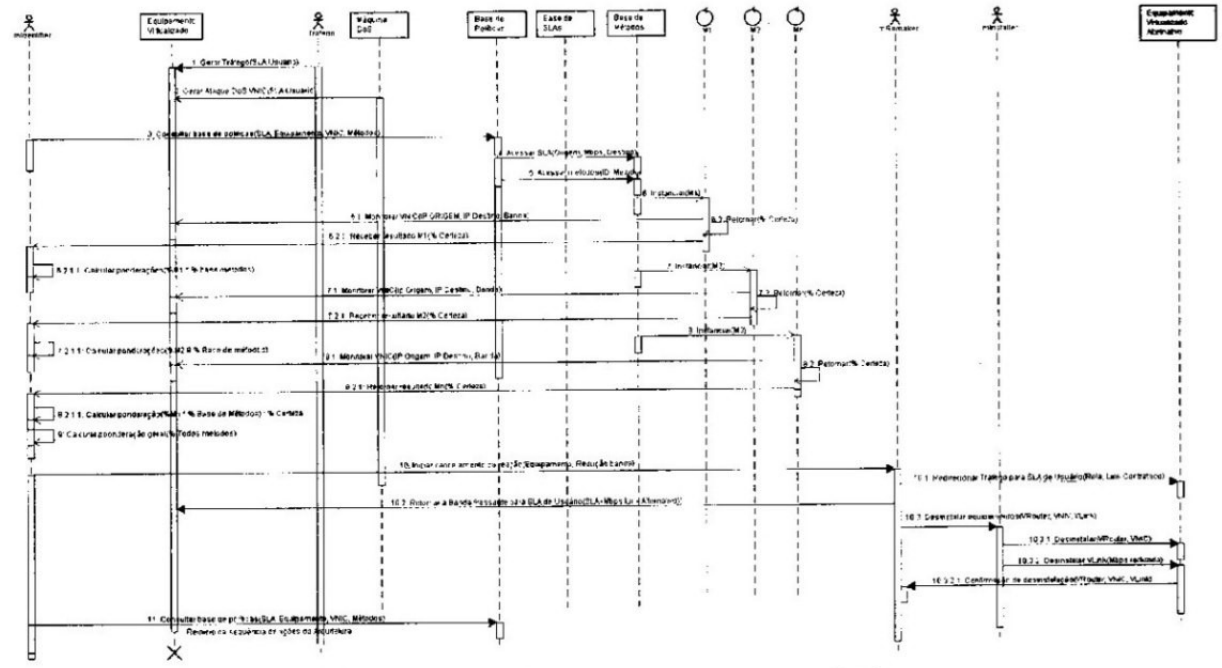


Figura 5.5. Atividades dos agentes para NDoS

Para identificar as reações, NDoS, IDoS e CDoS, a arquitetura utiliza as nomenclaturas ARVDoS 1 (para a primeira reação), ARVDoS 2 (para a segunda reação), ARVDoS 3 (para a terceira reação) e ARVDoS 4 (para a quarta reação).

5.4 CLASSIFICAÇÃO DAS REAÇÕES DA ARVDoS

A reação ARVDoS 1 representa atividades de instalação e configuração dos equipamentos virtuais (VNICs, VRouters, VEnlaces, tabelas de roteamento) assim que o valor de PC retornar uma incerteza de ataque ou uma certeza de ataque.

Com a reação ARVDoS 2 são disparadas as atividades de reconfiguração de fila no enlace virtual (VEnlace) quando o valor de PC é alterado no caso de uma incerteza de ataque, esta alteração pode ser para mais ou para menos.

A reação ARVDoS 3 representa o encerramento do tratamento com o retorno da topologia à sua configuração original (Rotas e VEnlaces), ocorre quando o valor de PC retornado é zero (0) após a identificação e aplicação de tratamento pela arquitetura de uma certeza de DoS ou incerteza de DoS.

A reação ARVDoS 4 possibilita o encaminhamento para o buraco negro, o repasse para um endereço de destino /null de todo ataque identificado pelo PC como certeza de DoS.

Para ilustrar esta caracterização de classificação das reações da arquitetura é apresentada na Figura 5.6.

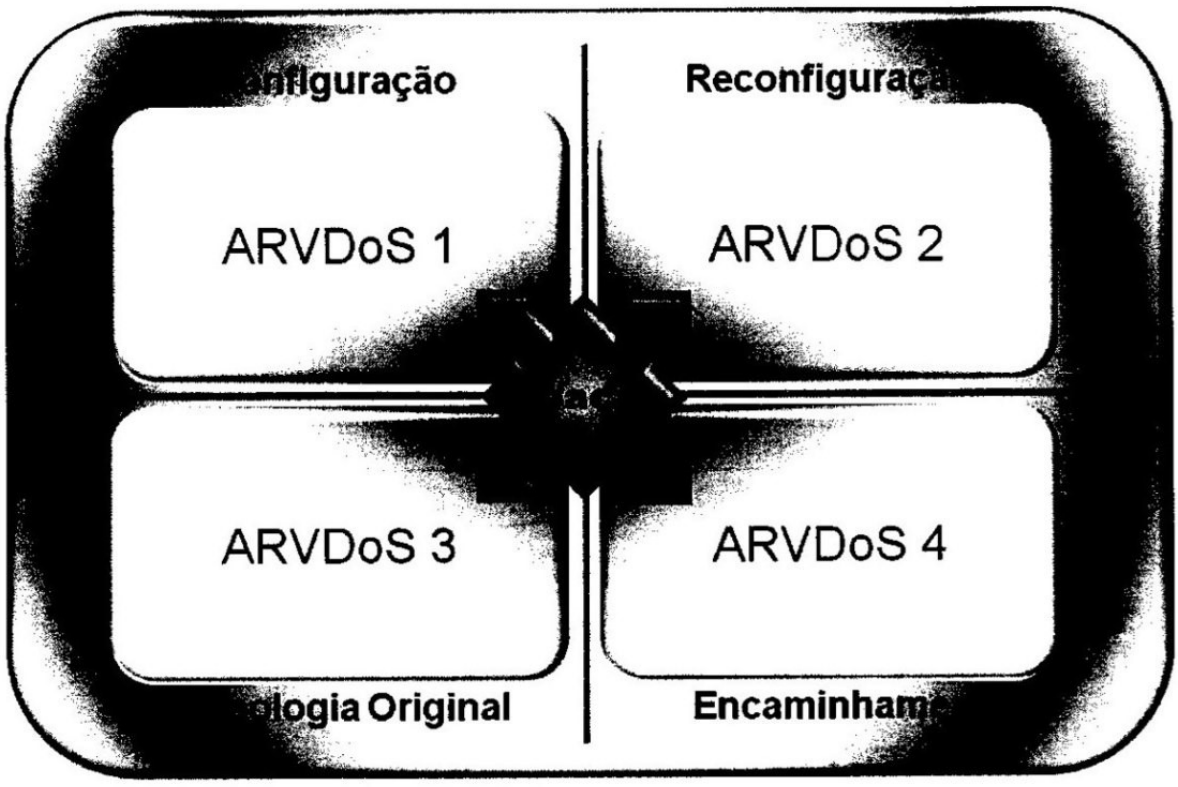


Figura 5.6. Classificação das Reações da ARVDos

A ARVDoS apresenta uma característica importante em sua implementação, ser escalar independente da quantidade de nós que compõe a topologia da rede virtual. As atividades da arquitetura são aplicadas no local em que a detecção foi identificada, assim os procedimentos executados pelos agentes *mRemaker* e *mInstaller* sempre acontecem entre os nós roteadores que formam o enlace da SLA do usuário. A Figura 5.7 ilustra três topologias diferentes: Topologia A com dois (2) nós roteadores, Topologia B com três (3) nós roteadores e a Topologia C com quatro (4) nós roteadores entre a rota do usuário para exemplificação da escalabilidade da ARVDoS .

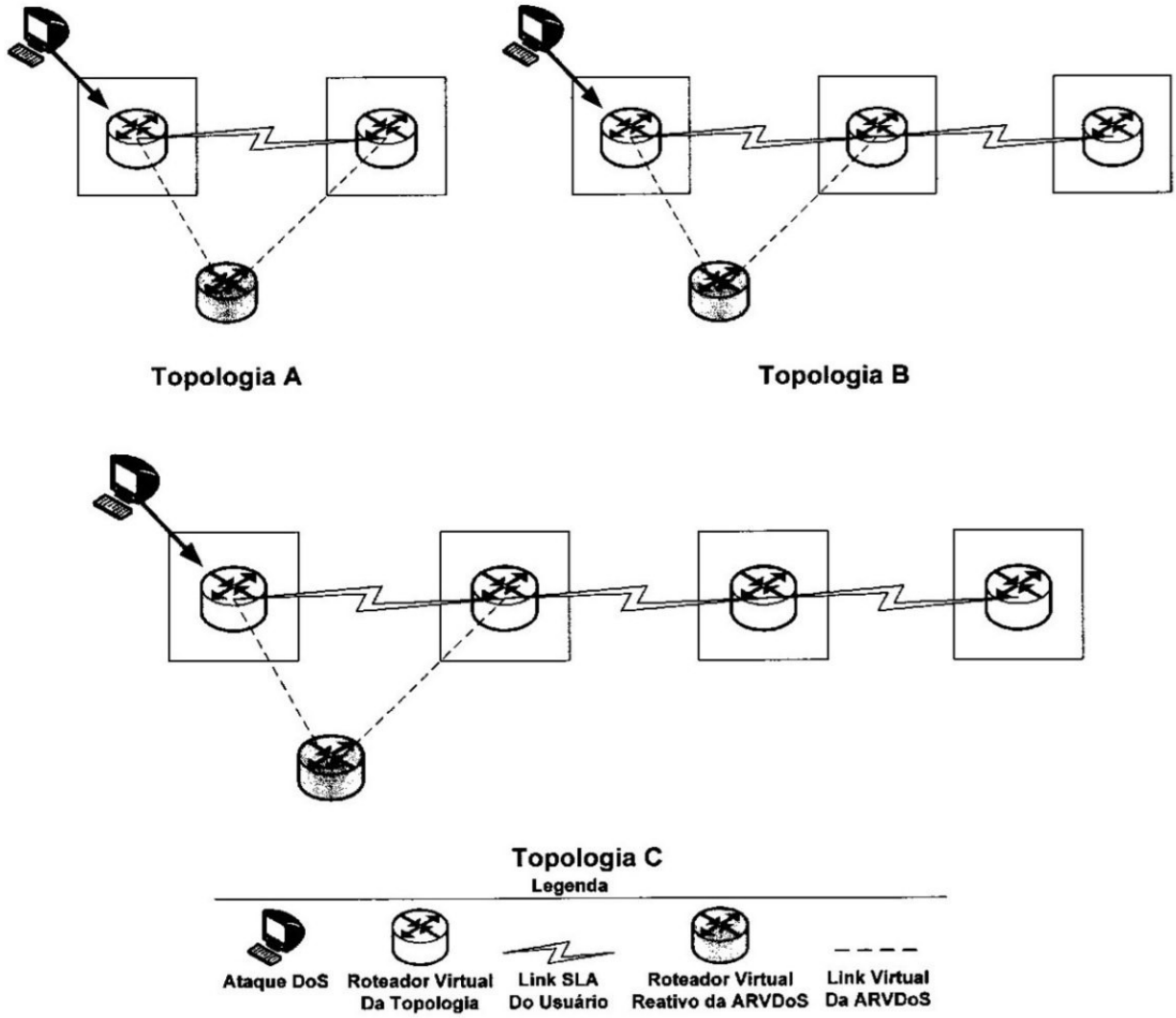


Figura 5.7. Escalabilidade da ARVDoS

Os agentes da arquitetura podem ser ativados remotamente a partir de uma máquina de agentes que irá conectar a interface do equipamento virtualizado ou é executada diretamente no equipamento virtual.

Como forma de visualização geral da interação entre os agentes da arquitetura e os pré requisitos determinados pelo InP e SP é apresentada a Figura 5.8.

A notação UML utilizada para representar de forma holística a arquitetura ARVDoS é um diagrama de Casos de Usos. Nesta diagramação os atores (bonecos de palito) representam os agentes de pré-requisitos da arquitetura (InP, SP, Tráfego), assim como os agentes da arquitetura (*mIdentifier, mRemaker, mInstaller*).

Da mesma forma os usos (elipses do diagrama), demonstram as atividades da ARVDoS, bem como os seus relacionamentos (setas continuas direcionais) com os agentes. Estes relacionamentos entre agentes e atividades geram dependências (setas seccionadas direcionadas à dependência) de existência ou execução de uma atividade. A ARVDoS desta forma apresenta características hierárquicas.

As seguintes atividades são modeladas para definição dos pré-requisitos de implementação da ARVDoS:

1) **Vinculadas ao Tráfego:** Gerar tráfego na rede virtual;
2) **Vinculadas ao InP:** Definir topologia física;
3) **Vinculadas ao SP:** Definir topologia virtual; Definir SLA de usuário; Definir base de políticas; Atribuir métodos de detecção;

As seguintes atividades são modeladas para definição das atividades dos agentes de implementação da ARVDoS:

1) **Vinculadas ao agente *mIdentifier*:** Analisar tráfego pela consulta à base de políticas e instanciação de métodos de detecção; Identificar IDoS; identificar NDoS; Identificar CDoS; Atualizar status do tráfego;
2) **Vinculadas ao agente *mRemaker*:** Verificar status do tráfego; Acionar tratamento com a configuração de rota para tratamento em enlace alternativo ou buraco negro; Acionar tratamento para reconfiguração de enlace reativo (LR), enlace de SLA (LS); Cancelar o tratamento;
3) **Vinculadas ao agente *mInstaller*:** Instalar recursos sejam eles roteadores virtuais, VNICs ou configuração de enlaces; desinstalar recursos; reconfigurar recursos.

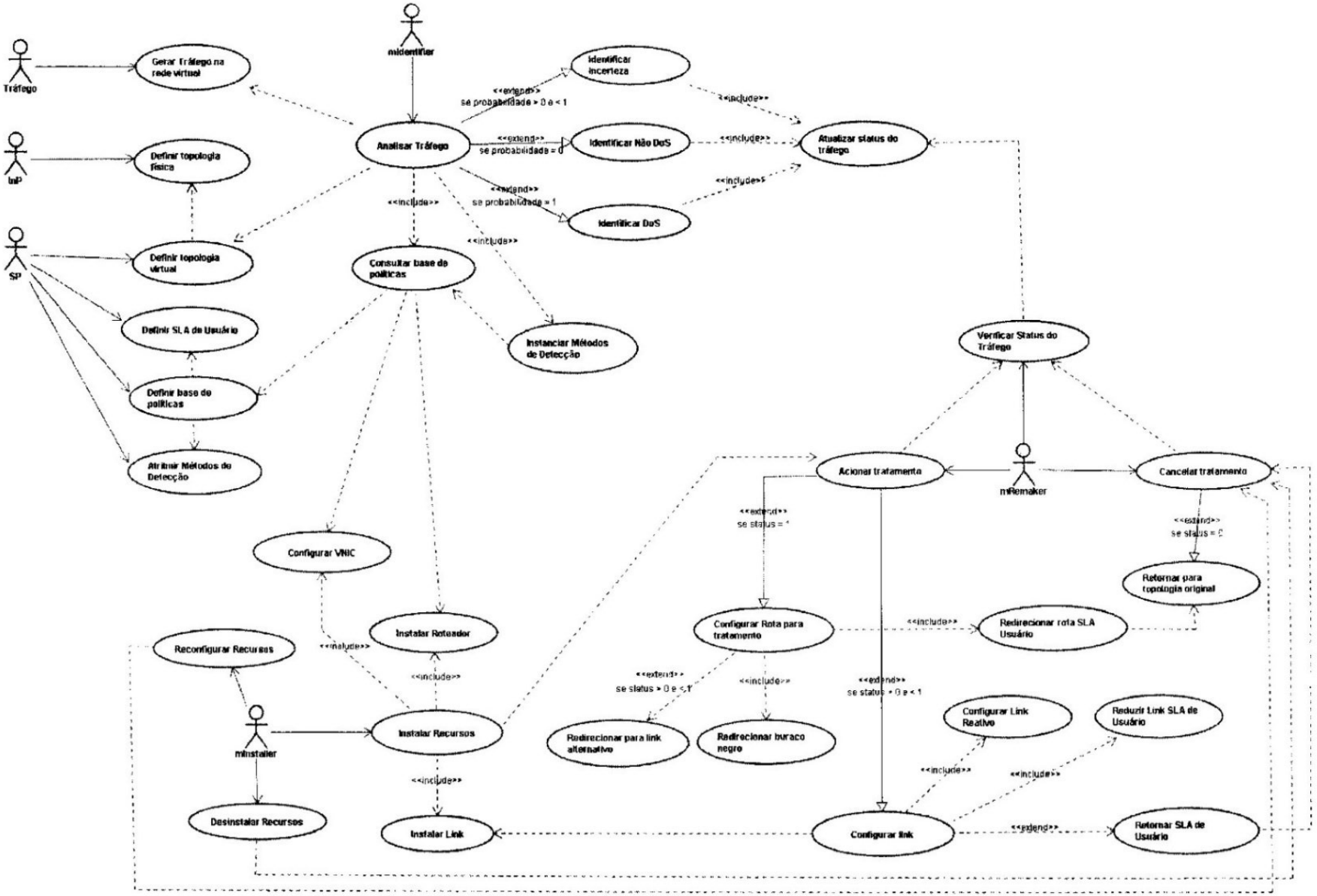


Figura 5.8. Visão geral da arquitetura ARVDoS

6. TESTES

Neste capítulo são apresentados os testes da arquitetura ARVDoS sobre os cenários de implementação desenvolvidos.

6.1. VIRTUALIZAÇÃO DE REDE

Para validação da ARVDoS serão coletados dados em uma VN real, construída sobre um InP em um laboratório de informática.

A arquitetura foi avaliada em computadores com CPUs Core 2 Duo 2.2 GHz, 1.0 GB de memória RAM. Para explorar o problema e possibilitar a configuração das redes o InP foi configurado da seguinte maneira:

- Configurar em cada computador do InP duas interfaces de redes conectados por cabos UTP categoria 5e, mantendo o padrão 568-B via enlace IEEE 802.3.

- Os computadores no InP foram utilizados com as plataformas de software Microsoft e GNU/Linux rodando Windows XP SP3 e Ubuntu 10.04 como Sistemas Operacionais (SO) hosts.

- Em cada SO foi instalado para configuração da VN uma máquina virtual (VM) com plataforma de software GNU/Linux rodando Fedora 13 em modo texto e modo gráfico.

- Os SO hospedados serão utilizados como roteadores virtuais (RV).

- A rede virtualizada é configurada, tendo como InP a rede física e seus nós roteadores com duas interfaces virtuais de rede (VNICs). A SLA de usuário é definida pelo SP em cada configuração de rede virtualizada.

- Para a virtualização das VMs utilizou-se três tecnologias:

    1. Virtual Box 3.2.6 (Oracle, 2010).

2. VMware Server 2.02 (VMware, 2010).

3. Xen Hypervisor 4.0 (Xen.org, 2010).

- As três tecnologias foram utilizadas nas máquinas virtuais GNU/Linux, enquanto nas VMs Microsoft® as tecnologias Virtual Box® e VMware®.

- A simulação dos clientes da rede assim como a simulação dos ataques DoS são realizadas utilizando o iperf (UCF, 2003) e o aplicativo hping3 (Sanfilippo, 2006) para gerar DoS por inundação TCP SYN. A origem dos ataques é dada pelos clientes apontando para a topologia virtual de rede. Nas simulações os testes decorrem por 300 segundos. Os ataques DoS para que apresentem uma resultante que indique sua ocorrência são disparados aleatoriamente.

- O mesmo teste foi realizado em cada tecnologia de virtualização, em duas situações de forma a verificar o tempo para reação da arquitetura e o desempenho de cada tecnologia utilizada.

  1º: Instalar e configurar todos os VNICs, RVs e enlaces a partir do valor do PC sem nenhuma configuração prévia definida.

  2º: Configurar a reação da arquitetura a partir do valor do PC com os RVs e VNICs pré-configurados, tendo somente a necessidade de iniciar o RV e alterar as tabelas de roteamento.

A partir desta metodologia, para realizar a avaliação da ARVDoS foram propostos quatro (4) cenários, os quais são apresentados nas seções 6.1.1 a 6.1.4.

6.1.1. Cenário I

Neste cenário são utilizados dois nós roteadores na composição da VN e a aplicação do ataque em um ponto VNIC A. O ataque considerado é um ataque do tipo IDoS. Posteriormente, após a identificação e tratamento do ataque é possível observar a aplicação do NDoS quando os métodos de detecção identificam que o percentual de certeza passa a ser zero ($PC = 0$).

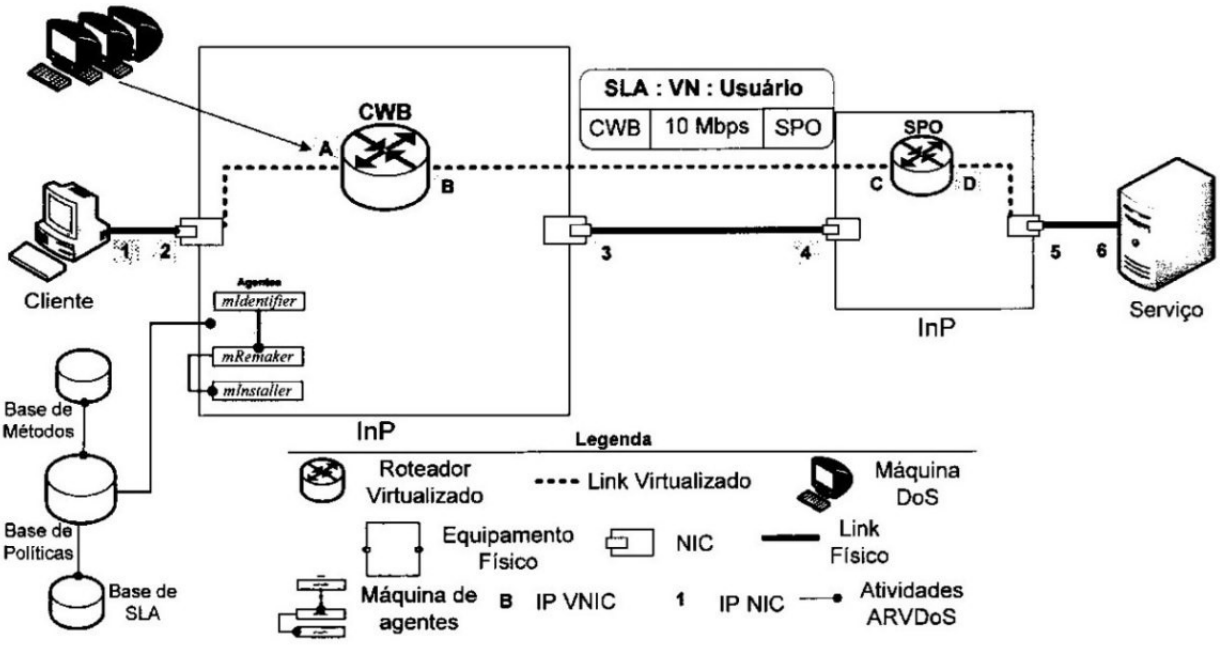


Figura 6.1. Cenário I de simulação InP e SP da rede virtual

A configuração de rede nas topologias do InP e da VN são apresentadas na Tabela 2. As configurações da VN são definidas como pré-requisito pelo SP.

Tabela 2. Configurações de endereçamento de rede do InP e da VN

InP		***VN***	
NIC	**IP/MÁSCARA**	**VNIC**	**IP/MÁSCARA**
1	192.168.13.10/24	1	192.168.13.10/24
2	192.168.13.1/24	A	192.168.13.2/24
3	200.27.16.13/30	B	201.10.15.1/30
4	200.27.16.14/30	C	201.10.15.2/30
5	192.19.9.1/24	D	192.19.9.2/24
6	192.19.9.10/24	6	192.19.9.10/24

As tabelas de roteamento para o roteador virtual CWB e para o roteador virtual SPO são apresentadas na Tabela 3 e Tabela 4.

Tabela 3. Tabela de roteamento do roteador virtual CWB

TABELA DE ROTEAMENTO CWB (Virtualizado)			
ID REDE DESTINO	MÁSCARA DESTINO	PRÓXIMO SALTO	INTERFACE
192.168.13.0	255.255.255.0	127.0.0.1	A
192.19.9.0	255.255.255.0	201.10.15.2	B
201.10.15.0	255.255.255.252	201.10.15.2	B

Tabela 4. Tabela de roteamento do roteador virtual SPO

TABELA DE ROTEAMENTO SPO (Virtualizado)			
ID REDE DESTINO	MÁSCARA DESTINO	PRÓXIMO SALTO	INTERFACE
192.168.13.0	255.255.255.0	201.10.15.1	C
201.10.15.0	255.255.255.252	201.10.15.1	C
192.19.9.0	255.255.255.0	127.0.0.1	D

O SP ainda define qual política será utilizada pelo *mIdentifier* para a VN do Cenário I, e pode ser observada pela Figura 6.2. São definidos pelo SP dois métodos de detecção existentes na base de métodos, assim como a SLA existente na base de SLAs. Da mesma forma, define em qual equipamento e qual interface deste equipamento será realizado o monitoramento. A política com $\mathrm{ID} = 1$ é utilizada neste caso.

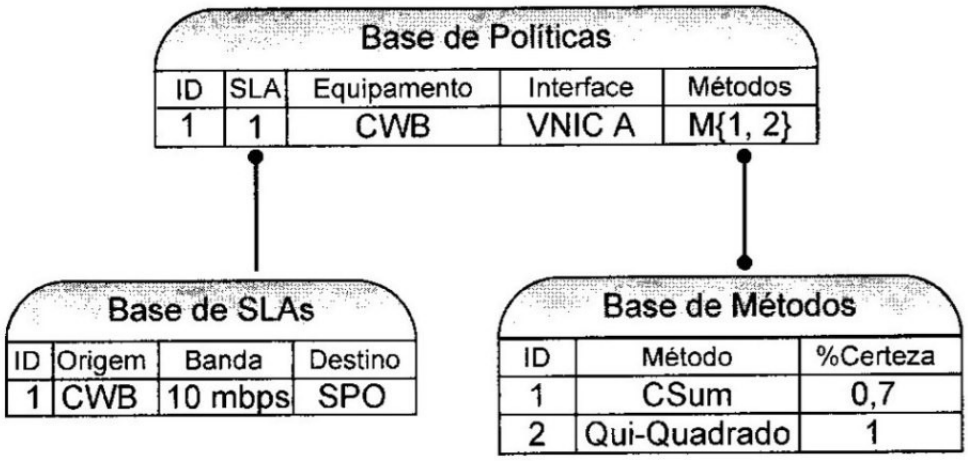


Figura 6.2. Base de políticas para o Cenário I

A máquina de agentes da ARVDoS é executada em paralelo no SP da VN. O agente *mIdentifier* passa a executar suas atividades, instanciando o método de detecção no VNIC de acesso do roteador virtual CWB, assim com o tráfego da rede ativo, é disparado um ataque pela máquina DoS no VNIC A.

Com a detecção de IDoS o agente *mIdentifier* envia uma mensagem ao agente *mRemaker* com endereço IP de origem do ataque, IP destino e resposta do(s) método(s) instanciado(s), M_i. Esta mensagem é enviada de acordo com o retorno do(s) método(s) de detecção instanciado(s), este retorno pode ocorrer em tempos diferentes. A Tabela 5 apresenta os dados desta mensagem.

Tabela 5. Mensagem de configuração para do *mIndentifier* para *mRemaker* para o Cenário I

Mensagem	IP ORIGEM	IP DESTINO	M_i	
			Chi-quadrado	CSUM
1	192.168.13.10	192.19.9.10	$M_2 = 0{,}85$	$M_1 = 0$
2	192.168.13.10	192.19.9.10	$M_2 = 0{,}85$	$M_1 = 0{,}37$
3	192.168.13.10	192.19.9.10	$M_2 = 0$	$M_1 = 0$

Com estes dados o agente *mRemaker envia uma mensagem* de autorização *ao agente mInstaller* a fim de executar a instalação e a configuração do Roteador Virtualizado da Arquitetura (RVA) e sua tabela de roteamento assim como a instalação e configuração das interfaces de rede deste equipamento. A Tabela 6 apresenta a configuração dos VNICs instalados e a Tabela 7 apresenta a tabela de roteamento para RVA.

Tabela 6. Configuração dos VNICs da VN reativa

VN REATIVA	
VNIC	**IP/MÁSCARA**
E	201.10.15.5/30
F	201.10.15.6/30
G	201.10.15.9/30
H	201.10.15.10/30

Tabela 7. Tabela de roteamento do RVA

TABELA DE ROTEAMENTO RVA			
ID REDE DESTINO	MÁSCARA DESTINO	PRÓXIMO SALTO	INTERFACE
192.168.13.0	255.255.255.0	201.10.15.5	F
201.10.15.4	255.255.255.252	201.10.15.5	F
201.10.15.8	255.255.255.252	201.10.15.10	G
192.19.9.0	255.255.255.0	201.10.15.10	G

O agente *mRemaker* estabelece uma sessão remota com o RV CWB e RV SPO para alteração e configuração dos VNICs e de suas tabelas de roteamento. A Tabela 8 e Tabela 9 mostram as rotas acrescentadas em RV CWB e RV SPO.

Tabela 8. Rotas adicionadas ao RV CWB após reação

TABELA DE ROTEAMENTO CWB			
ID REDE DESTINO	MÁSCARA DESTINO	PRÓXIMO SALTO	INTERFACE
201.10.15.4	255.255.255.252	201.10.15.6	E

201.10.15.8	255.255.255.252	201.10.15.6	E

Tabela 9. Rotas adicionadas ao RV SPO após reação

TABELA DE ROTEAMENTO SPO			
ID REDE DESTINO	MÁSCARA DESTINO	PRÓXIMO SALTO	INTERFACE
201.10.15.4	255.255.255.252	201.10.15.9	H
201.10.15.8	255.255.255.252	201.10.15.9	H

Com as configurações de endereçamento e repasse finalizadas o agente *mRemaker* realiza o cálculo do PC de acordo com os valores de β. Com o resultado do PC, *mRemaker* calcula e aplica o resultado de LS ao enlace da SLA contratada e o resultado de LR aos enlaces entre CWB → RVA e entre RVA → SPO. O resultado obtido tanto para LR quanto para LS indicam o valor máximo que cada enlace pode assumir, $max(LR)$ e $max(LS)$. Esta redução da banda resulta em uma fila de descartes aos pacotes IDoS no RV RVA que são enviados à taxa de ataque medida por um enlace com banda reduzida, desta forma reduzindo o impacto do ataque de negação de serviço de forma a conduzir o fluxo a NDoS.

A Tabela 10 apresenta os cálculos realizados pelo agente *mRemaker* para obtenção dos valores de reconfiguração do enlace entre CWB → RVA e a Tabela 11 para reconfiguração do enlace entre RVA → SPO.

Tabela 10. Valores de configuração de enlace entre CWB e RVA

Mensagem	VNIC	PERCENTUAL DE CERTEZA			LS (Mbps)	LR (Mbps)
		β		PC		
		Chi-quadrado	CSUM			
1	E	0,85	0	0,5	5,5	4,5
2	E	0,85	0,37	0,65	6,4	3,6

Tabela 11. Valores de configuração de enlace entre RVA e SPO

Mensagem	VNIC	PERCENTUAL DE CERTEZA			LS (Mbps)	LR (Mbps)
		β		PC		
		Chi-quadrado	CSUM			
1	G	0,85	0	0,5	5,5	4,5
2	G	0,85	0,37	0,65	6,4	3,6

A Figura 6.3 apresenta a reconfiguração da topologia gerada pela ARVDoS quando IDoS é identificada por dois métodos instanciados. Esta referencia tem relação à mensagem dois (2) apresentada na Tabela 10 e Tabela 11.

A Figura 6.4 apresenta o gráfico de comportamento da VN no Cenário I para as atividades da ARVDoS.

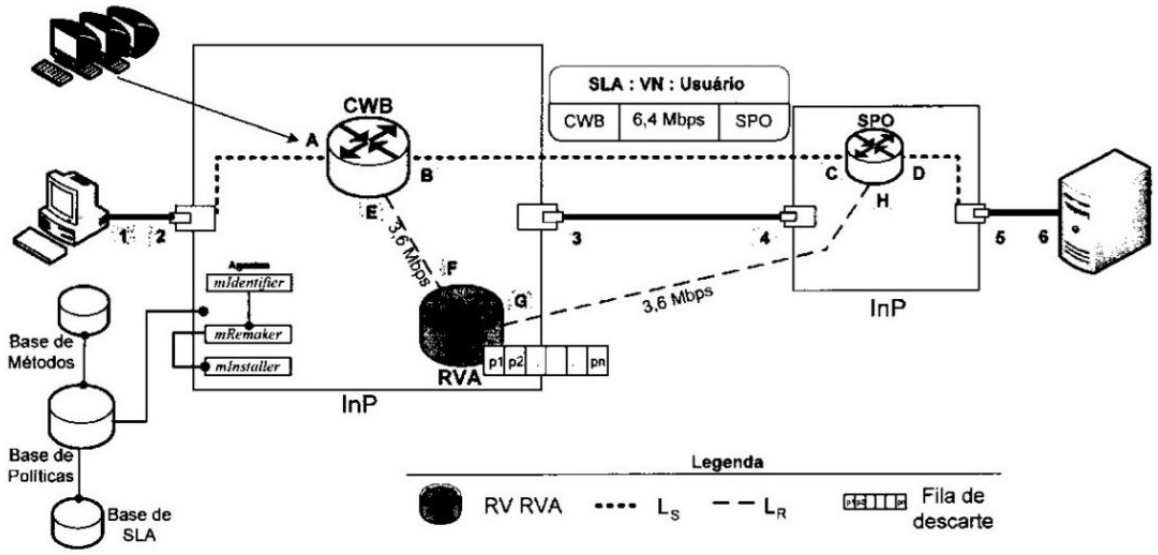


Figura 6.3. Topologia após a reação da ARVDoS para IDoS para o Cenário I

O comportamento do enlace no Cenário I para as atividades da arquitetura é representado pela Figura 6.4. É possível observar as variações do tráfego em diferentes situações em uma linha de tempo de trezentos segundos (300 s) em relação à SLA contratado pelo usuário dez (10) Mbps.

Tais situações destacadas no gráfico e detalhadas na Tabela 12 ilustram o tráfego antes do ataque, o instante em que uma situação de DoS se configura, o tempo de atraso (delay) que o método de detecção leva para identificar o ataque, o tempo de atraso da ARVDoS para aplicar suas atividades, as alterações do enlace quando um novo valor de LR ou LS são calculados, assim como o encerramento da reação quando é verificado NDoS.

Tabela 12. Comportamento do enlace na topologia para o Cenário I

#	INTERVALO DE TEMPO (s)	ATIVIDADE ARVDoS	MÉDIA (Mbps)	max(LR) (Mbps)	max(LS) (Mbps)
1	1-25	Tráfego sem detecção de anomalia	4,46	-	10
2	26-31	Início do DoS e atraso (delay) do método Chi-quadro em identificar a IDoS	8,16	-	10
3	32-41	Atraso (delay) da ARVDoS para realizar as atividades dos agentes *mIdentifier, mRemaker e mInstaller*	8,36	-	10

4	42-55	$PC = 0{,}5$ reação da ARVDoS com a criação do RVA e suas configurações e criação da fila de descarte com a aplicação do valor de LS nos VNICs A e B do RV CWB.	3,39	-	5,5
5	42-55	$PC = 0{,}5$ reação da ARVDoS com a criação entre CWB e RVA da fila de descarte com a aplicação do valor de LR nos VNICs E e G dos RVs CWB e RVA.	3,7	4,5	5,5
6	56-58	Atraso (delay) da ARVDoS para reconfiguração da fila de descarte e reconfigurar os valores de $LR\ e\ LS$ para realizar as atividades dos agentes *mIdentifier, mRemaker* devido ao tempo de detecção do método CSUM.	3,68	4,5	5,5
7	59-117	$PC = 0{,}65$, reação da ARVDoS com reconfiguração dos valores de LS.	4,9	-	6,4
8	59-117	$PC = 0{,}65$, reação da ARVDoS com reconfiguração dos valores de LR.	4,9	3,6	6,4
9	118-123	Atraso (delay) da ARVDoS para reconfiguração da VN do valor de $LR\ e\ LS$ quando NDoS é identificado pelos métodos instanciados. Os agentes *mIdentifier, mRemaker e mInstaller* desfazem as configurações reativas e retornam a topologia original.	5,12	3,6	6,4
10	123	$PC = 0$, NDoS identificado, o valor de LR é eliminado devido ao *mInstaller* ter desinstalado os VRs	-	-	10
11	124-300	$PC = 0$, o valor de LS é reconfigurado nos VNICs A e B do RV CWB para o valor da SLA. O tráfego	5,14	-	10

volta a comportar-se sem anomalias.

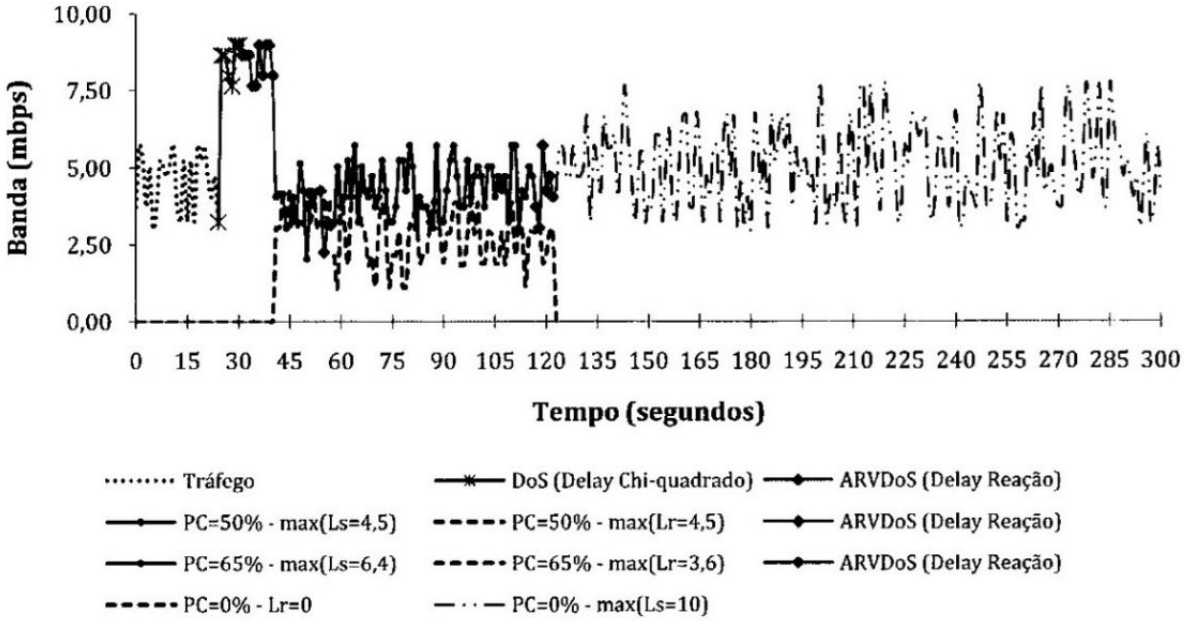


Figura 6.4. Comportamento do enlace na topologia para o Cenário I

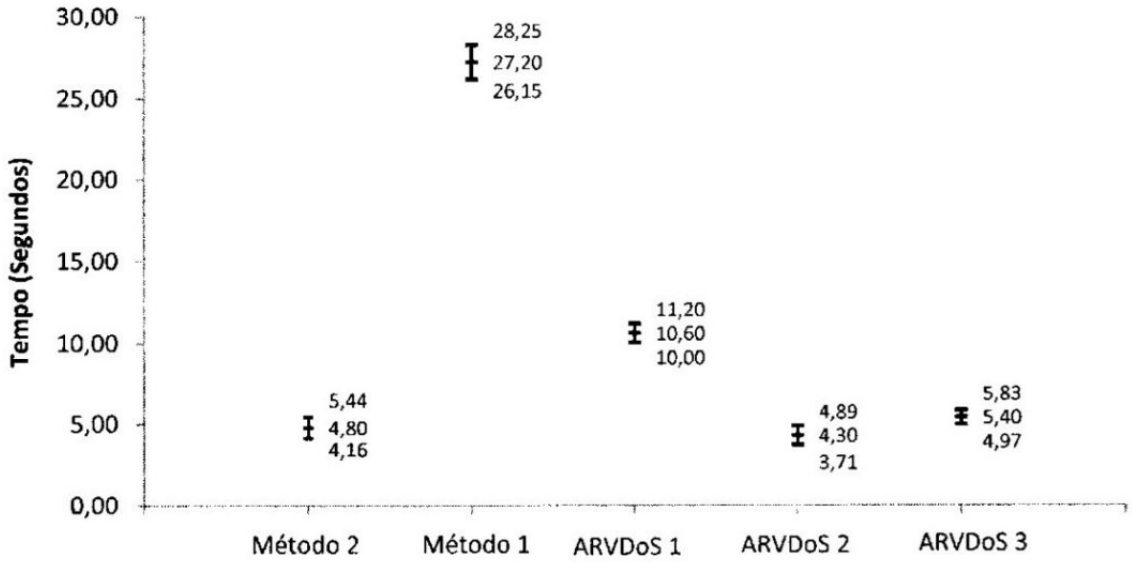
Para realizar uma observação do desempenho da ARVDoS é interessante observar o intervalo de confiança de forma a comparar os tempos de identificação da anomalia pelos métodos de detecção utilizados, assim como os tempos de reação da arquitetura.

O nível de confiança desejado é 95% (noventa e cinco por cento) ou 0,05. A amostra utilizada contém dez (10) avaliações do Cenário I, com a qual é obtida a média e o desvio padrão em segundos. O intervalo de confiança é calculado para os métodos de detecção utilizados no cenário e para os tempos gerados pelas reações da arquitetura aos percentuais de certeza obtidos.

A Tabela 13 e a Figura 6.5 apresentam estes resultados.

Tabela 13. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário I

Nível de Confiança = 95% (0,05)					
Atividade	**+**	**Média**	**-**	**Intervalo de Confiança**	**Amostras**
Método 2	5,44	4,8	4,16	0,64	10
Método 1	28,25	27,2	26,15	1,05	10
ARVDoS 1	11,2	10,6	10,0	0,6	10
ARVDoS 2	4,89	4,3	3,71	0,59	10
ARVDoS 3	5,83	5,4	4,97	0,43	10



Para as avaliações apresentadas anteriormente foi utilizado como referencia para implementação na ARVDoS a tecnologia de Virtualização Xen Hypervisor 4.0 por apresentar um tempo aproximado de 40% menor para inicialização das VMs.

Tabela 14. Avaliação do tempo de reação em diferentes tecnologias para o Cenário I

#	Plataforma	Virtualização	SO Host	Tempo de Reação (s)	
				1º Ensaio	2ª Ensaio
1	Windows XP	Virtual Box 3.2.6	Fedora 13	36,764	14,414
2	Windows XP	VMware Server 2.02	Fedora 13	33,453	11,593
3	Ubuntu 10.04	Virtual Box 3.2.6	Fedora 13	35,976	12,889
4	Ubuntu 10.04	VMware Server 2.02	Fedora 13	32,236	10,135
5	Ubuntu 10.04	Xen Hypervisor 4.0	Fedora 13	32,121	9,997

O tempo de resposta da arquitetura medido para cada uma das diferentes tecnologias citadas no trabalho em relação ao Cenário I é apresentado na Tabela 14.

6.1.2. Cenário II

Neste cenário são utilizados dois nós roteadores na composição da VN e a aplicação do ataque em dois pontos distintos VNIC A e VNIC D. O ataque considerado é um ataque IDoS.

São mantidas as mesmas configurações para as tabelas de roteamento e endereçamento IP do InP e da VN do cenário anterior.

O SP ainda define qual política será utilizada pelo *mIdentifier* para a VN do Cenário II, e pode ser observada pela Figura 6.6. São definidos pelo SP um método de detecção diferente para cada interface monitorada, existentes na base de métodos, e a SLA existente na base de SLAs. Da mesma forma define em qual equipamento e qual interface deste equipamento será realizado o monitoramento. As políticas com $\mathrm{ID}=2$ e $\mathrm{ID}=3$ são utilizadas neste caso.

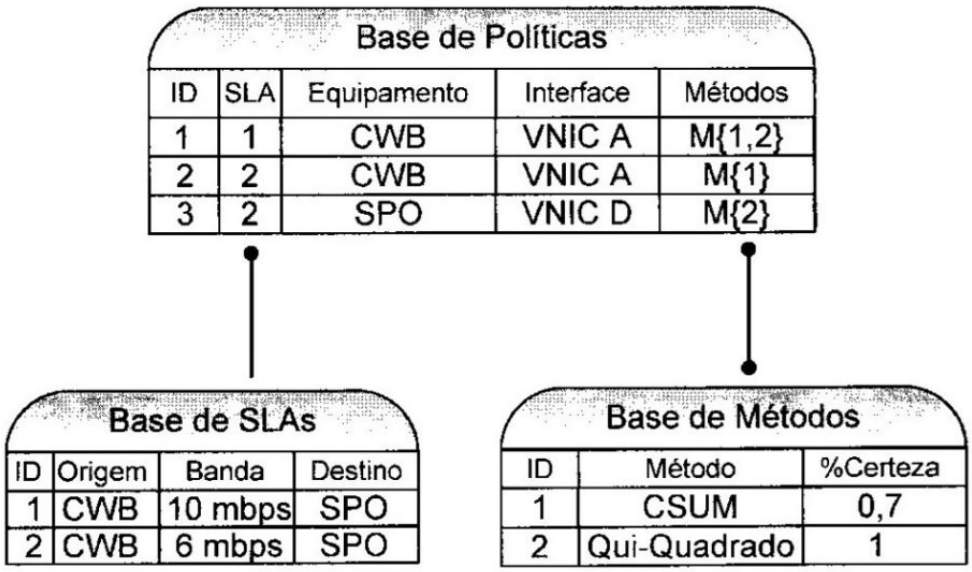


Figura 6.6. Base de políticas para o Cenário II

A máquina de agentes da ARVDoS é executada em momentos distintos no SP da VN. O agente *mIdentifier* passa a executar suas atividades, instanciando o método de detecção nos VNICs de acesso do roteador virtual CWB, assim com o tráfego da rede ativo, é disparado pela máquina DoS o ataque no VNIC A. A Figura 6.7 ilustra este cenário.

Com a detecção de IDoS o agente *mIdentifier* envia uma mensagem ao agente *mRemaker* com endereço IP de origem do ataque, IP destino e resposta do(s) método(s) instanciado(s), M_i. Esta mensagem é enviada de acordo com o retorno do(s) método(s) de detecção instanciado(s), este retorno pode ocorrer em tempos diferentes. A Tabela 15 apresenta os dados desta mensagem.

Tabela 15. Mensagem de configuração para do *mIndentifier* para *mRemaker* para o Cenário II

Mensagem	IP ORIGEM	IP DESTINO	M_i	
			Chi-quadrado	CSUM
1	192.168.13.10	192.19.9.10	$M_2 = 0$	$M_1 = 0{,}62$
2	192.19.9.10	192.198.13.10	$M_2 = 0{,}85$	$M_1 = 0$
3	192.168.13.10	192.19.9.10	-	$M_1 = 0$
4	192.19.9.10	192.168.13.10	$M_2 = 0$	-

Com estes dados o agente *mRemaker envia uma mensagem* de autorização *ao agente mInstaller* a fim de executar a instalação e a configuração do Roteador Virtualizado da Arquitetura (RVA) e sua tabela de roteamento assim como a instalação e configuração das interfaces de rede deste equipamento. As rotas reativas para este cenário têm as mesmas configurações observadas no Cenário I.

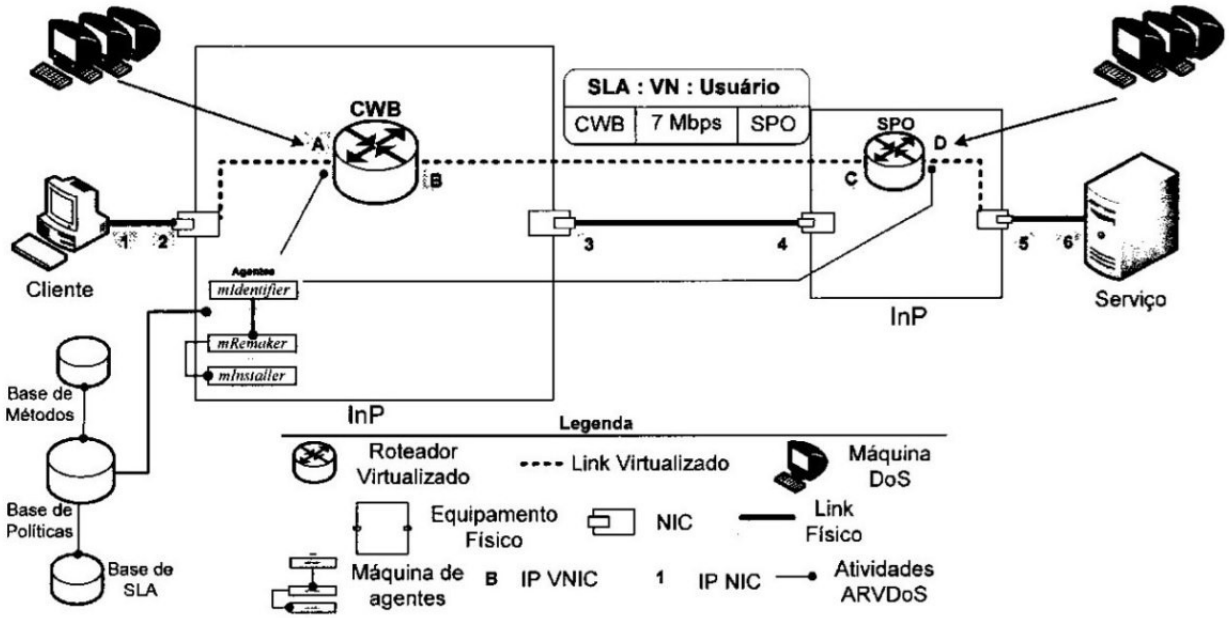


Figura 6.7. Cenário II de simulação InP e SP da rede virtual

Com o ataque DoS lançado no VNIC A e após o agente *mInstaller* realizar as configurações de endereçamento e repasse o agente *mRemaker* realiza o cálculo do PC de acordo com os valores de β. Com o resultado do PC, *mRemaker* calcula e aplica o resultado de LS ao enlace da SLA contratada e o resultado de LR aos enlaces entre CWB → RVA e entre RVA → SPO. O resultado obtido tanto para LR quanto para LS indicam o valor máximo que cada enlace pode assumir, $max(LR)$ e $max(LS)$. Esta redução da banda resulta em uma fila de descartes aos pacotes IDoS

no RV RVA que são enviados à taxa de ataque medida por um enlace com banda reduzida, desta forma reduzindo o impacto do ataque de negação de serviço de forma a conduzir o fluxo a NDoS.

A Tabela 16 apresenta os cálculos realizados pelo agente *mRemaker* para obtenção dos valores de reconfiguração do enlace entre CWB → RVA e a Tabela 17 para reconfiguração do enlace entre RVA → SPO.

Tabela 16. Valores de configuração de enlace entre CWB e RVA para o Cenário IIa

Mensagem	VNIC	PERCENTUAL DE CERTEZA		LS (Mbps)	LR (Mbps)
		β	PC		
1	E	CSUM 0,62	0,62	3,7	2,3

Tabela 17. Valores de configuração de enlace entre RVA e SPO para o Cenário IIb

Mensagem	VNIC	PERCENTUAL DE CERTEZA		LS (Mbps)	LR (Mbps)
		β	PC		
1	H	Qui-Quadrado 0,91	0,91	5,5	0,5

A Figura 6.8 apresenta a reconfiguração da topologia gerada pela ARVDoS quando IDoS é identificada pelo método instanciado. Esta referencia tem relação à mensagem um (1) apresentada na Tabela 16 e Tabela 17.

O comportamento do enlace no Cenário IIa para as atividades da arquitetura é representado pela Figura 6.9. É possível observar as variações do tráfego durante trezentos segundos (300 s) em relação à SLA de seis (6) Mbps.

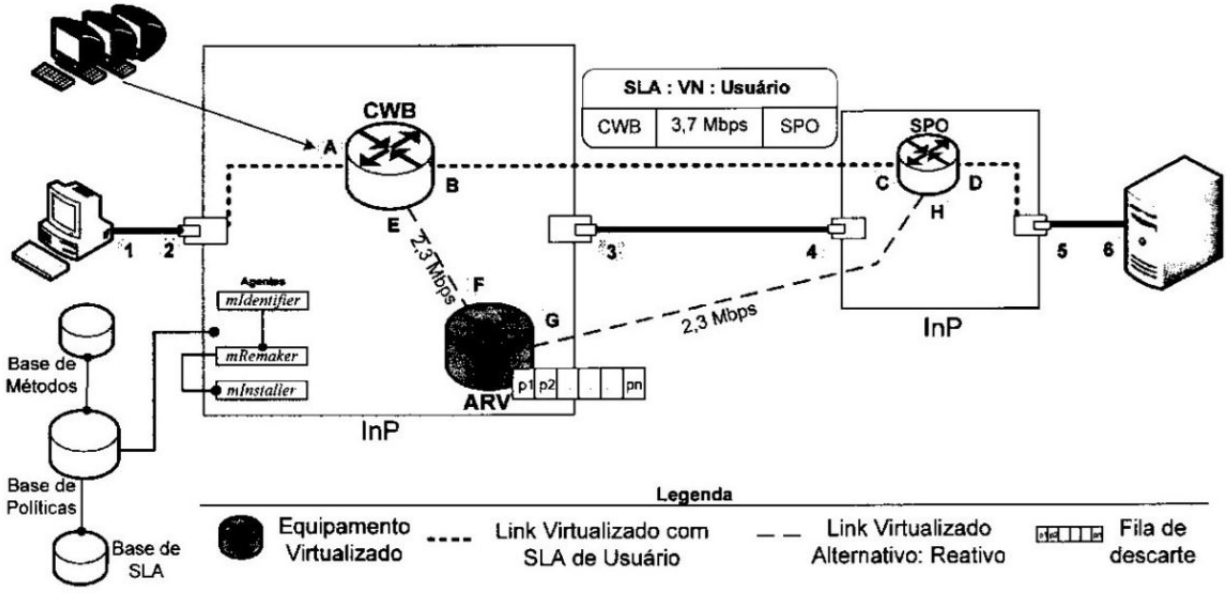


Figura 6.8. Topologia após a reação da ARVDoS para IDoS no Cenário IIa

Tais situações destacadas no gráfico e detalhadas na Tabela 18 ilustram o tráfego antes do ataque, o instante em que uma situação de DoS se configura, o tempo de atraso (delay) que o método de detecção leva para identificar o ataque, o tempo de atraso da ARVDoS para aplicar suas atividades, as alterações do enlace quando um novo valor de LR ou LS são calculados, assim como o encerramento da reação quando é verificado NDoS.

Tabela 18. Comportamento do enlace na topologia para o Cenário IIa

#	INTERVALO DE TEMPO (s)	ATIVIDADE ARVDoS	MÉDIA (Mbps)	max(LR) (Mbps)	max(LS) (Mbps)
1	1-60	Tráfego sem detecção de anomalia	3,02	-	6
2	61-83	Início do DoS e atraso (delay) do método CSUM em identificar a IDoS	5,05	-	6
3	84-94	Atraso (delay) da ARVDoS para realizar as atividades dos agentes *mIdentifier, mRemaker e mInstaller*	5,0	-	6
4	95-154	$PC = 0{,}62$ reação da ARVDoS com a criação do RVA e suas configurações e criação da fila de descarte com a aplicação do valor de LS nos VNICs A e B do RV CWB.	3,04	-	3,72
5	95-154	$PC = 0{,}62$ reação da ARVDoS com a criação entre CWB e RVA da fila de descarte com a aplicação do valor de LR nos VNICs E e G dos RVs CWB e RVA.	1,15	2,3	3,7
6	155-168	Atraso (delay) do método CSUM em identificar o NDoS, $PC = 0$.	3,06	2,3	3,7
7	169-172	Atraso (delay) da ARVDoS para reconfiguração da VN do valor de LR e LS quando NDoS é identificado pelo método instanciado. Os agentes *mIdentifier, mRemaker e mInstaller* desfazem as configurações reativas e retornam a topologia original.	3,11	2,3	3,7

8	173	$PC = 0$, NDoS identificado, o valor de LR é eliminado devido ao *mInstaller* ter desinstalado os VRs	-	-	6
9	174-300	$PC = 0$, o valor de LS é reconfigurado nos VNICs A e B do RV CWB para o valor da SLA. O tráfego volta a comportar-se sem anomalias.	3,65	-	6

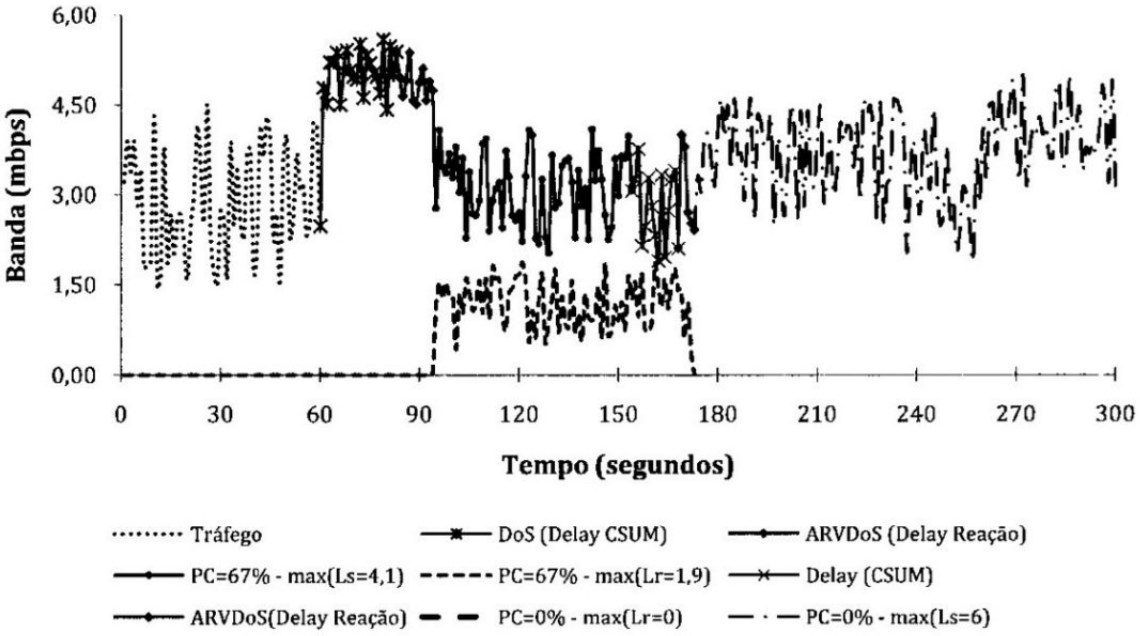


Figura 6.9. Comportamento do enlace na topologia para o Cenário IIa

O comportamento do enlace no Cenário IIb para as atividades da arquitetura é representado pela Figura 6.10. É possível observar as variações do tráfego em diferentes situações em uma linha de tempo de trezentos segundos (300 s) em relação à SLA que se mantêm seis (6) Mbps. Estas atividades ocorrem em paralelo no VNIC D e sua representação no gráfico são iniciadas aos duzentos e quarenta e cinco segundos (245 s) em paralelo ao monitoramento no VNIC A.

Tais situações destacadas no gráfico e detalhadas na Tabela 19 ilustram o tráfego antes do ataque, o instante em que uma situação de DoS se configura, o tempo de atraso (delay) que o método de detecção leva para identificar o ataque, o tempo de atraso da ARVDoS para aplicar suas atividades, as alterações do enlace quando um novo valor de LR ou LS são calculados, assim como o encerramento da reação quando é verificado NDoS.

Tabela 19. Comportamento do enlace na topologia para o Cenário IIb

#	INTERVALO DE TEMPO (s)	ATIVIDADE ARVDoS	MÉDIA (Mbps)	max(LR) (Mbps)	max(LS) (Mbps)
1	245-289	Tráfego sem detecção de anomalia	3,27	-	6
2	290-294	Início do DoS e atraso (delay) do método Chi-quadrdo em identificar a IDoS	5,26	-	6
3	295-308	Atraso (delay) da ARVDoS para realizar as atividades dos agentes *mIdentifier, mRemaker e mInstaller*	5,16	-	6
4	309-338	$PC = 0{,}89$ reação da ARVDoS com a criação do RVA e suas configurações e criação da fila de descarte com a aplicação do valor de LS nos VNICs D e VNIC C do RV SPO.	4,34	-	5,34
5	309-338	$PC = 0{,}89$ reação da ARVDoS com a criação entre SPO e RVA da fila de descarte com a aplicação do valor de LR nos VNICs H e F dos RVs SPO e RVA.	0,42	0,66	5,34
6	339-343	Atraso (delay) do método Chi-quadrado em identificar o NDoS, $PC = 0$.	4,66	0,66	5,34
7	344-347	Atraso (delay) da ARVDoS para reconfiguração da VN do valor de LR *e* LS quando NDoS é identificado pelo método instanciado. Os agentes *mIdentifier, mRemaker e mInstaller* desfazem as configurações reativas e retornam a topologia original.	4,33	0,66	5,34
8	348	$PC = 0$, NDoS identificado, o valor de LR é eliminado devido ao *mInstaller* ter desinstalado os VRs	4,33	-	6
9	349-399	$PC = 0$, o valor de LS é reconfigurado nos VNICs C e D do RV	3,28	-	6

		SPO para o valor da SLA. O tráfego volta a comportar-se sem anomalias.			
10	400-403	Início do DoS e atraso (delay) do método Chi-quadrdo em identificar a IDoS.	5,48	-	6
11	404-412	Atraso (delay) da ARVDoS para realizar as atividades dos agentes *mIdentifier, mRemaker e mInstaller*	5,43	-	6
12	413-457	$PC = 0{,}91$ reação da ARVDoS com a criação do RVA e suas configurações e criação da fila de descarte com a aplicação do valor de LS nos VNICs D e VNIC C do RV SPO.	4,42	-	5,5
13	413-457	$PC = 0{,}91$ reação da ARVDoS com a criação entre SPO e RVA da fila de descarte com a aplicação do valor de LR nos VNICs H e F dos RVs SPO e RVA.	0,3	0,5	5,5
14	458-462	Atraso (delay) do método Chi-quadrado em identificar o NDoS, $PC = 0$	4,5	0,5	5,5
15	463-465	Atraso (delay) da ARVDoS para reconfiguração da VN do valor de $LR\ e\ LS$ quando NDoS é identificado pelo método instanciado. Os agentes *mIdentifier, mRemaker e mInstaller* desfazem as configurações reativas e retornam a topologia original.	4,78	0,5	5,5
16	466	$PC = 0$, NDoS identificado, o valor de LR é eliminado devido ao *mInstaller* ter desinstalado os VRs	4,78	-	6
17	467-544	$PC = 0$, o valor de LS é reconfigurado nos VNICs C e D do RV SPO para o valor da SLA. O tráfego volta a comportar-se sem anomalias.	4,06	-	6

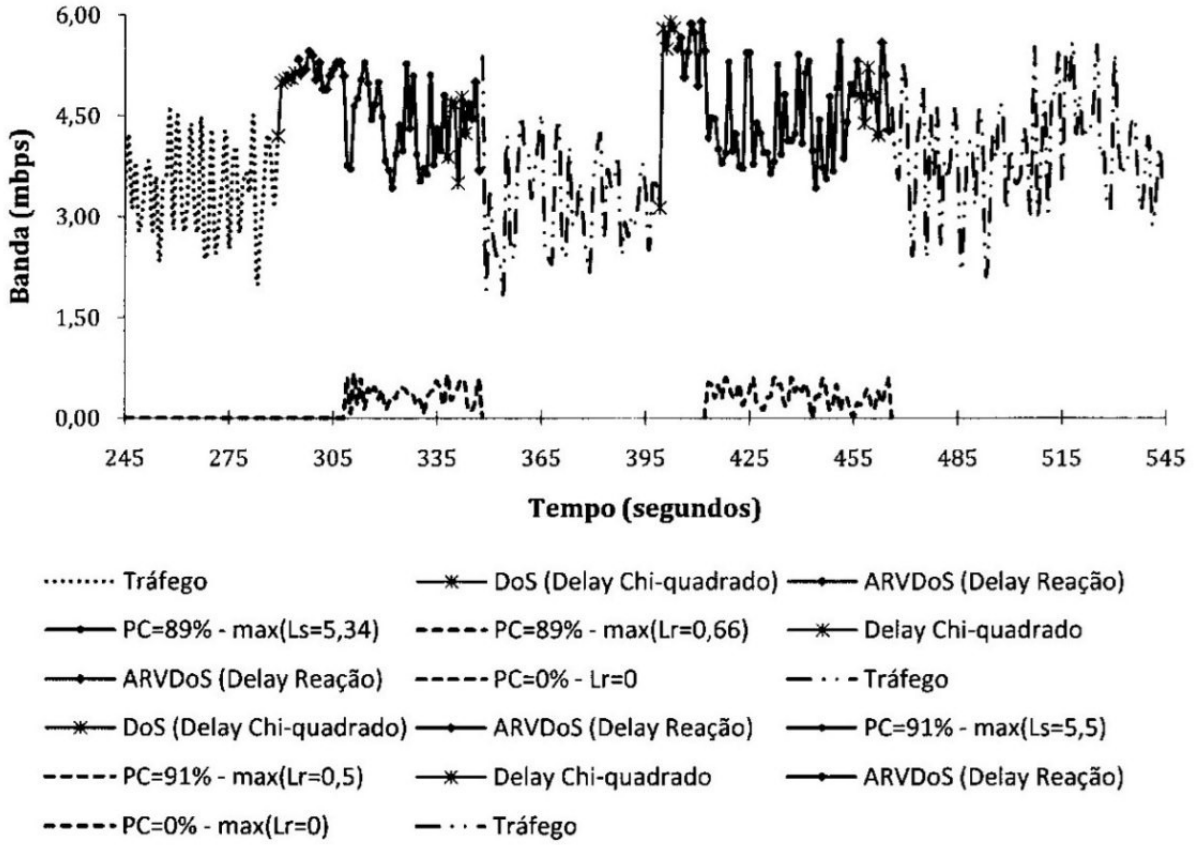


Figura 6.10. Comportamento do enlace na topologia para o Cenário IIb

Para realizar uma observação do desempenho da ARVDoS é interessante observar o intervalo de confiança de forma a comparar os tempos de identificação da anomalia pelos métodos de detecção utilizados, assim como os tempos de reação da arquitetura.

O nível de confiança desejado é 95% (noventa e cinco por cento) ou 0,05. A amostra utilizada contém dez (10) avaliações do Cenário IIa e vinte (20) avaliações para o Cenário IIb, com a qual é obtida a média e o desvio padrão em segundos. O intervalo de confiança é calculado para os métodos de detecção utilizados no cenário e para os tempos gerados pelas reações da arquitetura aos percentuais de certeza obtidos.

Para o cenário IIa são observadas as reações ARVDoS 1 e ARVDoS 2 e para o Cenário IIb as três reações. A Tabela 20 e a Figura 6.11 apresentam estes resultados para o Cenário IIa e a Tabela 21 e Figura 6.12 para o Cenário IIb.

Tabela 20. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário IIa

Nível de Confiança = 95% (0,05)					
Atividade	**+**	**Média**	**-**	**Intervalo de Confiança**	**Amostras**
Método 1	23,13	21,7	20,27	1,43	10
ARVDoS 1	10,84	10,1	9,36	0,74	10
ARVDoS 3	4,56	4,1	3,64	0,46	10

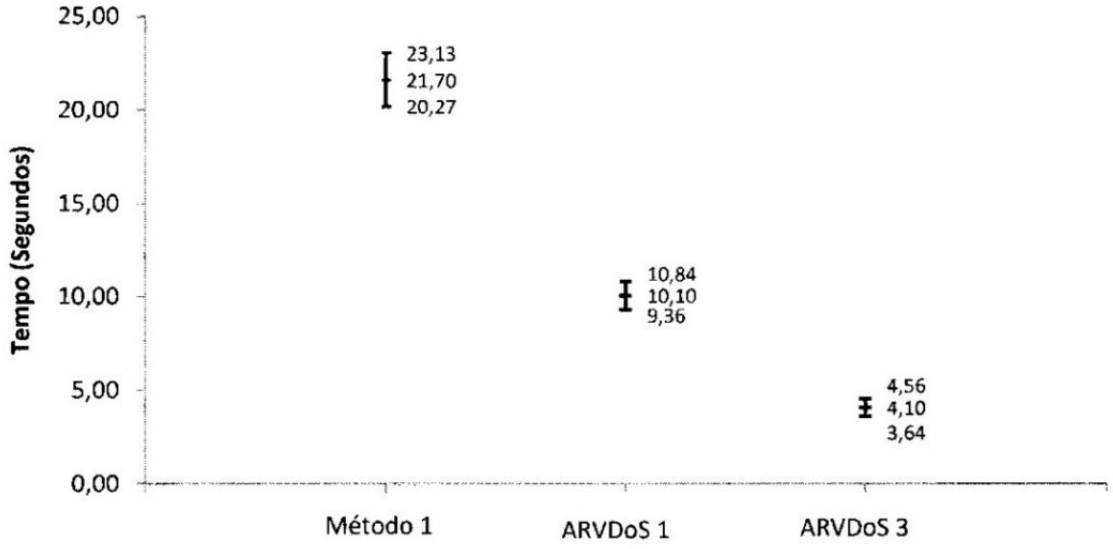


Figura 6.11. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário IIa

Tabela 21. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário IIb

Nível de Confiança = 95% (0,05)					
Atividade	**+**	**Média**	**-**	**Intervalo de Confiança**	**Amostras**
Método 2	4,55	4,2	3,85	0,35	20
ARVDoS 1	11,97	11,5	11,03	0,47	20
ARVDoS 3	4,6	4,3	4,0	0,30	20

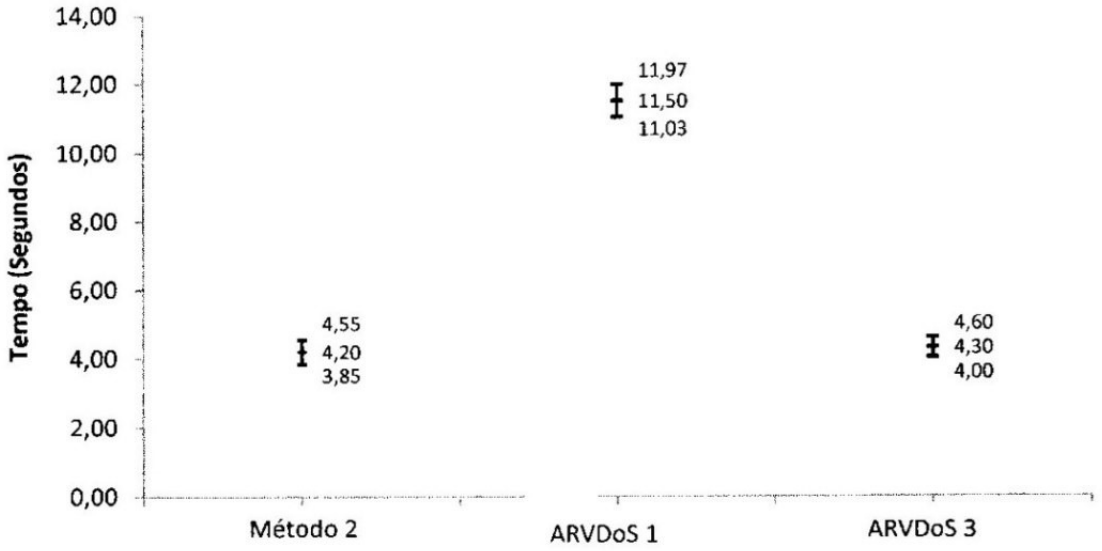


Figura 6.12. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário IIb

O tempo de resposta da arquitetura medido para o Cenário IIa e Cenário IIb com IDoS é apresentado na Tabela 22.

Tabela 22. Tempo de reação da arquitetura para IDoS para o Cenário IIa e Cenário IIb

#	Plataforma	Virtualização	SO Host	Tempo de Reação (s)			
				1º Ensaio VNIC A	2º Ensaio VNIC A	1º Ensaio VNIC D	2º Ensaio VNIC D
1	Windows XP	Virtual Box 3.2.6	Fedora 12	35,674	14,241	34,819	12,611
2	Windows XP	VMware Server 2.02	Fedora 12	33,964	10,981	34,698	12,057
3	Ubuntu 10.04	Virtual Box 3.2.6	Fedora 12	35,432	13,132	34,126	11,881
4	Ubuntu 10.04	VMware Server 2.02	Fedora 12	32,819	10,629	32,075	10,644
5	Ubuntu 10.04	Xen Hypervisor 4.0	Fedora 12	31,331	9,798	32,648	10,213

6.1.3. Cenário III

Neste cenário são utilizados três nós roteadores na composição da VN e a aplicação do ataque em um ponto VNIC A. O ataque considerado é um ataque do tipo IDoS. Posteriormente, após a identificação e tratamento do ataque é possível observar a aplicação do NDoS quando os métodos de detecção identificam que o percentual de certeza passa a ser zero ($PC = 0$). A Figura 6.13 ilustra este cenário.

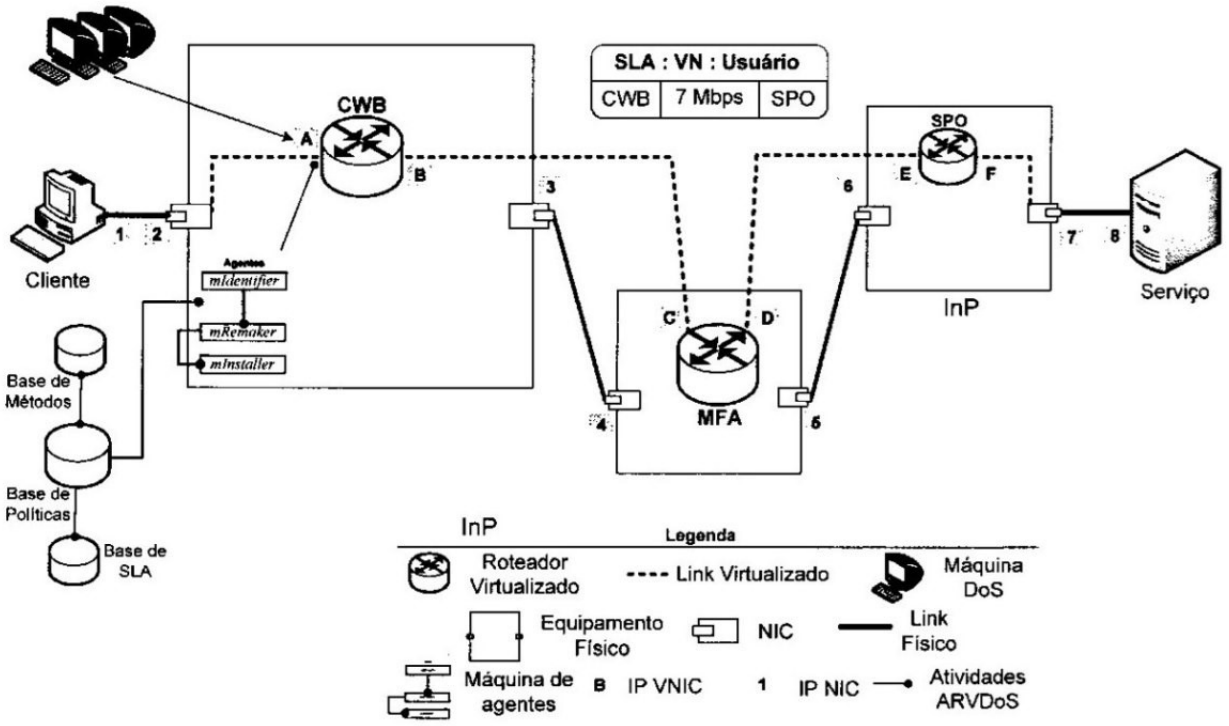


Figura 6.13. Cenário III de simulação InP e SP da rede virtual

A configuração de rede nas topologias do InP e da VN são apresentadas na Tabela 23.

Tabela 23. Configurações de endereçamento de rede do InP e da VN

InP		*VN*	
NIC	**IP/MÁSCARA**	**VNIC**	**IP/MÁSCARA**
1	192.168.13.10/24	1	192.168.13.10/24
2	192.168.13.1/24	A	192.168.13.2/24
3	200.27.16.13/30	B	201.10.15.1/30
4	200.27.16.14/30	C	201.10.15.2/30
5	200.27.16.17/30	D	201.10.15.5/30
6	200.27.16.18/30	E	201.10.15.6/30
7	192.19.9.1/24	F	192.19.9.2/24
8	192.19.9.10/24	8	192.19.9.10/24

As tabelas de roteamento para os roteadores virtuais CWB, SPO e MFA são apresentadas na Tabela 24, Tabela 25 e Tabela 26.

Tabela 24. Tabela de roteamento do roteador virtual CWB

TABELA DE ROTEAMENTO CWB (Virtualizado)			
ID REDE DESTINO	MÁSCARA DESTINO	PRÓXIMO SALTO	INTERFACE
192.168.13.0	255.255.255.0	127.0.0.1	A

192.19.9.0	255.255.255.0	201.10.15.2	B
201.10.15.0	255.255.255.252	201.10.15.2	B
201.10.15.4	255.255.255.252	201.10.15.2	B

Tabela 25. Tabela de roteamento do roteador virtual MFA

TABELA DE ROTEAMENTO MFA (Virtualizado)			
ID REDE DESTINO	MÁSCARA DESTINO	PRÓXIMO SALTO	INTERFACE
192.168.13.0	255.255.255.0	201.10.15.1	C
201.10.15.0	255.255.255.252	201.10.15.1	C
201.10.15.4	255.255.255.252	201.10.15.6	D
192.19.9.0	255.255.255.0	201.10.15.6	D

Tabela 26. Tabela de roteamento do roteador virtual SPO

TABELA DE ROTEAMENTO SPO (Virtualizado)			
ID REDE DESTINO	MÁSCARA DESTINO	PRÓXIMO SALTO	INTERFACE
192.168.13.0	255.255.255.0	201.10.15.5	E
201.10.15.0	255.255.255.252	201.10.15.5	E
201.10.15.4	255.255.255.252	201.10.15.5	E
192.19.9.0	255.255.255.0	127.0.0.1	D

O SP ainda define qual política será utilizada pelo *mIdentifier* para a VN do Cenário III, e pode ser observada pela Figura 6.14. São definidos pelo SP três métodos de detecção existentes na base de métodos e a duas SLAs existentes na base de SLAs.

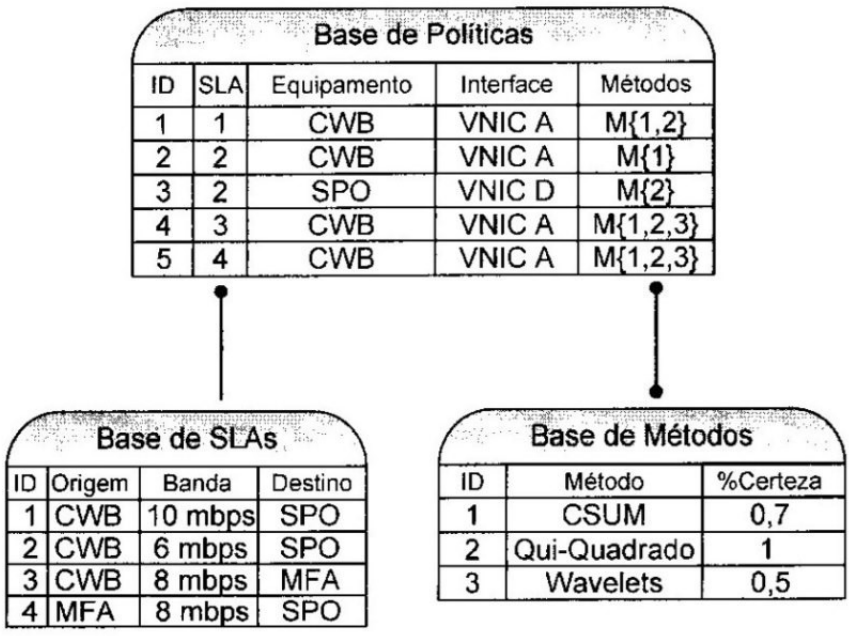


Figura 6.14. Base de políticas para o Cenário III

Da mesma forma define em qual equipamento e qual interface deste equipamento será realizada o monitoramento. As políticas com $ID=4$ e $ID=5$ são utilizadas neste caso.

A máquina de agentes da ARVDoS é executada em paralelo no SP da VN. O agente *mIdentifier* passa a executar suas atividades, instanciando os métodos de detecção no VNIC de acesso do roteador virtual CWB, assim com o tráfego da rede ativo, é disparado um ataque pela máquina DoS no VNIC A.

Com a detecção de IDoS o agente *mIdentifier* envia uma mensagem ao agente *mRemaker* com endereço IP de origem do ataque, IP destino e resposta do(s) método(s) instanciado(s), M_i. Esta mensagem é enviada de acordo com o retorno do(s) método(s) de detecção instanciado(s), este retorno pode ocorrer em tempos diferentes. A Tabela 27 apresenta os dados desta mensagem.

Tabela 27. Mensagem de configuração para do *mIndentifier* para *mRemaker* para o Cenário III

Mensagem	IP ORIGEM	IP DESTINO	M_i		
			Chi-quadrado	CSUM	Wavelets
1	192.168.13.10	192.19.9.10	$M_2 = 0{,}81$	$M_1 = 0$	$M_3 = 0$
2	192.168.13.10	192.19.9.10	$M_2 = 0{,}81$	$M_1 = 0{,}57$	$M_3 = 0$
3	192.168.13.10	192.19.9.10	$M_2 = 0{,}81$	$M_1 = 0{,}57$	$M_3 = 0{,}50$
4	192.168.13.10	192.19.9.10	$M_2 = 0$	$M_1 = 0$	$M_3 = 0$

Com estes dados o agente *mRemaker envia uma mensagem* de autorização *ao agente mInstaller* a fim de executar a instalação e a configuração do Roteador Virtualizado da Arquitetura (RVA) e sua tabela de roteamento assim como a instalação e configuração das interfaces de rede deste equipamento. A Tabela 28 apresenta a configuração dos VNICs instalados e a Tabela 29 apresenta a tabela de roteamento para RVA.

Tabela 28. Configuração dos VNICs da VN reativa

VN REATIVA	
VNIC	**IP/MÁSCARA**
G	201.10.15.9/30
H	201.10.15.10/30
I	201.10.15.13/30
J	201.10.15.14/30

Tabela 29. Tabela de roteamento do RVA

TABELA DE ROTEAMENTO RVA			
ID REDE DESTINO	MÁSCARA DESTINO	PRÓXIMO SALTO	INTERFACE
192.168.13.0	255.255.255.0	201.10.15.9	H
201.10.15.8	255.255.255.252	201.10.15.9	H
201.10.15.12	255.255.255.252	201.10.15.14	I
192.19.9.0	255.255.255.0	201.10.15.14	I

O agente *mRemaker* estabelece uma sessão remota com o RV para alteração e configuração dos VNICs e de suas tabelas de roteamento. A Tabela 30 e Tabela 31 mostram as rotas acrescentadas em RV CWB e RV MFA.

Tabela 30. Rotas adicionadas ao RV CWB após reação

TABELA DE ROTEAMENTO CWB			
ID REDE DESTINO	MÁSCARA DESTINO	PRÓXIMO SALTO	INTERFACE
201.10.15.8	255.255.255.252	201.10.15.10	G
201.10.15.12	255.255.255.252	201.10.15.10	G

Tabela 31. Rotas adicionadas ao RV MFA após reação

TABELA DE ROTEAMENTO MFA			
ID REDE DESTINO	MÁSCARA DESTINO	PRÓXIMO SALTO	INTERFACE
201.10.15.8	255.255.255.252	201.10.15.13	J
201.10.15.12	255.255.255.252	201.10.15.13	J

Com as configurações de endereçamento e repasse finalizadas o agente *mRemaker* realiza o cálculo do PC de acordo com os valores de β. Com o resultado do PC, *mRemaker* calcula e aplica o resultado de LS ao enlace da SLA contratada e o resultado de LR aos enlaces entre CWB → RVA e entre RVA → MFA. O resultado obtido tanto para LR quanto para LS indicam o valor máximo que cada enlace pode assumir, $max(LR)$ e $max(LS)$. Esta redução da banda resulta em uma fila de descartes aos pacotes IDoS no RV RVA que são enviados à taxa de ataque medida por um enlace com banda reduzida, desta forma reduzindo o impacto do ataque de negação de serviço de forma a conduzir o fluxo a NDoS.

A Tabela 32 apresenta os cálculos realizados pelo agente *mRemaker* para obtenção dos valores de reconfiguração do enlace entre CWB → RVA e a Tabela 33 para reconfiguração do enlace entre RVA → MFA.

Tabela 32. Valores de configuração de enlace entre CWB e RVA

Mensagem	VNIC	PERCENTUAL DE CERTEZA				LS (Mbps)	LR (Mbps)
		β			PC		
		Chi-quadrado	CSUM	Wavelets			
1	G	0,81	0	0	0,37	3,95	5,05
2	G	0,81	0,57	0	0,55	4,4	3,6
3	G	0,81	0,57	0,5	0,66	5,3	2,7

Tabela 33. Valores de configuração de enlace entre RVA e MFA

Mensagem	VNIC	PERCENTUAL DE CERTEZA				LS (Mbps)	LR (Mbps)
		β			PC		
		Chi-quadrado	CSUM	Wavelets			
1	I	0,81	0	0	0,37	3,95	5,05
2	I	0,81	0,57	0	0,55	4,4	3,6
3	I	0,81	0,57	0,5	0,66	5,3	2,7

A Figura 6.15 apresenta a reconfiguração da topologia gerada pela ARVDoS quando IDoS é identificada por dois métodos instanciados. Esta referencia tem relação à mensagem dois (2) apresentada na Tabela 10 e Tabela 11.

A Figura 6.16 apresenta o gráfico de comportamento da VN no Cenário I para as atividades da ARVDoS.

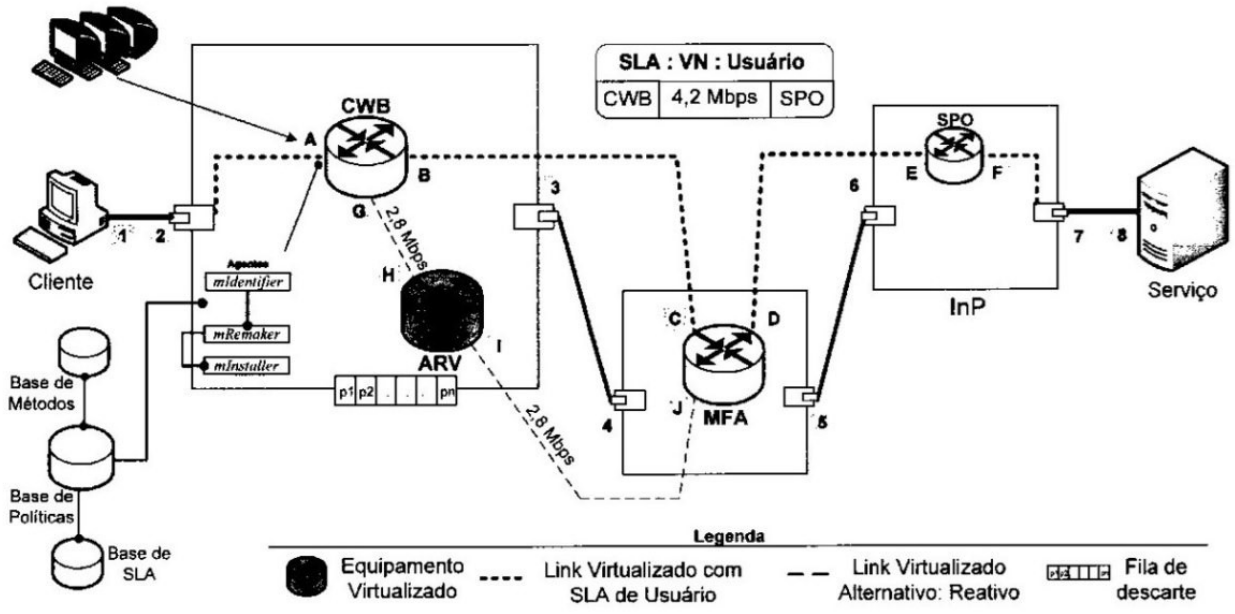


Figura 6.15. Topologia após a reação da ARVDoS para IDoS para Cenário III

O comportamento do enlace no Cenário III para as atividades da arquitetura é representado pela Figura 6.16. É possível observar as variações do tráfego em diferentes situações em uma linha de tempo de trezentos segundos (300 s) em relação à SLA contratado pelo usuário oito (8) Mbps.

Tais situações destacadas no gráfico e detalhadas na Tabela 34 ilustram o tráfego antes do ataque, o instante em que uma situação de DoS se configura, o tempo de atraso (delay) que o método de detecção leva para identificar o ataque, o tempo de atraso da ARVDoS para aplicar suas atividades, as alterações do enlace quando um novo valor de LR ou LS são calculados, assim como o encerramento da reação quando é verificado NDoS.

Tabela 34. Comportamento do enlace na topologia para o Cenário III

#	INTERVALO DE TEMPO (s)	ATIVIDADE ARVDoS	MÉDIA (Mbps)	max(LR) (Mbps)	max(LS) (Mbps)
1	1-60	Tráfego sem detecção de anomalia	3,77	-	8
2	61-66	Início do DoS e atraso (delay) do método Chi-quadro em identificar a IDoS	6,58	-	8
3	67-78	Atraso (delay) da ARVDoS para realizar as atividades dos agentes *mIdentifier, mRemaker e mInstaller*	6,56	-	8
4	79-86	$PC = 0{,}37$ reação da ARVDoS com a criação do RVA e suas configurações e criação da fila de descarte com a aplicação do valor de LS nos VNICs A e B do RV CWB.	2,76	-	3,0
5	79-86	$PC = 0{,}37$ reação da ARVDoS com a criação entre CWB e RVA da fila de descarte com a aplicação do valor de LR nos VNICs E e G dos RVs CWB e RVA.	4,36	5,0	3,0
6	87-92	Atraso (delay) da ARVDoS para reconfiguração da fila de descarte e reconfigurar os valores de LR e LS para realizar as atividades dos agentes *mIdentifier, mRemaker* devido ao tempo de detecção do método	2,62	5,0	3,0

		CSUM.			
7	93-99	$PC = 0{,}55$, reação da ARVDoS com reconfiguração dos valores de LS.	4,9	-	4,4
8	93-99	$PC = 0{,}55$, reação da ARVDoS com reconfiguração dos valores de LR.	4,9	3,6	4,4
9	100-105	Atraso (delay) da ARVDoS para reconfiguração da fila de descarte e reconfigurar os valores de $LR\ e\ LS$ para realizar as atividades dos agentes *mIdentifier, mRemaker* devido ao tempo de detecção do método Wavelets.	4,06	3,6	4,4
10	106-120	$PC = 0{,}66$, reação da ARVDoS com reconfiguração dos valores de LS.	4,63	-	5,3
11	106-120	$PC = 0{,}66$, reação da ARVDoS com reconfiguração dos valores de LR.	1,8	2,7	5,3
12	121-125	Atraso (delay) da ARVDoS para reconfiguração da VN do valor de $LR\ e\ LS$ quando NDoS é identificado pelos métodos instanciados. Os agentes *mIdentifier, mRemaker e mInstaller* desfazem as configurações reativas e retornam a topologia original.	4,82	2,7	5,3
11	126	$PC = 0$, NDoS identificado, o valor de LR é eliminado devido ao *mInstaller* ter desinstalado os VRs	-	-	8
12	127-300	$PC = 0$, o valor de LS é reconfigurado nos VNICs A e B do RV CWB para o valor da SLA. O tráfego volta a comportar-se sem anomalias.	4,12	-	8

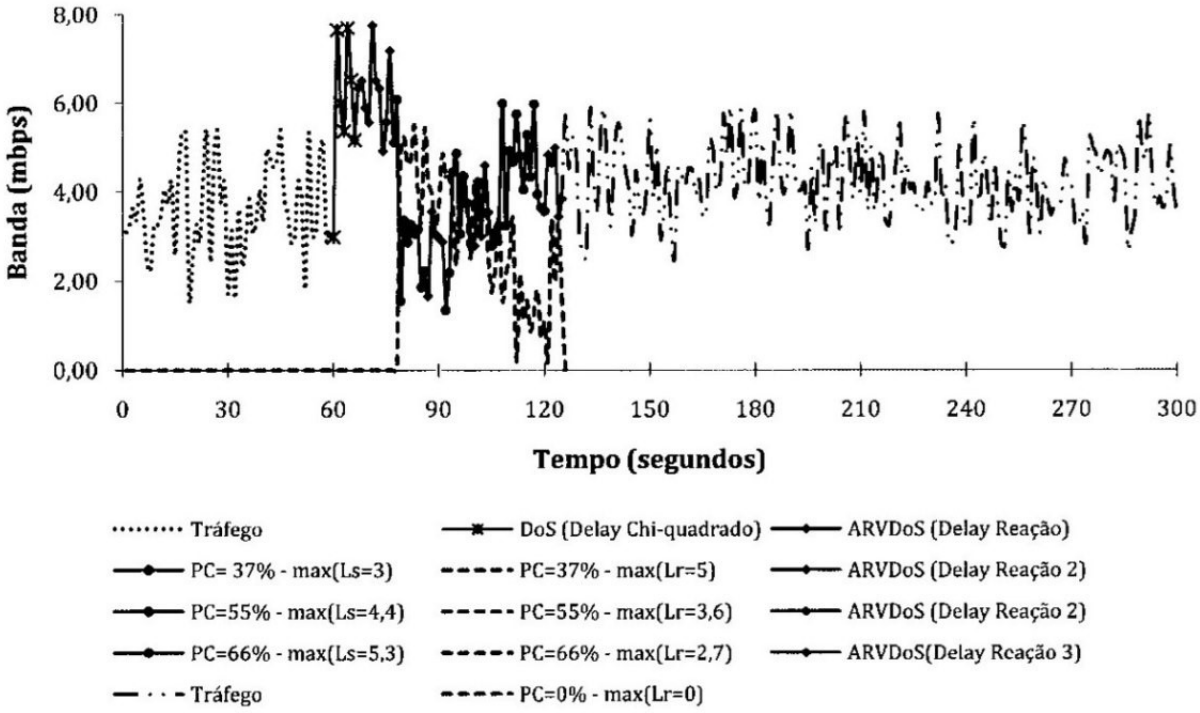


Figura 6.16. Comportamento do enlace na topologia para o Cenário III

Para realizar uma observação do desempenho da ARVDoS é interessante observar o intervalo de confiança de forma a comparar os tempos de identificação da anomalia pelos métodos de detecção utilizados, assim como os tempos de reação da arquitetura.

O nível de confiança desejado é 95% (noventa e cinco por cento) ou 0,05. A amostra utilizada contém dez (10) avaliações do Cenário III, exceto para a ARVDoS 2 que contém vinte (20) amostras, com as quais são obtidas a média e o desvio padrão em segundos. O intervalo de confiança é calculado para os métodos de detecção utilizados no cenário e para os tempos gerados pelas reações da arquitetura aos percentuais de certeza obtidos.

Neste cenário são observadas as reações ARVDoS 1 e ARVDoS 2 e ARVDoS 3. A Tabela 35 e a Figura 6.17 apresentam estes resultados.

Tabela 35. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário III

Nível de Confiança = 95% (0,05)					
Atividade	**+**	**Média**	**-**	**Intervalo de Confiança**	**Amostras**
Método 2	5,36	4,9	4,44	0,46	10
Método 1	25,56	24,8	24,04	0,76	10
Método 3	31,75	30,9	30,05	0,85	10

ARVDoS 1	11,64	11,1	10,56	0,54	10
ARVDoS 2	5,04	4,6	4,16	0,44	20
ARVDoS 3	5,29	4,8	4,31	0,49	10

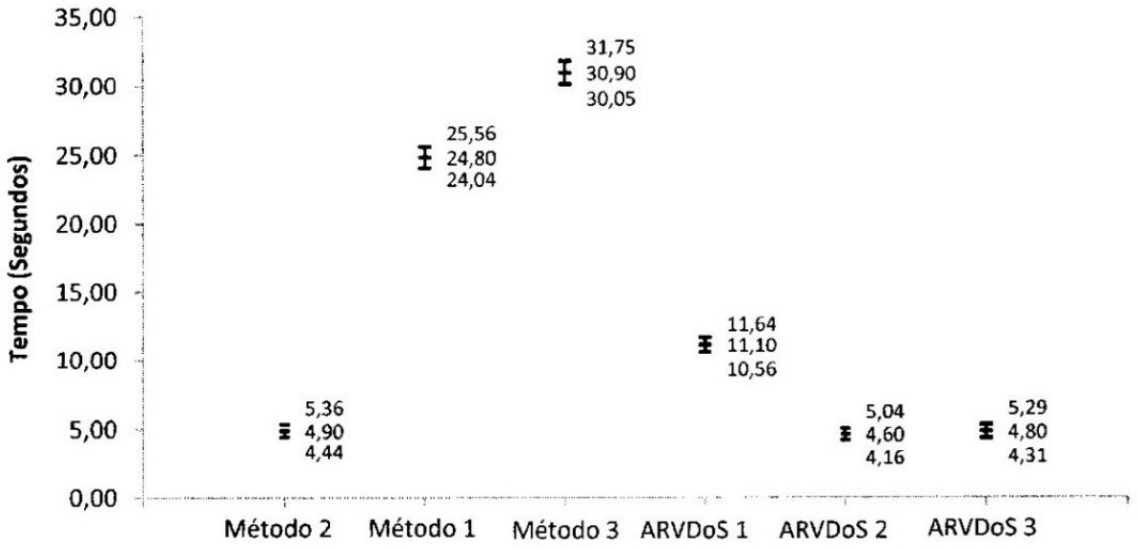


Figura 6.17. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário III

Para as avaliações apresentadas anteriormente foi utilizado como referencia para implementação na ARVDoS a tecnologia de Virtualização Xen Hypervisor 4.0 por apresentar um tempo aproximado de 40% menor para inicialização das VMs.

O tempo de resposta da arquitetura medido para cada uma das diferentes tecnologias citadas no trabalho em relação ao Cenário III é apresentado na Tabela 36.

Tabela 36. Tempo de reação da arquitetura para IDoS para o Cenário III

#	Plataforma	Virtualização	SO Host	Tempo de Reação (s)	
				1° Ensaio	2ª Ensaio
1	Windows XP	Virtual Box 3.2.6	Fedora 13	52,653	24,303
2	Windows XP	VMware Server 2.02	Fedora 13	50,342	21,482
3	Ubuntu 10.04	Virtual Box 3.2.6	Fedora 13	49,087	22,798
4	Ubuntu 10.04	VMware Server 2.02	Fedora 13	42,347	20,024
5	Ubuntu 10.04	Xen Hypervisor 4.0	Fedora 13	40,232	19,886

6.1.4. Cenário IV

Neste cenário são mantidas as mesmas configurações para as tabelas de roteamento e endereçamento IP do InP e da VN do Cenário I para observar um evento em que CDoS fosse identificado. São utilizados dois nós roteadores na composição da VN e a aplicação do ataque em um ponto distinto VNIC.

Posteriormente, após a identificação e tratamento do ataque é possível observar a aplicação do NDoS quando os métodos de detecção identificam que o percentual de certeza passa a ser zero ($PC = 0$).

Base de Políticas

ID	SLA	Equipamento	Interface	Métodos
1	1	CWB	VNIC A	M{1,2}
2	2	CWB	VNIC A	M{2}
3	2	SPO	VNIC D	M{2}

Base de SLAs

ID	Origem	Banda	Destino
1	CWB	10 mbps	SPO
2	CWB	6 mbps	SPO

Base de Métodos

ID	Método	%Certeza
1	CSUM	0,7
2	Qui-Quadrado	1

Figura 6.18. Base de políticas para o Cenário IV

O SP define qual política será utilizada pelo *mIdentifier* para a VN do Cenário IV, a qual pode ser observada na Figura 6.18. É definido pelo SP um método de detecção para a interface monitorada, existente na base de métodos, e a SLA existente na base de SLAs. Da mesma forma define em qual equipamento e qual interface deste equipamento será realizado o monitoramento. As políticas com $ID=2$ são utilizadas neste caso.

A máquina de agentes da ARVDoS é executada em paralelo no SP da VN. O agente *mIdentifier* passa a executar suas atividades, instanciando o método de detecção no VNIC de acesso do roteador virtual CWB, assim com o tráfego da rede ativo, é disparado um ataque pela máquina DoS no VNIC A.

Com a detecção de CDoS o agente *mIdentifier* envia uma mensagem ao agente *mRemaker* com endereço IP de origem do ataque, IP destino e resposta do método instanciado, M_i. A Tabela 37 apresenta os dados desta mensagem.

Tabela 37. Mensagem de configuração para do mIndentifier para mRemaker para o Cenário IV

Mensagem	IP ORIGEM	IP DESTINO	M_i
			Chi-quadrado
1	192.168.13.10	192.19.9.10	$M_2 = 1{,}0$

O agente *mRemaker* realiza o cálculo do PC de acordo com os valores de β. Com estes dados o agente *mRemaker envia uma mensagem ao agente mInstaller* a fim de direcionar o tráfego CDoS para o buraco negro (/null).

A Tabela 38 apresenta o cálculo realizados pelo agente *mRemaker* para obtenção do valor de PC e direcionamento para o buraco negro no RV CWB.

Tabela 38. Valores de configuração de enlace entre CWB e RVA

Mensagem	VNIC	PERCENTUAL DE CERTEZA		LS (Mbps)	LR (Mbps)
		β	PC		
		Chi-quadrado			
1	E	1,0	1,0	10	-

A Figura 6.19 apresenta a reconfiguração da topologia gerada pela ARVDoS quando CDoS é identificado. Esta referencia tem relação à mensagem um (1) apresentada na Tabela 38.

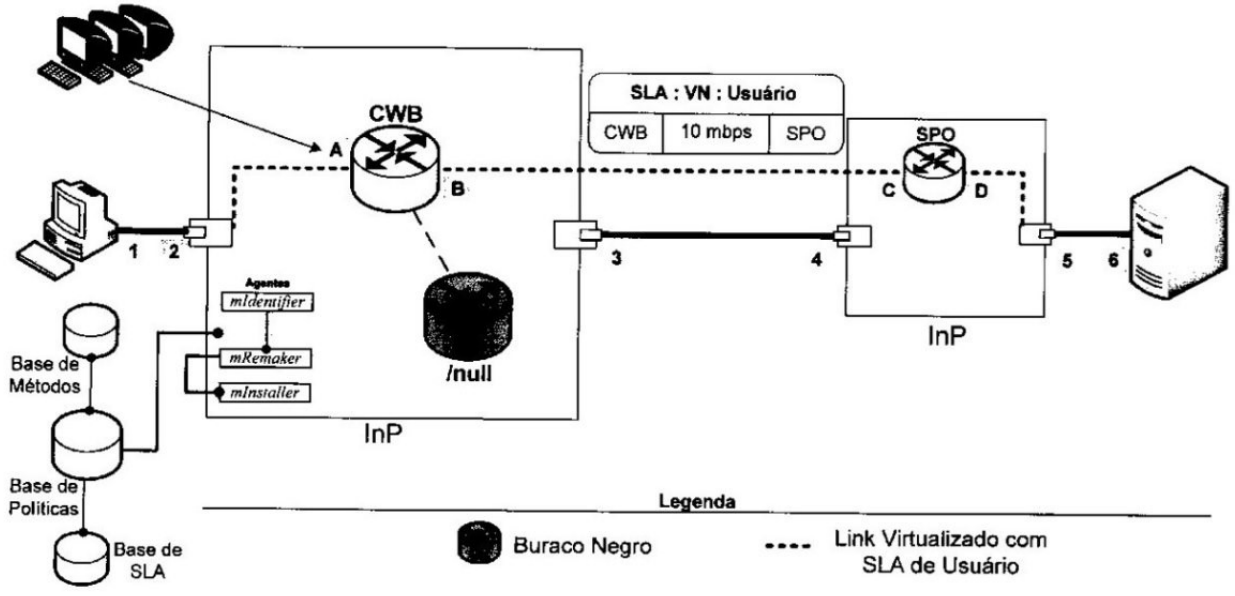


Figura 6.19. Topologia após a reação da ARVDoS para CDoS para o Cenário IV

O comportamento do enlace no Cenário IV para as atividades da arquitetura é representado pela Figura 6.20. É possível observar as variações do tráfego em diferentes situações em uma linha de tempo de trezentos segundos (300 s) em relação à SLA contratado pelo usuário dez (10) Mbps.

Tais situações destacadas no gráfico e detalhadas na Tabela 39 ilustram o tráfego antes do ataque, o instante em que uma situação de CDoS se configura, o tempo de atraso (delay) que o método de detecção leva para identificar o ataque, o tempo de atraso da ARVDoS para aplicar suas atividades, assim como o encerramento da reação quando é verificado NDoS.

Tabela 39. Comportamento do enlace na topologia para o Cenário IV

#	INTERVALO DE TEMPO (s)	ATIVIDADE ARVDoS	MÉDIA (Mbps)	max(LR) (Mbps)	max(LS) (Mbps)
1	1-30	Tráfego sem detecção de anomalia	5,43	-	-
2	31-37	Início do DoS e atraso (delay) do método Chi-quadro em identificar a IDoS	8,31	-	-
3	38-41	Atraso (delay) da ARVDoS para realizar as atividades dos agentes *mIdentifier, mRemaker e mInstaller*	8,36	-	-
4	41	$PC = 1$ reação da ARVDoS com o direcionamento do tráfego para o buraco negro.	7,14	-	-
5	42-300	$PC = 0$ reação da ARVDoS com a manutenção do tráfego de acordo com a SLA de usuário.	5,12	-	-

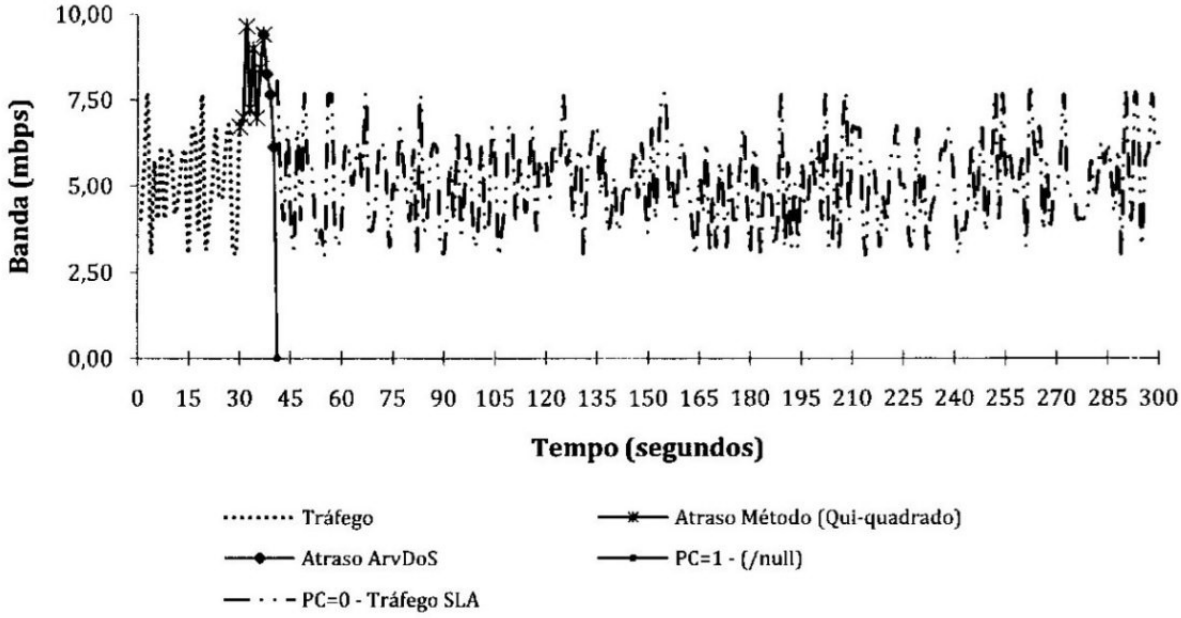


Figura 6.20. Comportamento do enlace na topologia para o Cenário IV

Para realizar uma observação do desempenho da ARVDoS é interessante observar o intervalo de confiança de forma a comparar os tempos de identificação da anomalia pelos métodos de detecção utilizados, assim como os tempos de reação da arquitetura.

O nível de confiança desejado é 95% (noventa e cinco por cento) ou 0,05. A amostra utilizada contém dez (10) avaliações do Cenário IV, com a qual é obtida a média e o desvio padrão em segundos. O intervalo de confiança é calculado para os métodos de detecção utilizados no cenário e para os tempos gerados pelas reações da arquitetura aos percentuais de certeza obtidos. Neste cenário é observada a reação ARVDoS 4.

A Tabela 40 e a Figura 6.21 apresentam estes resultados.

Tabela 40. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário IV

Nível de Confiança = 95% (0,05)					
Atividade	**+**	**Média**	**-**	**Intervalo de Confiança**	**Amostras**
Método 2	6,03	5,50	4,97	0, 53	10
ARVDoS 1	3,83	3,5	3,17	0,33	10

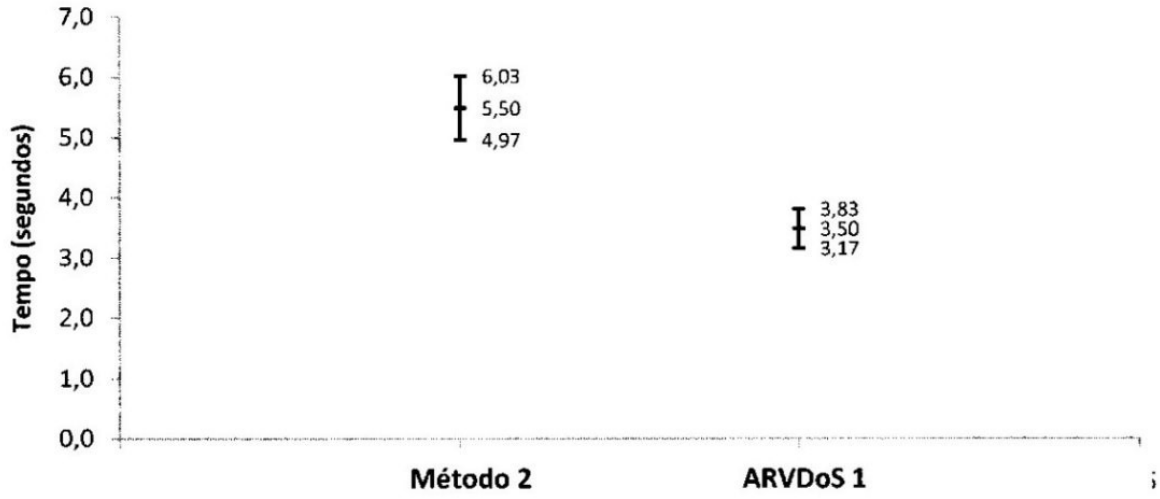


Figura 6.21. Comparação entre Tempo de Identificação e Reação na ARVDoS para o Cenário IV

7. CONCLUSÃO E TRABALHOS FUTUROS

A utilização da virtualização de redes na aplicação como solução para a próxima geração de redes Internet precisa ser estudada em diversas frentes. A possibilidade de criação de planos virtuais de redes sobre os recursos físicos pode prover um aumento da expansibilidade e segurança de aplicações sobre a internet e da mesma forma, criar redes virtualizadas privadas independentes, isoladas e escaláveis.

Na aplicação das VNs, os provedores de infraestrutura e os provedores de serviços agregam recursos de múltiplos e fornecem serviços fim a fim aos usuários finais, com transparência.

Além disso, a virtualização de redes permite a cada uma destas redes físicas e virtuais implementar o controle e o gerenciamento de forma individualizada. Assim, a implementação de ações de gerência e segurança nestes ambientes se aplica da mesma forma que em ambientes não virtualizados. Desta forma é possível afirmar que devido às suas características de empregar múltiplos métodos de detecção de ataques DoS, a arquitetura ARVDoS tem aplicabilidade tanto em redes físicas como em redes virtualizadas

Neste trabalho é apresentada a arquitetura ARVDoS para ambientes virtualizados de rede que atua de forma reativa para responder a ataques DoS. A ARVDoS apresenta duas características que atendem respectivamente a: escalabilidade de redes virtuais e independência às tecnologias de virtualização de redes.

A característica de escalabilidade tem fundamental importância na aplicação da arquitetura, permite assim que independente da quantidade de roteadores virtuais existentes na topologia da VN, suas atividades de reação serão executadas no acesso a rede, no ponto onde houve a identificação do ataque. Foi verificada a independência dos tempos de reação da arquitetura através de exemplos com diferentes tamanhos de topologia e diferentes quantidades de métodos a fim de observar que o mecanismo de reação ocorre localmente na interface que foi verificado o ataque, sem alterações das outras interfaces, sendo assim escalar.

A característica de independência às tecnologias de virtualização permite que sejam utilizadas tecnologias que são executadas em diferentes plataformas de hardware ou software.

As atividades da arquitetura são executadas por três agentes, os quais em suma, têm as funções de instanciar em uma interface da VN diferentes métodos de detecção de DoS, calcular o percentual de certeza de DoS de acordo com a SLA do usuário contratada e estabelecer uma nova configuração topológica com a instalação e configuração de tabelas de encaminhamento e enlace reativo sempre que um ataque DoS for identificado em qualquer segmento de uma VN sempre com a preocupação de continuar provendo o serviço aos seus usuários finais.

Em um nível macro a arquitetura propõe uma nova topologia adicionando equipamentos virtualizados a uma topologia de rede virtualizada de forma a isolar e reduzir o tráfego de negação de serviço que venha a atacar a VN. Em um nível mais específico ela proporciona a criação de novos NICs virtualizados, novos roteadores virtualizados assim como o estabelecimento de enlaces que restringem a banda para um tráfego identificado como atacante redirecionando este tráfego limitado para o destino.

Este redirecionamento não causa impactos no restante da VN, seu desempenho em relação à taxa de bits por segundo, tende a permanecer próxima ou igual ao tráfego medido antes de detectar DoS no segmento da rede.

Foi possível observar que a taxa média de transmissão de dados se mantém durante o tratamento de um ataque, pois a criação de outra rede virtualizada garante o isolamento das atividades. O isolamento é característica importante para avaliação da arquitetura, por se tratar de um conceito fundamental para VN.

Com os dados obtidos neste trabalho é possível observar que a Arquitetura suporta diferentes métodos, desde que o resultado possa ser representado por um valor entre zero (0) e um (1).

Estes métodos instanciados não interferem na reação da ARVDoS, assim os tempos de reação da arquitetura são independentes do(s) método(s) utilizado(s). Estes tempos de reação da arquitetura medidos para quatro (4) reações diferentes são de pequena variação, ± 10,68 segundos para a primeira reação com inicialização a de máquinas virtuais, configuração de rotas, interfaces e enlaces virtuais; ± 4,37 segundos para a segunda reação para alterações de filas e filtros com o objetivo de alteração dos enlaces; ± 4,58 segundos para finalização das configurações de reação e ± 3,5 segundos para configurações de envio para o buraco negro.

Desta forma é possível concluir que a arquitetura ARVDoS, em que seus agentes atuam de forma integrada, mostrou-se confiável e com boa regularidade nos ensaios e testes observados.

Este trabalho apresenta várias oportunidades de pesquisa e pode ser estendido em várias direções:

- Projeto de operação dinâmica de novos serviços de redes virtualizadas;
- Gerenciamento de recursos com a aplicação da arquitetura;
- Eficiência e performance de gerenciamento em VNs;
- Situações de segurança e novas ameaças para a virtualização de redes;
- Novos modelos econômicos de redes para provedores para infraestrutura de redes;
- Modelos de isolamento e gerenciamento de falhas;
- Tecnologias que oferecem suporte a virtualização;

A virtualização tem se apresentado como um facilitador para o gerenciamento eficiente em ambientes de rede (i.e. VLANs, VPNs, caminhos virtuais em redes MPLS). O recente desenvolvimento e a evolução em virtualização de hardware e software apresentam novas e promissoras perspectivas para redes mais flexíveis, construindo um caminho para novos e avançados serviços. Entretanto, para permitir esta evolução e torná-la uma das bases das tecnologias que darão suporte ao futuro da internet, muitos desafios precisam ainda ser abordados.

REFERENCIAS BIBLIOGRÁFICAS

1. **Blazek, R B, et al.** A novel approach to detection of "denial-of-service" attacks via adaptative sequencial and batch-sequencial change-point detection methods. *IEEE Systems man and Cybernetics Information Assurance Workshop.* June 2001.

2. **Carl e Kesidis.** Denial-of-service attack-detection techniques. *IEEE Distributed Systems Online.* Feb 2006.

3. **Anderson, T, et al.** Overcoming the Internet impasse in through virtualization. *Computer.* 2005, Vol. 38, 4, pp. 34-41.

4. **Feamster, N, Gao, L and Rexford, J.** How to lease the internet in your spare time. *SIGCOMM CCR.* 2007, Vol. 37, 1, pp. 61-64.

5. **Kulkami, A B, Bush, S F e Evans, S C.** *Detecting distributed denial-of-service attacks using Kolmogorov complexity metrics.* Dec de 2001, Technical Information Series. 2001CRD176.

6. **Mirkovic, J and Reiher, P** *A taxonomy of DDoS attack and DDoS defense mechanisms..* Apr 2004, ACM SIGCOMM Computer Communication Review., Vol. 34.

7. **Mosharaf, N M, Chowdhury, K and Boutaba, R.** A survey of network virtualization. *University of Waterloo Technical Report CS-2008-25.* Oct 2008.

8. **Mosharaf, N M, Chowdhury, K and Boutaba, R.** Network virtualization: the past, the present, and the future. *IEEE Communication MAgazine, In Evaluation.* Jan 2009.

9. **Mosharaf, N M, et al.** iMark: an identy management framework for network virtualization environment. *The 11th IFIP/IEEE International Symposium on Integrated Network Management (IM'2009).* Oct 2008.

10. **Turner, J and Taylor, D.** Diversifying the Internet. *GLOBECOM'05.* 2005, Vol. 2.

11. **Wang, Y, et al.** Virtual routers on the move: Live router migration as a network-management primitive. *ACM SIGCOMM.* 2008, pp. 231–242.

12. **IEEE, LAN/MAN Standards Committee.** IEEE standard for local and metropolitan area networks– virtual bridged local area networks. IEEE Std 802.1Q-2005, May 2006.

13. **Rosen E. and Rekhter Y.** BGP/MPLS IP Virtual Private Networks (VPNs). RFC 4364, February 2006.

14. **Rosen E. and Rekhter Y.** BGP/MPLS VPNs. RFC 2547, March 1999.

15. **Ferguson P. and Huston G.** What is a VPN? Technical report, Cisco Systems, 1998.

16. **Andersson L. and Madsen T.** Provider Provisioned Virtual Private Network (VPN) Terminology. RFC 4026, March 2005.

17. **Andersen, David. Balakrishnan, Hari. Kaashoek, Frans. Morris, Robert.** Resilient overlay networks. *SIGOPS Operating Systems Review*, 35(5):131–145, 2001.

18. **Savage, S. Anderson, T. Aggarwal, A. Becker, D. Cardwell, N. Collins. Hoffman, A. Snell, J. Vahdat, A. Voelker, G. and J. Zahorjan.** Detour: A case for informed Internet routing and transport. *IEEE Internet Computing*, 19(1):50–59, January 1999.

19. **Eriksson, Hans.** MBone: The multicast backbone. *Communications of the ACM*, 37(8):54–60, 1994.

20. **Chu, Yang. Rao, Sanjay. Seshan, Srinivasan. and Zhang, Hui.** Enabling conferencing applications on the internet using an overlay muilticast architecture. *SIGCOMM Computer Communication Review*, 31(4):55–67, 2001.

21. **Subramanian, Lakshminarayanan. Stoica, Ion. Balakrishnan, Hari. and Katz, Randy.** OverQoS: An overlay based architecture for enhancing internet QoS. In *Proceedings of the 1st Symposium on Networked Systems Design and Implementation (NSDI)*, pages 71–84, March 2004.

22. **Keromytis, A. Misra, V. and Rubenstein, D.** SOS: Secure overlay services. In *Proceedings of the ACM SIGCOMM Conference (SIGCOMM'02)*, August 2002.

23. **Andersen, David G.** Mayday: Distributed filtering for Internet services. In *Proceedings of the 4th conference on USENIX Symposium on Internet Technologies and Systems (USITS'03)*, Berkeley, CA, USA, 2003.

24. **Krishnamurthy, Balachander. Wills, Craig. and Zhang, Yin.** On the use and desempenho of content distribution networks. In *Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement (IMW'01)*, pages 169–182, 2001.

25. **Lua, Eng Keong. Crowcroft, J. Pias, M. Sharma, R. and Lim, S.** A survey and comparison of peer-to-peer overlay network schemes. *IEEE Communications Surveys & Tutorials*, 7(2):72–93, 2005.

26. **PlanetLab:** An open platform for developing, deploying, and accessing planetary-scale services. http://www.planet-lab.org/. Acesso em julho, 2009.

27. **Anderson, Thomas. Peterson, Larry. Shenker, Scott. and Turner, Jonathan.** Overcoming the Internet impasse through virtualization. *Computer*, 38(4):34–41, 2005.

28. **Boucadair, M. Levis, P. Griffin, D. Wang, N. Howarth, M. Pavlou, G. Mykoniati, E. Georgatsos, P. Quoitin, B. Rodriguez Sanchez, J. and Garcia-Osma, M.L.** A framework for end-to-end service differentiation: Network planes and parallel Internets. *IEEE Communications Magazine*, 45(9):134–143, September 2007.

29. **Zhu, Y. and Ammar, M.** Algorithms for assigning substrate network resources to virtual network components. In *Proceedings of IEEE INFOCOM*, 2006.

30. **CERT, Computer Emergency Response Team.** "Denial of Service Attacks". Disponível em http://www.cert.org/tech_tips/denial_of_service.html. Acesso em julho de 2009.

31. **CERT Incident NOTE IN-99-04.** Disponível em: http://www.cert.org/incident_notes/IN-99-04.html. Acesso em julho de 2009.

32. **Axelsson, S.** Intrusion detection systems: A survey and taxonomy. Technical Report 99-15, Department of Computer Engineering, Chalmers University, March 2000.

33. **Bernstein, D. J.** Syn cookies. http://cr.yp.to/syncookies.html. Acesso em agosto 2009.

34. **Ferguson, P. and Senie, D.** Network Ingress Filtering: Defeating Denial of Service Attacks which Employ IP Source Address Spoo¯ng. *RFC 2827*, May 2000.

35. **Lau, F. Rubin, S. H. Smith, M. H. and Trajkovic, L.** Distributed Denial of Service Attacks. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 2275{2280, Nashville, TN, USA, October 2000.

36. **Bellovin, S. M.** "ICMP Traceback Messages", Março 2000. Disponível em http://www.cs.columbia.edu/~smb/papers/draft-bellovin-itrace-00.txt. Acesso em: junho 2009.

37. **Ioannidis, J. and Bellovin, S.** "Implementing pushback: Router-based defense against DDos attacks." Symposium on Network and Distributed System Security (NDSS-02), 2002.

38. **Mirkovic, J. Prier, G. and Reiher, P.** "Attacking DDoS at the Source," 10th IEEE International Conference on Network Protocols, pp. 312-321, Novembro 2002.

39. **Thomas, R. Mark, B. Johnson, T. and Croall, J.** "NetBouncer: client-legitimacy-based highdesempenho DDoS filtering," DARPA Information Survivability Conference and Exposition, 2003. Proceedings Vol. 1, 22-24, pp. 14 – 25, Abri de 2003.

40. **Keromytis, D. Misra, V. and Rubenstein, D.** "SOS: An Architecture For Mitigating DDoS Attacks", IEEE Journal on Selected Areas in Communications (JSAC), Janeiro de 2004.

41. **Garg, A. and Reddy, A. L. N.** Mitigation of DoS attacks through QoS Regulation. *IWQOS workshop*, May 2002.

42. **Sourcefire.** *Snort: The Open Source Network Intrusion Detection System.* Disponivel em: http://www.snort.org. Acesso em: agosto de 2009.

43. **Mahajan, R. Bellovin, S. Floyd, S. Paxson, V. and Shenker, S.** Controlling high bandwidth aggregates in the network. *ACM Computer Communications Review*, 32(3), July 2002.

44. **ISI, Information Sciences Institute.** *Dynabone.* Disponível em http://www.isi.edu/dynabone/. Acesso em: agosto 2009.

45. **Gil, T. M. and Poletto, M.** MULTOPS: a data-structure for bandwidth attack detection 10*th Usenix Security Symposium*, August 2001.

46. **Cs3. Inc.** *MANAnet DDoS MANAnet™: Infrastructure-level DDoS Defense.* Disponível em: http://www.cs3-inc.com/mananet.html. Acesso em: Agosto, 2009.

47. **Arbor Networks.** *The Peakflow Platform.* Disponível em: http://www.arbornetworks.com. Acesso em: Agosto, 2009.

48. **Mirkovic, J.** *D-WARD: Source-End Defense Against Distributed Denial-of-Service Attacks.* PhD thesis, University of California Los Angeles, August 2003.

49. **Savage, S. Wetherall, D. Karlin, A. and Anderson, T.** *Practical Network Support for IP Traceback.* In *ACM SIGCOMM 2000*, August 2000.

50. **Darmohray, T. and Oliver, R.** *Hot spares for DDoS attacks.* Disponível em: http://www.usenix.org/publications/login/2000-7/apropos.html. Acesso em: agosto de 2009.

51. **Brooks, R. R.** *Disruptive Security Technologies with Mobile Code and Peer-to-Peer* 10. *Networks*, CRC Press, 2005.

52. **Jung, J. et al.**, *"Fast Portscan Detection Using Sequential Hypothesis Testing," Proc. IEEE Symp. Security and Privacy*, IEEE CS Press, 2004, pp. 211–225.

53. **Alves, C.C.** "Gráficos de controle CUSUM: um enfoque dinâmico para a análise estatística de processos". Dissertação de Mestrado em Engenharia de Produção, UFSC, 2003.

54. **Feinstein, L. et al.** *"Statistical Approaches to DDoS Attack Detection and Response," Proc. DARPA Information Survivability Conf. and Exposition,* vol. 1, 2003, IEEE CS Press, pp. 303–314.

55. **Mirkovic, J. et al.** *Internet Denial of Service: Attack and Defense Mechanisms*, Prentice Hall, 2005.

56. **Giampaolo, Bella.** *Retaliation Against Protocol Attacks.* Journal of Information Assurance and Security 3 (2008) 313-325.

57. **Cisco Systems.** *Defeating DDoS Attacks.* White Paper. Cisco System Inc. 2004.

58. **Silva, Ken. Scalzo, Frank. Barber, Piet.** *Anatomy of Recent DNS Reflector Attacks from the Victim and Reflector Point of View.* White Paper. Verysign, 2006.

59. **Luo, Xiapu. Chan, Edmond W. W. Chang, Rocky K. C.** *Detecting Pulsing Denial-of-Service Attacks with Nondeterministic Attack Intervals. Research Article.* Hindawi Publishing Corporation. EURASIP Journal on Advances in Signal Processing. Article ID 256821, 13 pages doi:10.1155/2009/25682. Volume 2009.

60. **Montgomery, D. C.** Introduction to Statiscal Quality Control. New York: Wiley, 2005.

61. **Patcha, A. and Park, J. M.** "An overview of anomaly detection techniques: existing solutions and latest technological trends," Elsevier Computer Networks, Vol. 51, Issue 12, 2007, pp. 3448–3470.

62. **Moreno, Victor. Reddy, Kumar.** Network Virtualization. Cisco Press. Pages 408. Print ISBN-10: 1-58705-248-2, Print ISBN-13: 978-1-58705-248-4. July, 2006.

63. **Sanfelippo, Salvatore.** Hping3 Documentation. Disponível em: http://www.hping.org/documentatio.php. 2010.

64. **Iperf.** University of Central Florida. Disponível em: http://www.noc.ucf.edu/Tools/Iperf/. 2003.

65. **Barford P.** et al., "A Signal Analysis of Network Traffic Anomalies," *Proc. ACM SIGCOMM Internet Measurement Workshop,* ACM Press, 2002, pp. 71–82.

66. **Lu Wei and Ghorbani Ali A.** Network Anomaly Detection Based on Wavelet Analysis. EURASIP Journal on Advances in Signal Processing. Volume 2009.

67. **Moore D., Voelker G.M., and Savage S.**, "Inferring Internet Denial-of-Service Activity," *Proc. Usenix Security Symp.,* Usenix Assoc., 2001.

68. **Jung J., Krishnamurthy B., and Rabinovich M.**, "Flash Crowds and Denial of Service Attacks: Characterization and Implications for CDNs and Web Sites," *Proc. Int'l World Wide Web Conference*, ACM Press, 2002, pp. 252–262.

69. **Wang H., Zhang D., and Shin K.**, "Detecting SYN Flooding Attacks," *Proc. 21st Joint Conf. IEEE Computer and Comm. Societies (IEEE INFOCOM)*, IEEE Press, 2002, pp. 1530–1539.