NOVAS PROPOSTAS, COM AVALIAÇÕES, DE TÉCNICAS PREVENTIVAS E REATIVAS CONTRA ATAQUES NA INTERNET E REDES DE PRÓXIMA GERAÇÃO

Renato Souza Silva

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia de Sistemas e Computação.

Orientador: Luís Felipe Magalhães de Moraes

Rio de Janeiro
Novembro de 2021

NOVAS PROPOSTAS, COM AVALIAÇÕES, DE TÉCNICAS PREVENTIVAS E REATIVAS CONTRA ATAQUES NA INTERNET E REDES DE PRÓXIMA GERAÇÃO

Renato Souza Silva

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Orientador: Luís Felipe Magalhães de Moraes

Aprovada por: Prof. Luís Felipe Magalhães de Moraes
Prof. Claudio Luis de Amorim
Prof. Felipe Maia Galvão França
Prof. Jorge Lopes de Souza Leão
Prof. Magnos Martinello
Prof. Nilton Alves Júnior

RIO DE JANEIRO, RJ – BRASIL
NOVEMBRO DE 2021

Silva, Renato Souza

Novas Propostas, com Avaliações, de Técnicas Preventivas e Reativas Contra Ataques na Internet e Redes de Próxima Geração/Renato Souza Silva. – Rio de Janeiro: UFRJ/COPPE, 2021.

XX, 152 p.: il.; 29, 7cm.

Orientador: Luís Felipe Magalhães de Moraes

Tese (doutorado) – UFRJ/COPPE/Programa de Engenharia de Sistemas e Computação, 2021.

Referências Bibliográficas: p. 133 – 149.

1. Sistema de Detecção de Intrusões. 2. BGP. 3. Fusão de Dados. 4. Plano de Controle. 5. DDoS. 6. Sinalização. 7. Teoria de Jogos. I. Moraes, Luís Felipe Magalhães de. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

Dedico cada linha deste trabalho à minha Mãe. Uma decisão "maior" a levou antes que eu pudesse lhe dizer pessoalmente o quanto ela foi importante nesta conquista em particular e na minha missão como ser humano. Antes disto tudo começar, o que mais me motivava a fazer o melhor não era ela exatamente, mas sim a minha própria felicidade em vê-la feliz e orgulhosa. Mal tinha noção na época da irrelevância do meu próprio sentimento, perto do imenso amor dela e das Mães em geral por seus filhos, sem precisar de mais nada...

Agradecimentos

Tenho tantas pessoas para agradecer que às vezes penso em simplesmente fazer algo genérico e inclusivo. Mesmo correndo o risco de me esquecer de pessoas importantes nesta conquista, devo sobretudo agradecer à minha esposa Cynthia pela paciência e pelo apoio incondicional em todos os momentos. Esta conquista, pela qual sofremos e lutamos juntos, é muito mais sua do que minha.

Agradeço às minhas Filhas Eduarda e Victória, simplesmente pelo fato de existirem. Foram sempre a minha fonte de inspiração e determinação.

Agradeço ao meu Pai por ter me ensinado desde cedo que a humildade derruba muros.

Agradeço ao meu Orientador Professor Luís Felipe por me ensinar a converter frustração em energia.

Agradeço aos membros desta banca que aceitaram participar da mesma.

Agradeço ao meu Irmão mais novo Evandro pelos momentos bons e ruins que dividimos juntos.

Agradeço aos meus colegas de trabalho, por segurar a barra pra mim quando eu precisava estudar para as provas.

Agradeço ao amigo Vitor, pela ajuda na fase final.

Agradeço ao pessoal do Programa, pelas orientações para vencer a burocracia.

Agradeço aos Professores Magnos Martinello , Giovanni Comarella, Moisés Ribeiro e Evandro Evandro Ottoni e todos os outros que fizeram parte desta jornada.

Agradeço ao Professor Otto (*in memorian*), pela amizade.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

NOVAS PROPOSTAS, COM AVALIAÇÕES, DE TÉCNICAS PREVENTIVAS E REATIVAS CONTRA ATAQUES NA INTERNET E REDES DE PRÓXIMA GERAÇÃO

Renato Souza Silva

Novembro/2021

Orientador: Luís Felipe Magalhães de Moraes

Programa: Engenharia de Sistemas e Computação

Os ataques cibernéticos desafiam a evolução das redes de comunicação de última geração como o 5G e 6G. Para se defenderem e a seus clientes, os provedores de serviços têm investido em sistemas tradicionais de detecção e mitigação. Embora eficientes, estes sistemas apresentam problemas, induzindo decisões incorretas, que podem comprometer tráfego legítimo. A estratégia de defesa em camadas oferece novas abordagens, dificultando a progressão do ataque através de múltiplas linhas de defesa. Esta tese apresenta dois sistemas avançados para detecção e mitigação de ataques cibernéticos. O sistema distribuído de detecção de intrusões se baseia na rede BGP para criar uma federação de agentes, que cooperam entre si para alarmar fluxos maliciosos na Internet. Métricas de desempenho obtidas analiticamente, combinando fusão de dados com inferência Bayesiana, mostram resultados comparáveis aos melhores sistemas. A modelagem proposta pode inclusive ser utilizada para avaliar o desempenho de outros sistemas distribuídos de detecção similares. Para prevenir ataques contra a própria plataforma de detecção, propõe-se um sistema de aprendizado de máquina para inferir sobre a reputação dos anúncios BGP. Testes baseados num *dataset* com 15 atributos extraídos individualmente do cabeçalho das mensagens, mostram que é possível aprender com acurácia acima de 90%. Na parte de mitigação, propõe-se um sistema baseado em Teoria de Jogos, capaz de conter os efeitos dos ataques de negação de serviços. O mecanismo de escalonamento de recursos usa o ambiente virtualizado do plano de controle 5G para balancear o tráfego, impedindo a imediata exaustão provocada pelo ataque e interrompendo-o por dissuasão. Testes de desempenho usando um modelo de filas demonstram que o sistema é capaz de reduzir a carga em até 20% a cada nível de escalonamento.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

NEW PROPOSALS, WITH EVALUATIONS, OF PREVENTIVE AND REACTIVE TECHNIQUES AGAINST ATTACKS ON THE INTERNET AND NEXT GENERATION NETWORKS

Renato Souza Silva

November/2021

Advisor: Luís Felipe Magalhães de Moraes

Department: Systems Engineering and Computer Science

Cyberattacks challenge the evolution of next-generation communication networks such as 5G and 6G. To defend themselves and their customers, service providers have invested in traditional detection and mitigation systems. Although they are efficient, these systems have problems, inducing incorrect decisions, which can compromise legitimate traffic. The Defense-in-Depth strategy offers new approaches, hampering the attack progression through multiple lines of defense. This thesis presents two advanced systems for detecting and mitigating cyberattacks. The distributed intrusion detection system is based on the BGP network to create a federation of agents, which cooperate with each other to alarm malicious flows in the Internet. Performance metrics obtained analytically, combining data fusion with Bayesian inference, show results comparable to the best systems. This proposed modeling can even be used to assess the performance of similar distributed systems. To prevent attacks against the detection platform itself, it is proposed a machine learning system to infer the reputation of BGP advertisements. Tests based on a *dataset* with 15 attributes extracted individually from the message headers show that it is possible to learn with accuracy above 90%. In the mitigation part, a system based on Game Theory is presented to confine the effects of denial of service attacks. The resource scaling engine uses the virtualized environment of the 5G control plane to balance traffic, preventing the immediate exhaustion caused by the attack and stopping it by deterrence. Performance tests using a queuing model demonstrate that the system is able to reduce the load by up to 20% at each scaling level.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Símbolos

$2^{\|\Omega\|}$	Conjunto potência, com todas as combinações possíveis do quadro de discernimento., p. 65
Δt	Intervalo de tempo, p. 62
Ω	Conjunto exaustivo que representa o quadro de discernimento das evidências, p. 64
Θ	Variável aleatória contínua $\Theta \in [0, 1]$, p. 19
α	Parâmetro concordante da função Beta, p. 60
β	Parâmetro conflitante da função Beta, p. 60
λ	Taxa total de transações que entram no plano de controle., p. 106
λ_a	Taxa de transações maliciosas que entram no plano de controle., p. 106
λ_l	Taxa de transações legítimas que entram no plano de controle., p. 106
μ	Taxa de processamento (serviço) dos vMMEs., p. 106

Lista de Acrônimos

$Bin(N_D, PPV_{av})$	Distribuição binomial com parâmetros (N,p), p. 66
C_a	Função de custo do atacante., p. 106
C_d	Função de custo do defensor., p. 106
I	Variável aleatória de Bernoulli que indica a ocorrência de uma intrusão num dado instante do tempo, p. 64
K	Capacidade máxima de armazenamento do sistema., p. 106
K_X	Índice de conflito entre as fontes de evidência, p. 65
M	Número total de *bots*., p. 106
M_{C_i}	Massa de crença consolidada do anúncio gerado pelo AS_i., p. 81
N	Número de *smartphones* legítimos., p. 106
$N(0,1)$	Distribuição normal entre 0 e 1, p. 18
N_D	Número total de IDSs federados que detectaram a intrusão, p. 63
N_F	Número total de IDSs federados, p. 63
N_I	Número total de IDSs federados atravessados pelo fluxo intruso, p. 63
Q	Tamanho da fila., p. 106
S	Número máximo de vMMEs (servidores)., p. 106
S_a	Função de estratégia do atacante., p. 106
S_d	Função de estratégia do defensor., p. 106
T	Tempo de espera em fila., p. 107

$U(0,1)$	Distribuição uniforme entre 0 e 1, p. 18
U_a	Função de ganho do atacante., p. 106
U_d	Função de ganho do defensor., p. 106
U_i	Variável aleatória de Bernoulli, que representa a detecção de uma intrusão pelo IDS_i, p. 62
m	Número de vMMEs (servidores) em operação., p. 106
$m_C(A)$	Massa de crença combinada das fontes de evidência sobre a hipótese A, p. 65
$m_i(A)$	Massa de crença da fonte de evidência *i* sobre a hipótese A, p. 65
n	Número total de *smartphones*., p. 106
q	Tamanho do *buffer* do servidor., p. 106
t	Instante de tempo que ocorreu uma intrusão, p. 62
3GPP	*3rd Generation Partnership Project*, p. 28
AIS	*Artificial Immune System*, p. 38
AMF	*Access & Mobility Management Function*, p. 29
AS	*Autonomous System*, p. 53, 56, 62
BGP	*Border Gateway Protocol*, p. 56, 62
Bel	Função Crença, p. 22
CERT	*Cyber Emergency Response Teams*, p. 54
CMIP	*Common Management Information Protocol*, p. 39
CRIM	*Cooperative Intrusion Detection Module*, p. 40
DAI/DAO	*Data in/Data out*, p. 21
DDoS	*Distributed Denial of Services*, p. 6, 53
DIDS	*Distributed Intrusion Detection System*, p. 3, 5, 38, 55
DNS	*Domain Name Service*, p. 29, 42
DSCP	*DiffServ Code Point*, p. 15

DST	*Dempster-Shafer Theory*, p. 22
DiD	*Defense-in-Deppth*, p. 2
DoS	*Denial of Services*, p. 40, 53
E-UTRAN	*Universal Terrestrial Radio Access Network*, p. 114
EPC	*Evolved Packet Core*, p. 101
EU	Equipamento do Usuário, p. 114
FN_{ij}	Número de alarmes falso-negativos do IDS_i, testado no *dataset* j, p. 61
FNR_{DIDS}	Taxa de alarmes falso-negativos da plataforma DIDS, p. 67
FNR_i	Taxa de alarmes falso-negativos do IDS_i, p. 62
FNR	*False Negative Rate*, p. 54, 62
FN	*False Negatives*, p. 61
FP_{ij}	Número de alarmes falso-positivos do IDS_i, testado no *dataset* j, p. 61
FPGA	*Field Programmable Gate Array*, p. 41
FPR_{DIDS}	Taxa de alarmes falso-positivos da plataforma DIDS, p. 67
FPR_i	Taxa de alarmes falso-positivos do IDS_i, p. 62
FPR	*False Positive Rate*, p. 54
FP	*False Positives*, p. 61
HIDS	*Host-based Intrusion Detection Systems*, p. 54
HIS	*Human Immune System*, p. 38
HMM	*Hiden Markov Model*, p. 44
ICMP	*Internet Control Message Protocol*, p. 79
IDS_i	IDS membro da federação, p. 68
IDS	*Intrusion Detection Systems*, p. 1, 54
IGP	*Exterior Gateway Protocol*, p. 25

IGP *Interior Gateway Protocol*, p. 25

IPS *Intrusion Prevention Systems*, p. 1

IP *Internet Protocol*, p. 79

ISPs *Internet Service Providers*, p. 53

ITS *Inteligent Transport Systems*, p. 23

IoT *Internet of Things*, p. 23, 47

LAN *Local Area Network*, p. 39

LSTM *Long Short-Term Memory*, p. 46

LTE *Long-Term Evolution*, p. 42

MAIDA *Multi Agent Intrusion Detection Architecture*, p. 39

MEC *Mobile Edge Computing*, p. 42

MIMO *Multiple-Input/Multiple-Output*, p. 100

MLB MME *Load Balance*, p. 48

MLP *Multi-Layer Perceptron*, p. 46

MME *Mobility Management Entity*, p. 101

MMP MME *Processing*, p. 48

NGN *Next Generation Networks*, p. 41

NIC *Network Interface Card*, p. 41

NIDS *Network-based Intrusion Detection Systems*, p. 5, 54

NLRI *Network Layer Reachability Information*, p. 56

NPV *Negative Prediction Value*, p. 78, 129

NSA *Negative-Selection Algorithm*, p. 38

NTP *Network Time Protocol*, p. 29

OAISIM *Openairinterface Simulator*, p. 114

P2P *Peer-to-Peer*, p. 40

PDF	*Probability Density Function*, p. 19
PGP	*Pretty Good Privacy*, p. 40
PPV_{DIDS}	Valor do PPV da plataforma DIDS, p. 66
PPV_{av}	Valor médio de PPV de todos os IDSs federados, p. 66
PPV_i	Valor do PPV do IDS_i, p. 65
PPV	*Positive Prediction Value*, p. 61, 64
PR_{av}	Taxa média de detecção positiva, p. 68
PR_i	*Positive Rate do* IDS_i, p. 69
$PRE(i)$	Reputação do AS_i, p. 87
Pls	Plausibilidade, p. 22
RAN	*Radio Access Network*, p. 101
$REP(i)$	Representatividade do AS_i, p. 87
RFC	*Request for Change*, p. 15
ROC	*Receiver Operating Characteristic*, p. 74, 77
$RPT(i)$	Reputação do $AS_i \in [0, 1]$., p. 81
S1U	Código da interface entre a enodeB e o plano de dados do EPC, p. 103
SBA	*Service-Based Architecture*, p. 28
SMF	*Session Management Function*, p. 29
SON	*Self-Organising Networks*, p. 100
SVM	*Support Vector Machines*, p. 44
TN_{ij}	Número de alarmes verdadeiro-negativos do IDS_i, testado no *dataset* j, p. 61
TNR_{DIDS}	Taxa de alarmes verdadeiro-negativos da plataforma DIDS, p. 67
TNR_i	Taxa de alarmes verdadeiro-negativos do IDS_i, p. 62
TNR	*True Negative Rate*, p. 54, 62

TN	*True Negatives*, p. 61
TP_{ij}	Número de alarmes verdadeiro-positivos do IDS_i, testado no *dataset* j, p. 61
TPR_{DIDS}	Taxa de alarmes verdadeiro-positivos da plataforma DIDS, p. 67
TPR_i	Taxa de alarmes verdadeiro-positivos do IDS_i, p. 62
TPR	*True Positive Rate*, p. 54, 62
TP	*True Positives*, p. 61
UE	*User Equipment*, p. 48, 102
URLLC	*Ultra-Reliable Low-Latency Communication*, p. 102
VA	Variável Aleatória., p. 19
VM	*Virtual Machine*, p. 114
VNF	*Virtual Network Function*, p. 49
WAN	*Wide Area Network*, p. 79
XML	*Extensible Markup Language*, p. 40
mRMR	*minimum Redundancy Maximum Relevance*, p. 45
vEPC	*virtualized Evolved Packet Core*, p. 102
vHSS	*virtual Home Subscriber Server*, p. 103
vSPGW	*virtual Serving Packet Gateway*, p. 103

Capítulo 1

Introdução

Os sistemas de segurança vêm sendo utilizados há muito tempo para proteger ativos contra ataques cibernéticos. Nos tempos passados, quando um dos principais problemas era para proteger arquivos confidenciais, surgiu o primeiro sistema de segurança para controlar o acesso a estes arquivos através de senhas [3]. Entretanto, com o surgimento da Internet nos anos 60 e a disseminação da conectividade entre computadores, a segurança se tornou uma das maiores preocupações e os sistemas de segurança precisaram se adaptar às novas ameaças que surgiram com a conectividade. Desta necessidade, exemplificada na prática em 1986 com um ataque russo para roubar informações secretas do Pentágono, surgiu em 1990 o primeiro *firewall* [4]. Desde então, os sistemas de segurança vêm evoluindo continua, mas reativamente para tentar acompanhar a diversidade e a complexidade dos ataques cibernéticos, que surgem praticamente todos os dias. Os sistemas de detecção (IDS) e prevenção de intrusões (IPS) são exemplos de sistemas de segurança reativos, mas que também podem atuar proativamente, dependendo da sua estratégia de operação. Sistemas distribuídos que operam com largas superfícies de detecção podem ser considerados proativos, pois são capazes de detectar uma nova ameaça e impedir a ocorrência de um ataque, partindo do pressuposto de que uma ameaça ainda desconhecida numa parte do globo, pode ser bem conhecida e já contar com medidas de proteção mapeadas num outro ponto geograficamente distante.

Os IDSs e IPSs são sistemas que geralmente atuam de forma complementar. Enquanto os IDSs monitoram parâmetros do sistema para detectar eventos suspeitos, os IPSs operam a partir desta sinalização para atuar de forma pré-programada, dependendo do alarme enviado pelo IDS, para bloquear ou mitigar os possíveis efeitos do evento. O IPS pode ser entendido como a combinação das funcionalidades do IDS com as de um *firewall* dinâmico, cujas configurações de segurança podem ser alteradas dinamicamente (*online*), de acordo com as características do evento malicioso detectado. Os IPSs ainda podem atuar diretamente em *firewalls* ou roteadores de borda, traduzindo os alarmes provenientes de um IDS para comandos de bloqueio

específicos, a serem aplicados direta e automaticamente nestes elementos [5]. Entretanto, apesar dos inúmeros avanços tecnológicos, tanto na parte de detecção quanto nas estratégias de prevenção, atuações automáticas rigorosas, como bloquear um tráfego legítimo baseado num alarme falso-positivo do IDS, ainda são muito comuns e representam um grave risco ao funcionamento dos sistemas. Além disto, as estruturas monolíticas destes sistemas que operam dentro de um perímetro relativamente pequeno, os transformam muitas vezes nos próprios alvos do ataque.

Os sistemas avançados de segurança são baseados na estratégia de defesa militar conhecida como *Defense-in-Depth* (DiD), ou defesa em camadas. A estratégia DiD parte do princípio que nenhum sistema de segurança é perfeito e sempre haverá uma forma do atacante superá-lo, dependendo dos recursos e da motivação. Ao invés de uma tentativa de bloqueio imediata, a ideia básica desta estratégia é aumentar o custo da progressão do atacante com múltiplas linhas de defesa, que são distribuídas em camadas independentes e com diferentes níveis táticos de proteção. Os sistemas avançados correspondem justamente à estas camadas de linha frente, que têm a função de reconhecer ou atrasar o ataque, retirando seu *momentum* e garantindo o tempo necessário para fortalecer as camadas subsequentes [6–8]. Quanto maior e mais importante o bem ativo a ser protegido, maior o número de camadas a serem transpostas pelo suposto agressor [9] para atingir o seu alvo. A estratégia DiD de distribuir o sistema de segurança em camadas independentes também funciona para permitir reações de proteção imediatas e aplicadas, à medida que as camadas são superadas. Esta estratégia também é eficiente para restringir o funcionamento de mecanismos conhecidos como “cavalos de troia", mantendo o risco em níveis aceitáveis [10]. A Figura 1.1 abaixo oferece uma visão gráfica desta estratégia, aplicada ao mundo da segurança de informação, onde o bem a ser protegido são os próprios dados do sistema.

A principal vantagem desta abordagem holística em relação aos sistemas tradicionais é a sua capacidade combinada de oferecer níveis mais elevados de segurança, podendo, contudo, controlar eventuais impactos indesejados nos serviços, durante o exercício da segurança. Como a segurança de qualquer sistema não é maior que seu elo mais fraco, esta estratégia garante que, se uma camada anterior é superada pelo atacante, outras camadas poderão atuar de forma complementar, reagindo mais adequadamente e cumprindo a missão de proteger o sistema.

Consoante a taxonomia proposta em [11], os sistemas avançados de segurança podem ser classificados como reativos ou proativos (ou preventivos). Os sistemas reativos são considerados medidas de segurança básicas, que se concentram na construção de defesas contra riscos de segurança cibernética conhecidos e em contramedidas para resistir a estes. Os sistemas reativos partem do princípio que não se conserta o que já está quebrado, considerando que o tempo que o invasor leva

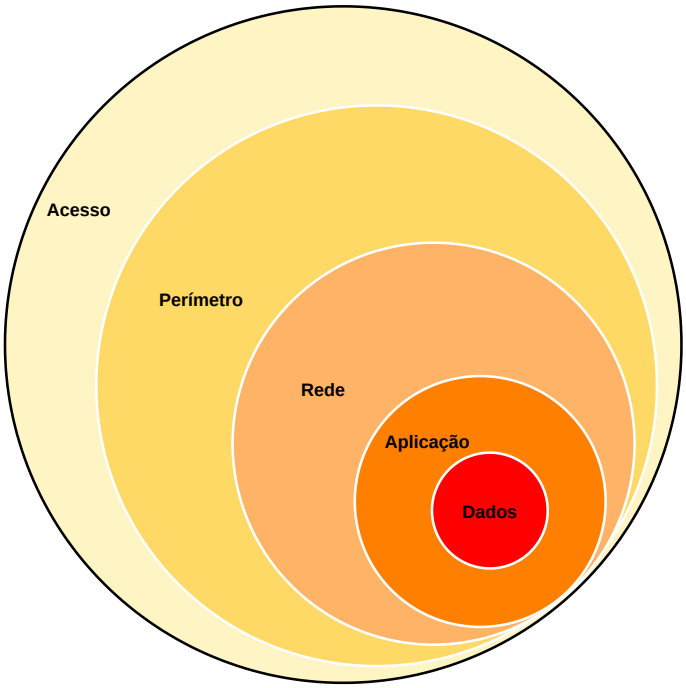


Figura 1.1: Diagrama de camadas de defesa segundo estratégia *Defense-in-Depth*. As diferentes cores representam os diferentes níveis de proteção.

para invadir o sistema e iniciar o ataque é maior que o necessário para detectá-lo e detê-lo. Em outras palavras, não há motivos para alarmes até que seja detectado realmente haver algo errado. Os sistemas de defesa reativos são disparados assim que um ataque é detectado e têm como principal objetivo mitigar os danos no sistema, bloqueando-o ou minimizando seu efeito através de contra-medidas restritivas. Para que funcionem de maneira adequada, sem comprometer serviços legítimos, os sistemas reativos precisam ser ajustados ao tipo e à intensidade do ataque. O IDS e o IPS são dois mecanismos de defesa reativos, que têm por objetivo detectar e aplicar contra-medidas no caso de um ataque, respectivamente. Assim, pode-se entender que os principais desafios dos sistemas reativos são:

- Detectar rápida e precisamente o ataque.
- Bloquear o ataque sem comprometer o tráfego legítimo.

A segurança proativa se propõe a impedir ou atrasar eventos de ataque. Ao contrário da segurança reativa, que se concentra nas ameaças que já entraram na sua rede, a segurança proativa se concentra em corrigir qualquer vulnerabilidade que torne a rede suscetível a ataques, antes que sejam exploradas pelo atacante. Os sistemas proativos têm como principal objetivo preparar o sistema de modo a minimizar os impactos de um ataque, ou mesmo evitá-los antes que aconteçam [12]. Dentro da estratégia *DiD*, os sistemas distribuídos de detecção de intrusões (DIDSs) podem ser considerados sistemas proativos, uma vez que são capazes de informar às

camadas posteriores sobre algum eventual risco, permitindo reações mais adequadas posteriormente. Um outro exemplo de sistema proativo é a resiliência. A resiliência está relacionada com a capacidade do sistema em permanecer funcionando, mesmo sob ataque. Configurações e projetos redundantes e bem atualizados são exemplos de medidas para aumentar a resiliência do sistema. Embora sejam imprescindíveis na composição da arquitetura de segurança, a constante evolução dos ataques e a diversificação de vetores derrubam a eficiência dos sistemas proativos, tornando-os desatualizados e fáceis de serem superados.

Este trabalho de doutorado propõe dois sistemas avançados de defesa híbridos, podendo atuar de forma reativa ou proativa na detecção e mitigação de ataques. Isto é, sob a perspectiva de atuarem detectando e mitigando ataques já em andamento, os sistemas podem ser considerados reativos. Entretanto, no conceito de *Defense-in-Depth*, estes mesmos sistemas podem funcionar proativamente como linhas de defesa avançadas, detectando o que poderia se transformar num ataque novo, ou aumentando a resiliência do sistema, de modo a mitigar seus efeitos e manter o sistema disponível durante um certo tempo. Adicionalmente, esta tese também apresenta um sistema baseado em aprendizado de máquina para reduzir os riscos de ataques internos contra a plataforma distribuída de detecção de intrusões.

O Capítulo 4 apresenta um sistema distribuído de detecção de intrusões formado por IDSs membros autônomos que cooperam entre si, como numa federação. Na estratégia *Defense-in-Depth*, este sistema seria uma primeira camada de detecção, ampliando-a suficientemente visando antecipar informações sobre um possível ataque, para que as próximas camadas e o próprio sistema de defesa como um todo possa manter os níveis de segurança adequados. O sistema considera a permeabilidade do protocolo BGP para formar uma rede sobreposta, capaz de interligar coerentemente todos os IDSs membros, estabelecendo assim a federação. A proposta se inspira na própria essência de conectividade plena da Internet, que permite aos atacantes encontrarem seus alvos em qualquer lugar da Internet, para aumentar a probabilidade de detecção, à medida que o fluxo intruso avança na direção do seu alvo. O sistema proposto é particularmente útil aos provedores de acesso à Internet, que além de terem seus próprios ativos em constante risco, também são responsáveis por “transportar"os fluxos ofensores aos seus próprios clientes.

Relacionado à primeira proposta do sistema distribuído de detecção de intrusões baseado em federação apresentado em detalhes no Capítulo 4, o Capítulo 5 apresenta um sistema para reduzir riscos de ataques internos contra a própria plataforma distribuída ou que a mesma seja utilizada como vetor de outros ataques. O sistema proposto se baseia num modelo de aprendizado de máquina supervisionado para inferir sobre a massa de crença individual dos anúncios BGP que chegam no AS de destino para serem combinados. Também é proposto um *dataset* composto por

15 atributos diretos e indiretos, extraídos individualmente de cada anúncio. Os testes não-supervisionados, executados ainda sem os rótulos, mostram ser possível separar os dados do *dataset* em dois conjuntos bem distintos entre si. Apesar de ainda não ser possível afirmar que se trata efetivamente dos conjuntos esperados "ataque" e "normal", este resultado indica duas claras tendências de aglomeração. Os testes supervisionados, utilizando rótulos extraídos da combinação com outro *dataset*, mostram resultados de acurácia e sensibilidade bastante elevados (maior que 90%), apesar da precisão razoável (68%).

A proposta descrita no Capítulo 6, considera as facilidades do núcleo virtualizado do 5G para propor um sistema de mitigação baseado no escalonamento de recursos, que ficariam ociosos durante a operação normal, como num plano de seguros. Ao invés de aplicar medidas duras de bloqueio, que podem inclusive comprometer o tráfego legítimo, o sistema propõe um jogo não cooperativo entre atacante e defensor para definir os melhores momentos para escalar os recursos do plano de controle de modo a absorver todo o tráfego. Ainda no conceito de *DiD*, além de garantir algum tempo precioso para adaptar as camadas de defesa posteriores no processo de mitigação, a abordagem proposta ainda oferece a possibilidade de frustrar o atacante, a ponto dele desistir à medida que aumenta a relação custo/benefício do seu intento.

1.1 Motivações e Definição dos Problemas

A principal motivação para esta tese é o risco e a ameaça crescentes que os ataques cibernéticos impõem aos sistemas que se baseiam na Internet [13]. Uma outra motivação importante é o desenvolvimento de novas tecnologias para as redes de próxima geração como o 5G e o 6G, que oferecem perspectivas inovadoras para ataque o problema dos ataques cibernéticos.

Os problemas endereçados no Capítulo 4 estão relacionados com o baixo desempenho de detecção dos IDSs baseados em análise de comportamento da rede (NIDS) e as vulnerabilidades dos sistemas tradicionais monolíticos de detecção de intrusões. Os NIDS baseados em anomalias conseguem detectar intrusões a partir de comparação de padrões no comportamento da rede. Entretanto, apesar dos avanços tecnológicos na combinação com métodos de aprendizado de máquina e da capacidade deste tipo de IDS em conseguir detectar ataques novos (*zero-day attacks*), o desempenho de detecção ainda é considerado baixo. Um segundo problema, que ocorre em função da arquitetura monolítica dos sistemas de detecção de intrusões tradicionais, é a sua vulnerabilidade sistêmica como único ponto de falha. Muitas vezes esta vulnerabilidade é explorada por atacantes que desejam comprometer o funcionamento do IDS para então prosseguir furtivamente com ataques mais elabo-

rados contra outros alvos [14]. Abordagens promissoras para contornar estes problemas indicam uma direção na utilização de redes de IDSs distribuídos (DIDS), que cooperam entre si para aumentar o desempenho de detecção. Entretanto, apesar de existirem inúmeras propostas de IDSs distribuídos, ainda existem uma série de desafios importantes a serem superados:

- A interconexão entre os agentes, permitindo que a federação se forme de maneira segura e robusta.
- A escalabilidade da plataforma de detecção. A relação custo/benefício em fazer parte da federação devem ser a menor possível para incentivar seu crescimento.
- Aspectos de segurança para reduzir os riscos de ataques internos.

O problema endereçado no Capítulo 5 diz respeito à segurança do próprio sistema distribuído de detecção de intrusões proposto no Capítulo 4, para reduzir os riscos de um ataque contra sua própria infraestrutura ou que a federação seja utilizada como vetor para outros tipos de ataques. Alarmes falso-positivos provenientes de ASs maliciosos podem comprometer o desempenho da plataforma de detecção, induzindo decisões incorretas e/ou causando exaustão nos processos de combinação.

Os novos serviços de arrendamento de infraestrutura de rede para múltiplos locatários (*multi-tenancy*) e do fatiamento baseado em serviços (*multi-slicing*) elevam a importância da disponibilidade do plano de controle como um dos principais bens a serem protegidos na rede 5G. Neste contexto altamente heterogêneo e denso, cada vez mais propício para o desenvolvimento de novos tipos de ataques, o caráter furtivo e de difícil mitigação do ataque de DDoS o eleva a um patamar superior de grave ameaça. Em se tratando de um ecossistema tão complexo e importante dentro do sistema de comunicação 5G, a maioria dos sistemas de mitigação existentes ou são restritos a algum tipo de ataque específico, ou implementam contra-medidas de bloqueio/restrição mais duras, correndo o risco de comprometer o tráfego legítimo, principalmente no caso de um alarme falso-positivo. Além disto, uma vez superados os bloqueios e contando com prerrogativas de usuário interno, o atacante tem à sua frente um campo aberto para explorar, causando efeitos devastadores ao sistema como um todo.

1.2 Principais Contribuições

A contribuição geral deste trabalho de doutorado é a proposta de dois sistemas de defesa baseados na estratégia DiD, que endereçam problemas típicos relacionados com a detecção e a mitigação de ataques cibernéticos em dois ambientes críticos. Os sistemas de segurança propostos podem funcionar como linhas de defesa avançadas,

aumentando o poder de detecção de ameaças e mitigando os efeitos de um ataque, sem comprometer o tráfego legítimo, respectivamente.

A principal contribuição do Capítulo 4 é a proposta de uma plataforma de arquitetura distribuída para detecção de intrusões, formada por IDSs federados que cooperam entre si para aumentar o desempenho de detecção da plataforma como um todo. A proposta considera a autonomia da Internet e a amplitude do protocolo BGP como pilares para expandir a superfície de detecção e poder escalar mundialmente para potencialmente se transformar num serviço público de Internet.

O Capítulo 4 propõe um modelo matemático que combina inferência Bayesiana com a teoria de fusão de dados de Dempster-Shafer para avaliar o desempenho de sistemas de detecção distribuídos. Mais do que isso, a utilização da teoria de fusão de dados permite que sejam combinadas informações diversas, colhidas em diferentes domínios e de IDSs diferentes, caracterizando a autonomia e heterogeneidade de cada sistema autônomo da Internet em definir seu próprio sistema de detecção. Neste modelo, a função de distribuição Beta é utilizada para modelar o conhecimento prévio da probabilidade de detecção média (probabilidade à priori) da federação de IDSs, baseado em dados experimentais (*datasets*), modulados com as hipóteses de detecção avaliadas (verossimilhança).

O capítulo 3 endereça um dos problemas mais sérios quando se fala numa rede de agentes distribuídos geograficamente e independentes que cooperam mutuamente para detectar intrusões, que é a segurança interna e/ou auto-proteção. No Capítulo 5 é proposto um modelo de aprendizado de máquina para inferir sobre a massa de crença dos anúncios BGP que chegam para serem combinados no AS de destino. A informação da massa de crença tem como principal objectivo mitigar os riscos de um ataque contra a própria arquitetura do DIDS, ou sua utilização como vetor de ataque. Além do próprio modelo, baseado em aprendizado supervisionado através de uma rede neural, o Capítulo 5 também propõe a estrutura de um *dataset* para treinar o modelo, cujos atributos podem ser extraídos individualmente dos anúncios, dispensando o processamento em lotes.

O Capítulo 6 propõe uma abordagem inovadora baseada no conceito das companhias de seguros para preservar os serviços do plano de controle do 5G, e potencialmente mitigar os ataques de DDoS de sinalização, através do escalonamento inteligente de recursos virtualizados na nuvem, que ficariam ociosos durante o regime normal de operação. A análise das tendências de comportamento no sistema é feita através de um modelo analítico baseado em Teoria de Jogos para encontrar os pontos de equilíbrio que irão determinar (*i*) os melhores momentos para iniciar o escalonamento de recursos pelo ponto de vista do defensor e (*ii*), para inferir sobre o comportamento do agressor, através da relação custo/benefício do ataque. A avaliação de desempenho do sistema é feita utilizando-se um modelo de filas para

reproduzir o cenário de congestionamento do plano de controle e os efeitos do balanceamento de carga, extrapolando os resultados dos testes de emulação. Os dados de entrada dos modelos analíticos são obtidos através de um protótipo virtualizado de EPC (vEPC), capaz de emular o funcionamento dos mecanismos de sobrecarga do LTE para balancear o tráfego da rede de acesso durante o ataque de DDoS.

1.3 Organização do Trabalho

O restante deste trabalho de doutorado está organizado em 5 capítulos, que descrevem detalhadamente os sistemas de defesa propostos.

- Uma abrangente fundamentação teórica e prática aplicada ao entendimento desta tese é oferecida no Capítulo 2. Para os leitores que já estiverem familiarizados com os itens que compõem esta fundamentação, também é oferecido um resumo objetivo, enfatizando os principais pontos de interesse.

- O Capítulo 3 faz uma extensa e exaustiva revisão bibliográfica da tese e monta um panorama abrangente e objetivo do estado da arte referente aos Capítulos 4 e 6.

- No Capítulo 4, a ameaça dos ataques de DDoS contra os provedores de serviços é consolidada, junto com as dificuldades para se protegerem e a seus clientes, utilizando sistemas de detecção de intrusões típicos. Neste contexto, é apresentada uma solução distribuída para detecção de intrusões que funciona de forma cooperativa, aproveitando a essência de conectividade da Internet proporcionada pelo BGP para aumentar a superfície de detecção e melhorar o desempenho.

  - Na Seção 4.1, são apresentadas as bases técnicas/tecnológicas da arquitetura proposta e o seu princípio de funcionamento.
  - A Seção 4.2 detalha a estrutura do modelo matemático utilizado para avaliar o desempenho da plataforma proposta a partir da análise das métricas de detecção.
  - Os resultados das métricas modeladas são apresentados na Seção 4.3, através de gráficos, com parametrizações estratégicas para enfatizar os diferentes aspectos do desempenho.
  - Na Seção 4.4 é apresentada uma análise conjuntural dos resultados e uma análise comparativa, em relação aos sistemas típicos.

  - Para comprovar a viabilidade técnica da proposta, um modelo de emulação é proposto na Seção 4.5, mostrando o funcionamento do protocolo FlowSpec, como a rede sobreposta que interliga os membros da federação.

- No Capítulo 5 é apresentado um sistema baseado em aprendizado de máquina para inferir sobre a massa de crença dos anúncios BGP que chegam ao AS de destino para serem combinados e darem origem a uma informação de intrusão consolidada. O cálculo da massa de crença a partir de informações contidas no próprio anúncio BGP pode evitar ataques internos a partir de decisões incorretas causadas por alarmes falso-positivos ou falso-negativos. Neste capítulo também é proposto um *dataset* com atributos extraídos individualmente da mensagem BGP, dispensando o processamento em lotes.

  - A Seção 5.1 discorre sobre o conjunto de dados (*dataset*) como item crítico no processo de aprendizado de máquina. Em particular, são apresentados detalhes sobre a montagem do *dataset* utilizado para treinar o modelo de aprendizado proposto neste Capítulo 5.
  - A Seção 5.2 descreve cada um dos 3 atributos extraídos diretamente de cada um dos anúncios BGP que chegam para serem combinados no AS de destino e que compõem o conjunto de dados utilizado para treinar o modelo de aprendizado proposto.
  - Os 12 atributos indiretos, extraídos a partir das informações dos anúncios BGP, são descritos em detalhes na Seção 5.3.
  - Os testes do *dataset* em algoritmos de aprendizado não-supervisionados são apresentados na Seção 5.4, inclusive com os resultados que permitem observar o nível de separação nos modelos de aglomeração utilizados: K-médias (Seção 5.4.1) e hierárquica (Seção 5.4.2).
  - A Seção 5.5 descreve os testes supervisionados, executados a partir do conjunto de dados rotulado, cujo processo é detalhado na Seção 5.5.1. Uma breve introdução sobre Redes Neurais é apresentada na Seção 5.5.2. O modelo de rede neural utilizado para inferir sobre a massa de crença dos anúncios é descrito na Seção 5.5.3. Finalmente, os resultados obtidos com o treinamento do *dataset* rotulado sobre o modelo de rede neural utilizado são apresentados e analisados na Seção 5.5.4.

- No Capítulo 6, A ameaça dos ataques distribuídos de negação de serviços contra o plano de controle dos sistemas móveis celulares é consolidada, em particular para o sistema 5G, considerando a novas funcionalidades de compartilhamento de infra-estrutura para diferentes usuários e serviços. Também

são apresentados alguns detalhes inciais da proposta de mitigação dos efeitos destes ataques, escalonando recursos virtualizados do plano de controle do 5G para balancear o tráfego e evitar imediatas situações de exaustão.

  - O estado da arte e os trabalhos relacionados com o sistema proposto no Capítulo 6 são analisados na Seção 3.3.1. Também a mostrada uma tabela compacta, relacionando os principais pontos de cada trabalho em relação ao sistema proposto.
  - A Seção 6.1 aborda os principais pontos da arquitetura proposta, detalhando a arquitetura e o funcionamento do sistema, contextualizando com os problemas endereçados.
  - Os detalhes da concepção e montagem dos modelos matemáticos utilizados para (*i*) avaliar o desempenho da proposta - Seção 6.2.1 e (*ii*) analisar o comportamento dos agentes durante o período do ataque - Seção 6.2.2 são apresentados na Seção 6.2.
  - A Seção 6.3 descreve o modelo experimental construído para testar o funcionamento do balanceamento de carga entre os vMMEs e o roteiro utilizado para coletar dados de entrada para os modelos matemáticos.
  - O desempenho do sistema de mitigação no escalonamento de recursos - Seção 6.4.1, bem como a tendência comportamental dos agentes - Seção 6.4.2 são apresentados e analisados na Seção 6.4.

- O Capítulo 7 fecha a Tese de Doutorado com uma visão geral dos sistemas propostos dentro o contexto tecnológico atual que se aplicam. Algumas projeções de trabalhos futuros na linha de sistemas avançados que podem ser desenvolvidos a partir das ideias contidas neste documento são propostas como uma forma de motivar novos estudos, dando continuidade a esta tese.

Capítulo 2

Fundamentação

Neste Capítulo 2, são apresentadas fundamentações teóricas e práticas com o objetivo de facilitar o entendimento dos leitores para o restante desta tese. No final deste capítulo é apresentada a Seção 2.13, com um resumo geral, discorrendo apenas sobre os principais pontos utilizados no desenvolvimento do trabalho.

2.1 Sistema de Detecção de Intrusões - IDS

A intrusão é considerada como uma das fases de um ataque cibernético. Pode ser definida como qualquer tentativa desautorizada de ultrapassar as barreiras de proteção para atentar contra a integridade, disponibilidade ou confidencialidade de um sistema. A intrusão pode ser interna ou externa, dependendo se é originada dentro ou fora do perímetro de segurança da rede. O IDS é um sistema de segurança que monitora o perímetro da rede (exemplo: tráfego da rede, sistema operacional dos computadores, bancos de dados, aplicações) e alarma quando encontra alguma atividade considerada suspeita. Existe uma grande diversidade de IDSs e muitas são as propostas de taxonomia para classificar toda esta diversidade [15–19]. A Figura 2.1 logo abaixo apresenta uma destas propostas.

Apesar das inúmeras abordagens propostas, que diferem basicamente nos métodos de obtenção e análise dos dados de auditoria, a arquitetura geral de um IDS se baseia em quatro componentes: geradores de eventos, analisadores de eventos, unidades de resposta e banco de dados de eventos.

- Auditoria - Coleta de dados e extração criteriosa do vetor de medidas (*features*) cuidadosamente definido para gerar os eventos, de acordo com o objetivo do IDS. Número de tentativas de acesso sem sucesso, ou sessões TCP semi-abertas são exemplos de *features* que podem ser utilizadas no processo de auditoria.

- Classificação - Modelos utilizados para distinguir uma atividade normal de uma intrusão, classificando-a de acordo com regras pré-definidas.

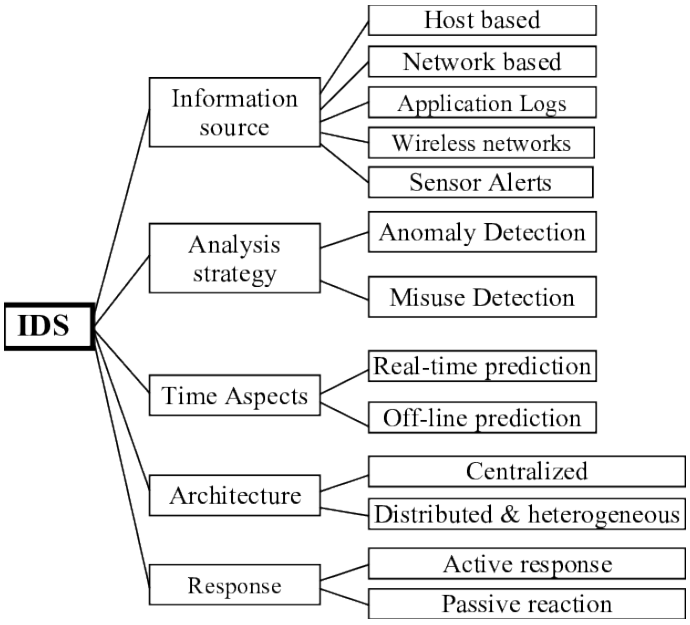


Figura 2.1: Taxonomia de sistemas detectores de intrusão proposta em [20].

- Resposta - Os eventos de intrusão são categorizados de acordo com sua classificação e severidade e externados sob forma de alarmes e/ou disparo de contra-medidas de proteção.
- Histórico - Gravação de todo processo de detecção e geração de alarmes, permitindo uma pós-auditoria de alto nível.

A Figura 2.2 mostra graficamente esta arquitetura.

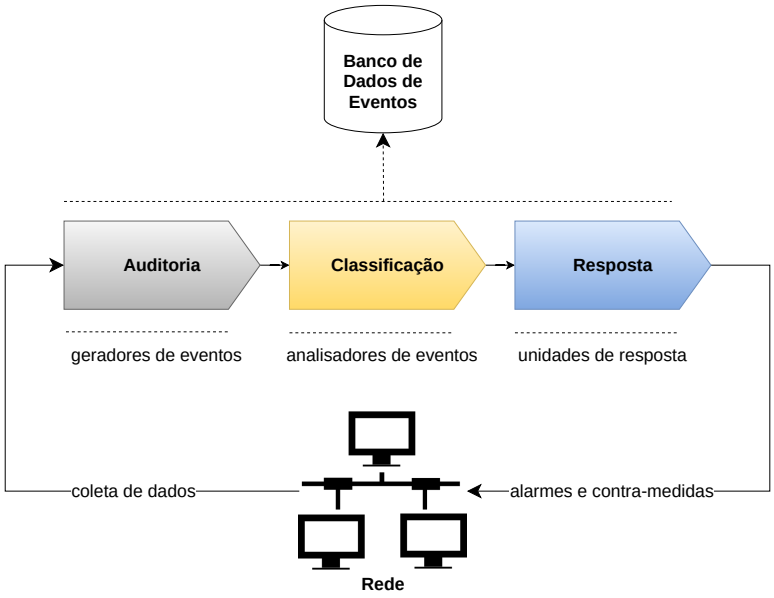


Figura 2.2: Arquitetura básica de um IDS descrita em módulos funcionais [20].

Com relação à fonte de dados de auditoria, existe o IDS que monitora os dados de um único hospedeiro (HIDS - *Host-based Intrusion Detection System*), tais como

tentativas interrupções, pedido de dados, tentativas de conexões, etc. Existe também o IDS que monitora o tráfego de uma rede inteira, junto com outros elementos. Estes IDS são conhecidos como NIDS(*Network-based Intrusion Detection System*). A Figura 2.3 mostra como ficam os perímetros de proteção no caso dos HIDSs e NIDSs.

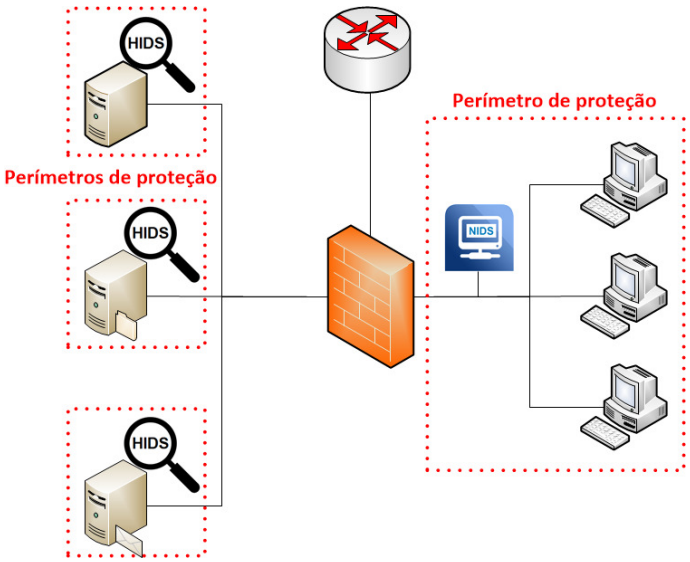


Figura 2.3: Diagrama de rede com 3 HIDSs e 1 NIDS com a representação de seus respectivos perímetros de proteção.

Com relação à metodologia de detecção, o IDS pode ser baseado em assinatura (*signature-based*), quando os dados monitorados são comparados à padrões de intrusões previamente configurados para detecção. A detecção também pode ser feita por detecção de anomalias (*anomaly-based*). Neste caso, um padrão de comportamento é definido pelo IDS, que alarma sempre que houver uma variação considerada anormal em relação a este padrão. A definição do padrão pode ser obtida de diversas formas, dependendo da tecnologia empregada. Uma das abordagens mais modernas para a definição de um padrão é baseada em aprendizado de máquina, como a proposta apresentada em [21]. Seja qual for a fonte de dados, tecnologia, metodologia de detecção, arquitetura, todos os IDSs enfrentam o mesmo desafio com relação desempenho de detecção: aumentar as taxas de alarmes verdadeiro-positivos e reduzir as taxas de falso-positivos.

2.1.1 Sistema Distribuído de Detecção de Intrusões

Os sistemas distribuídos de detecção de intrusão (DIDSs) surgiram como uma promessa para aumentar o desempenho de detecção e resolver os problemas inerentes à arquitetura monolítica dos IDSs. O conceito de detecção distribuída não é novo e a maioria das propostas são baseadas no sistema de imunização do corpo humano. A

ideia básica consiste na criação de um ambiente cooperativo, onde IDSs autônomos distribuídos (agentes) compartilham informações uns com os outros para aumentar o desempenho de detecção de forma geral. A Figura 2.4 ajuda a entender uma destas arquiteturas, onde 5 IDSs diferentes enviam mensagens de ataques para um ponto central, encarregado de combinar as mensagens e subsidiar medidas protetivas e de segurança. As setas tracejadas, ligando os IDSs entre si, indicam que os IDSs também podem cooperar enviando mensagens entre si próprios.

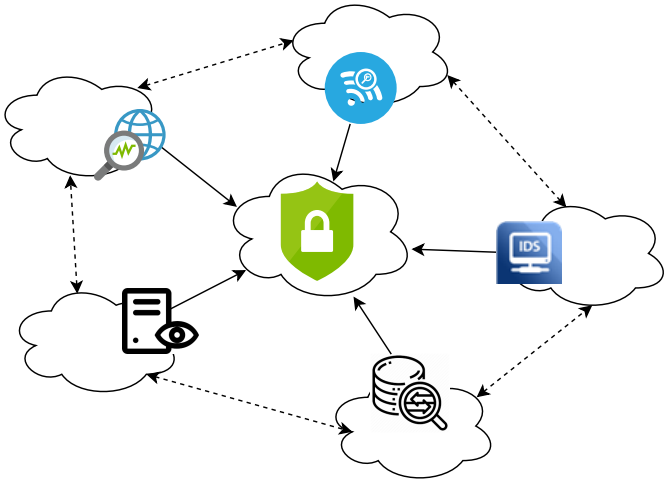


Figura 2.4: Arquitetura básica de um DIDS, onde os 5 IDSs distintos enviam mensagens entre si próprios e para um centro de processamento centralizado.

Diferentemente da arquitetura tradicional, na arquitetura distribuída a ausência ou a falha de um dos agentes também não chega a comprometer o funcionamento do sistema como um todo. A heterogeneidade dos IDSs agentes também é um fator importante a ser considerado na detecção de ataques de dia zero, uma vez que, o que seria um novo ataque (dia zero) para um determinado agente pode não sê-lo para um outro IDS geograficamente distante [22]. Apesar do grande número de propostas, ainda não se pode dizer que os sistemas distribuídos de detecção são uma realidade prática. Isto se deve basicamente aos desafios relacionados com a infraestrutura de transporte de informações entre os agentes e à complexidade para criar um ambiente cooperativo entre os agentes autônomos.

Os sistemas de detecção de intrusões previstos para a quinta geração das redes sem fio 5G já incorporam conceitos de combinação de dados para reagir rápida e efetivamente, de acordo com as características do seu ecossistema [23–25]. Um dos pilares da tecnologia 5G, o conceito de redes auto-organizáveis (*Self-Organizing Networks* - SON), inclui uma plataforma de detecção inteligente e distribuída, que consegue aprender e reagir de acordo com o ambiente e com as características da intrusão.

2.2 BGP FlowSpec

A motivação principal da especificação do protocolo FlowSpec é possibilitar contramedidas automáticas para mitigação de ataques de negação de serviços, o mais perto possível de sua origem. O FlowSpec está definido na RFC (*Request for Change*) 5575 [26] e o seu funcionamento é baseado nas possibilidades de interfuncionamento oferecidas pelo protocolo MP-BGP (RFC 4670) [27].

O FlowSpec permite a disseminação de fluxos, considerados ofensores, através do campo NLRI, para toda vizinhança BGP. Os vizinhos que receberem estas atualizações BGP poderão aplicar regras de filtragem, baseadas nos atributos do fluxo trazidas no campo NLRI da mensagem FlowSpec.

O campo NLRI do FlowSpec oferece 12 opções para especificação do fluxo ofensor, o que permite um filtro consideravelmente mais preciso. Esta precisão é importante para evitar que os filtros acabem por efetivar um determinado ataque, bloqueando tráfego considerado lícito também. A Tabela 2.1 logo abaixo mostra estas opções.

Tabela 2.1: Atributos que podem ser utilizados na especificação do fluxo ofensor no FlowSpec.

Tipo	**Descrição**	**Tipo**	**Descrição**
1	*Destination Prefix*	7	*ICMP type*
2	*Source Prefix*	8	*ICMP code*
3	*IP Protocol*	9	*TCP flags*
4	*Port*	10	*Packet length*
5	*Destination Port*	11	*DSCP*
6	*Source Port*	12	*Fragment*

Apesar de não ser uma RFC nova, a 5575 ainda não é utilizada extensivamente pelos provedores de acesso. Isto se deve principalmente ao risco de acabar prejudicando um tráfego lícito, a partir do recebimento de uma mensagem de FlowSpec, enviada por um AS distante e desconhecido, contendo informações incorretas ou mesmo maliciosas. Assim, por causa destes problemas de confiabilidade, o FlowSpec é mais utilizado hoje em dia para complementar sistemas de mitigação de ataques proprietários e restritos a um determinado AS. A Figura 2.5 mostra o funcionamento do FlowSpec num cenário envolvendo um cliente e seu provedor de acesso.

2.3 Inferência Bayesiana

A inferência Bayesiana pode ser definida como um processo de ajuste de um modelo probabilístico de acordo com um conjunto de dados, resultando numa distribuição

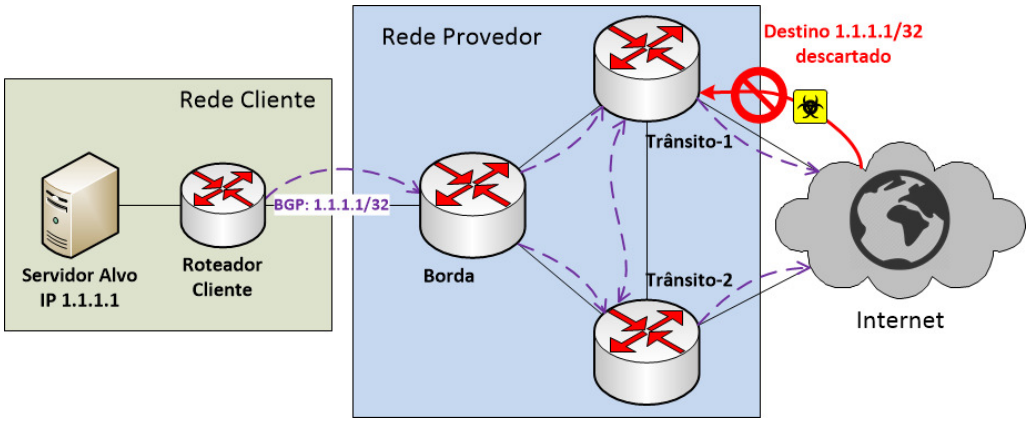


Figura 2.5: Princípio de funcionamento do protocolo FlowSpec.

de probabilidade condicionada a parâmetros, que pode ser usada para prever (predizer) dados que ainda não foram observados [28]. Independentemente das discussões filosóficas envolvendo as teorias frequentista e Bayesiana, um dos maiores benefícios desta última abordagem está flexibilidade e generalidade para a quantificação direta da incerteza. De fato, a inferência Bayesiana interpreta a probabilidade como uma medida de incerteza sobre uma hipótese. Esta probabilidade, que pode ser definida subjetivamente, converge através de um processo de atualização pelos dados. A inferência Bayesiana é baseada no teorema de Bayes.

O teorema de Bayes descreve a probabilidade de um determinado evento baseado no conhecimento prévio relacionado com este evento e nas evidências provenientes dos dados coletados. Para dois eventos quaisquer A e B, a probabilidade condicional de A dado B é denotada como $P[A|B]$ e pode ser definida como:

$$P[A|B] = \frac{P[B|A].P[A]}{P[B]} \tag{2.1}$$

Na Equação 2.1, a probabilidade condicional $P[A|B]$ é chamada de probabilidade a posteriori. Os termos $P[B|A]$ e $P[A]$ são conhecidos como verossimilhança e probabilidade à priori respectivamente. O termo $P[B]$ no denominador da Equação 2.1 é chamado de termo normalizador, uma vez que pode ser decomposto em $P[B] = P[B|A].P[A] + P[B|\overline{A}].P[\overline{A}]$.

Diferentemente da clássica teoria frequentista, onde a probabilidade é interpretada através longas repetições de experimentos, a teoria de Bayes entende probabilidade como níveis de crença. Enquanto na inferência frequentista a incerteza provém apenas da aleatoriedade dos resultados, na inferência Bayesiana a falta de conhecimento sobre o evento também pode ser modelada.

Na inferência Bayesiana, a verossimilhança é função dos estados da variável de interesse, correspondentes aos dados fixos observados no experimento. A probabilidade à priori representa o conhecimento prévio sobre a variável de interesse,

antes das observações. A probabilidade a posteriori, por sua vez, é a probabilidade de ocorrência de um determinado evento, considerando as evidências observadas e algum conhecimento prévio antes das mesmas. A Figura 2.6 logo abaixo mostra graficamente esta relação.

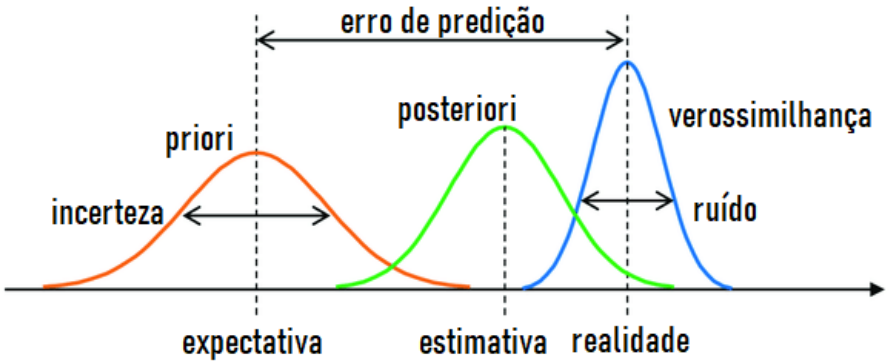


Figura 2.6: Termos da inferência Bayesiana de acordo com [29].

2.3.1 Probabilidade à Priori

A necessidade de se incorporar um conhecimento prévio na equação de Bayes sobre o parâmetro de interesse é de longe o assunto mais controverso na comparação entre a teorias frequentista e Bayesiana. Isto se deve ao fato de que, uma vez tendo evidências de uma determinada variável de interesse, a única informação restante para se chegar na condição de decisão é a probabilidade à priori. Portanto, a importância da escolha da probabilidade à priori aumenta quando o tamanho da amostra de dados é pequeno. As probabilidades à priori podem ser classificadas com relação à sua origem e com relação ao nível de crença que carrega. Com relação à origem, as probabilidades à priori se dividem em:

- Subjetiva: quando não depende da amostra (verossimilhança). Geralmente provém da crença pessoal de um indivíduo ou de um grupo antes da coleta de dados, ou de uma outra amostragem relacionada. Por exemplo: consultando um grupo de especialistas sobre aquele assunto específico ou através da matriz de confusão de um *dataset* prévio.

- Objetiva: quando depende da amostra (verossimilhança). Por exemplo: funções conjugadas.

Com relação ao nível de informação que carrega, as probabilidades à priori podem ser classificadas como:

- Informativa: quando assumem um grau de conhecimento prévio acerca do parâmetro. Por exemplo: $\theta \sim N(0, 100)$ é menos informativa que $\theta \sim N(0, 1)$.

- Não-informativa: quando assumem desconhecimento total em relação ao parâmetro. Por exemplo: proporcional a uma constante $p(\theta) = U(0,1)(\theta)$.
- Semi-informativa: quando decorrem da combinação do conhecimento prévio com outras informações obtidas da própria evidência.

A escolha pela utilização de probabilidades a priori subjetivas ou muito informativas deve levar em conta uma série de aspectos, o que a torna difícil em algumas situações:

- Disponibilidade de dados históricos confiáveis acerca das hipóteses em teste.
- Disponibilidade de opinião qualificada para servir como base.
- Modelos complicados multi-dimensionais.

Probabilidades à priori muito informativas podem resultar em valores de probabilidade à posteriori incorretos ou sem sentido. Uma opção possível é utilizar probabilidades vagas (proporcionais a uma constante), mas mesmo estas podem ser demasiadamente informativas ou gerar probabilidades à posteriori impróprias (não convergente quando integrada nos valores possíveis).

A escolha por utilizar probabilidades à priori objetivas não-informativas é adequada quando se deseja enfatizar os resultados dos testes de hipótese, contando com um conjunto amostral pequeno. Nesta condição, a escolha ainda deve considerar o nível de incerteza que se deseja expressar em relação aos dados coletados, e a tratabilidade matemática do cálculo da posteriori. Em relação à representação da incerteza, a definição de probabilidades à priori de referência tem sido uma tendência. As probabilidades de referência possuem propriedades consensualmente desejadas e passíveis de serem aplicadas em situações onde haja pouca informação disponível a priori, são elas:

- A capacidade de gerar informações estatísticas com o mínimo de informação subjetiva à priori.
- A produção de análise neutra, baseada num padrão convencionado.
- A característica de permitir que a evidência decorrente do experimento seja mais forte que a evidência decorrente da priori.

Uma das distribuições de referência mais utilizadas como probabilidade à priori objetiva é a distribuição Beta. A distribuição Beta é uma função contínua, definida no intervalo $[0,1]$. Geralmente é utilizada para modelar o comportamento de variáveis aleatórias limitadas à intervalos finitos, como uma função de probabilidade. Por

sua flexibilidade em adaptar seu nível de informação de acordo com seus parâmetros e por ser conjugada de distribuições como a distribuição de Bernoulli, Binomial e a Geométrica, a função Beta encontra muita utilização para resolver problemas de inferência Bayesiana.

2.4 Função Distribuição Beta

A função de distribuição Beta é uma família contínua que pode assumir inúmeras formas em função da combinação de seus parâmetros α e β. Esta flexibilidade dá à função de distribuição Beta uma larga gama de aplicações, sendo a principal delas para representar probabilidades à priori. Isto é, representando todos os possíveis valores de probabilidade que uma determinada variável aleatória (VA) Θ pode assumir entre $[0, 1]$. Por exemplo, imagine que se deseja calcular o quão tendenciosa é uma moeda para o resultado "cara" $\Pr(cara) = p$. Uma forma de se chegar neste resultado é arremessar a moeda $(n + m)$ vezes, contabilizar n como o número de resultados "cara" obtidos e calcular $p = \frac{n}{n+m}$. Entretanto, se $n + m$ for um número pequeno, é provável que a estimativa de p fique comprometida, não conseguindo representar adequadamente a incerteza em relação a p. Neste caso, a probabilidade p pode ser representada pela variável aleatória $\Theta \sim Beta(\alpha, \beta)$.

A função densidade de probabilidade (PDF) para $\Theta \sim Beta(\alpha, \beta)$ pode ser escrita como:

$$f_\Theta(\theta) = \begin{cases} \frac{1}{B(\alpha,\beta)}\theta^{(\alpha-1)}(1-\theta)^{(\beta-1)}; & \text{se } 0 < \theta < 1 \\ 0; & \text{em qualquer outra condição} \end{cases} \tag{2.2}$$

onde $B(\alpha, \beta) = \int_0^1 \theta^{(\alpha-1)}(1-\theta)^{(\beta-1)}d\theta$.

A utilização da função de distribuição Beta na inferência Bayesiana é especialmente conveniente conjugando com a verossimilhança, quando esta for do tipo Bernoulli, geométrico ou binomial. Neste caso, o resultado da operação de multiplicação mantém o formato original, mudando apenas os seus parâmetros (hiper-parâmetros).

Neste trabalho, a função Beta é utilizada para modelar a probabilidade à priori das equações Bayesianas que calculam as métricas de desempenho de detecção da plataforma DIDS. Nesta modelagem, o valor da probabilidade de detecção obtido empiricamente através dos valores das matrizes de confusão utilizadas são modulados com a probabilidade à priori correspondente à cada hipótese de detecção considerada.

Os parâmetros positivos α e β são expoentes da variável aleatória e controlam o formato da distribuição. Enquanto o parâmetro $\alpha - 1$ corresponde ao número de sucessos no experimento, o parâmetro $\beta - 1$ está relacionado com o número de fracassos. Quanto maiores forem os valores de α e β mais informativa será a

probabilidade, maior será a crença num determinado valor da verossimilhança e menor a variância da probabilidade à posteriori modulada. Se α e β forem iguais, porém pequenos, o valor resultante da função Beta exercerá apenas a função de aumentar a variância do resultado da probabilidade à posteriori. A Figura 2.7 mostra o comportamento em forma de "sino" da função Beta para diferentes valores de $\alpha = \beta$.

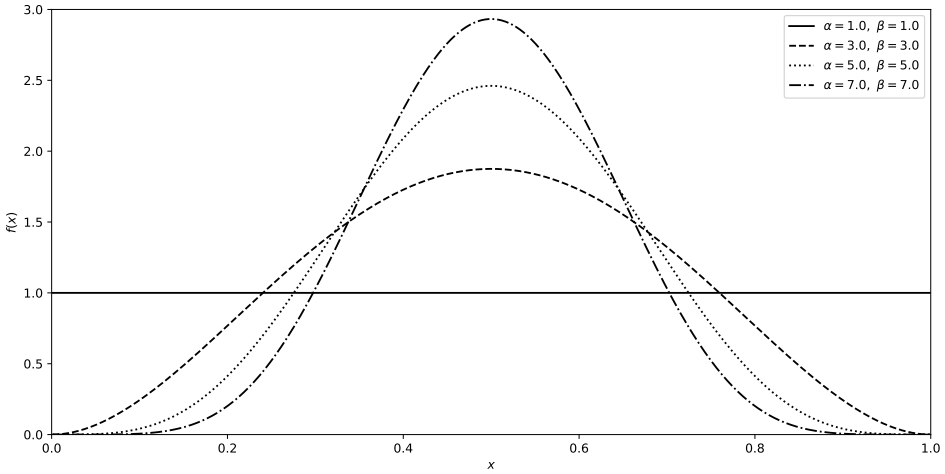


Figura 2.7: Função de densidade de probabilidade Beta com $\alpha = \beta$ para diferentes valores.

2.5 Fusão de Dados

O conceito de fusão de dados originou-se em 1980 como um projeto da força aérea americana para monitoração e reconhecimento de alvos. A fusão de dados pode ser definida como um processo automático ou semi-automático que associa, correlaciona e combina dados e informações provenientes de uma única ou de múltiplas fontes, com o objetivo de consolidar uma determinada informação e sua significância [30]. A fusão de dados é amplamente utilizada em várias áreas, tais como: redes de sensores, robótica, processamento de imagens, diagnósticos médicos, etc. Um dos principais desafios de qualquer sistema de fusão de dados está relacionado com a imperfeição dos dados a serem fundidos por causa de incerteza, imprecisão ou diferentes granularidades [31]. Fusão de dados e fusão de informações são termos muitas vezes utilizados como sinônimos. Entretanto, em algumas situações, o termo fusão de dados é mais utilizado quanto se deseja fundir dados brutos (*row data*), enquanto a fusão de informações é utilizado para fundir níveis semânticos mais altos.

Existem diversos métodos para fazer fusão de dados. Estes métodos se diferenciam basicamente pela representação de aspectos relacionados com a imperfeição

dos dados de entrada e com a propagação de crença. Independente do método utilizado, os principais objetivos de qualquer processo de fusão de dados são a redução da probabilidade de erro e o aumento do nível de crença das informações fundidas, através da utilização de dados de múltiplas fontes [32].

Ainda de acordo com o trabalho de Castanedo [32], os métodos de fusão podem ser classificados de acordo com:

- Relação entre os dados de entrada.
  - Complementares - quando os dados de entrada se complementam entre si.
  - Redundantes - quando os dados de entrada provêm a mesma informação.
  - Cooperativos - quando os dados de entrada são combinados para formar informações mais complexas.
- Tipo de dados de entrada ou saída. Avalia o formato dos dados de entrada e saída. Por exemplo, se os dados de entrada forem brutos e os dados de saída permanecem brutos, porém com uma maior acurácia (*Data in/Data out* - DAI/DAO).
- Nível de abstração. Refere-se ao significado abstrato dos dados de entrada em relação com o produto do processo de fusão. Por exemplo: uma fusão de alto nível, também conhecida como fusão de decisão, recebe como entrada símbolos e os combina para a tomada de decisões com maior acurácia. O método Bayesiano é um exemplo prático deste tipo de fusão.
- Nível de fusão. Considera o nível de processamento dos dados de entrada que serão fundidos.
- Arquitetura.
  - Centralizada. Todo o processo de fusão é executado num só elemento central.
  - Descentralizada. Parte do processamento é executado localmente e não existe um único ponto de falha.
  - Distribuída. As fusões são processadas em cada um dos elementos distribuídos, antes de serem enviadas para a fusão.
  - Hierárquica. Combina as arquiteturas descentralizada e distribuída para executar fusões em diferentes níveis.

Com relação à metodologia utilizada para fazer a fusão dos dados, há uma diversidade de abordagens. Entretanto, as principais são:

- Probabilística. Se baseia nas funções de distribuição de probabilidades para processar as incertezas dos dados de entrada. O método de fusão probabilística mais conhecido é o Bayesiano, que assume o conhecimento prévio de probabilidades à priori para calcular as probabilidades condicionais à posteriori. Apesar de largamente utilizado, a dependência de se conhecer as probabilidades à priori é uma das principais desvantagem deste método. Métodos baseados em frequência relativa (treinamento/aprendizado) ou paramétricos, baseados em máxima entropia [33] ou informação mútua [34], podem ser utilizados para estimar estas probabilidades à priori.

- Crença evidencial. A teoria das funções de crença também é conhecida como Teoria da Evidência de Dempster-Shafer (DST) [35]. Neste caso, ao invés de atribuir uma probabilidade à priori para uma determinada hipótese, a DST atribui uma massa de crença, considerando que há uma ou mais evidências suportando-a. A DST permite uma explícita representação da falta de conhecimento, da ambiguidade ou da alienação[1] das fontes de evidência para produzir informações com limites inferiores (Bel) e superiores (Pls) de probabilidades. Apesar da elegância matemática e da simplicidade, a regra de fusão de dados de Dempster apresenta algumas desvantagens, sendo as principais relacionadas com o trabalho computacional para processar quadros de discernimento muito grandes ($2^{|\Omega|}$) e quando as evidências são muito conflitantes entre si.

- Nebulosa. Também conhecida como método de fusão possibilística, se baseia em lógica de conjuntos nebulosos (*fuzzy*) para representar dados incompletos ou vagos. Apesar de ser bastante prática na representação contínua da incerteza implícita do conhecimento humano, pode gerar inconsistências na composição de diferentes granularidades lógicas [36].

Existem ainda as abordagens híbridas, combinando algumas das classificações acima para determinação do nível de crença a partir da fusão de dados provenientes de múltiplas fontes. Uma destas propostas, conhecida como redes de confiança subjetiva [37], se baseia em lógica subjetiva e redes Bayesianas para expressar a crença numa determinada hipótese, considerando o grau de credibilidade subjetiva de sua fonte. Neste caso, a soma dos níveis de crença de cada uma das hipóteses do hiper-domínio (quadro de discernimento) deve ser menor ou igual a 1 e o complemento desta soma é entendido como a massa de incerteza, que reflete o grau de credibilidade na opinião. As opiniões também contêm taxas básicas de probabilidade, que expressam o conhecimento prévio (probabilidade à priori) sobre uma classe específica da variável aleatória, cuja distribuição é buscada.

[1] Alienação no sentido de simplesmente não evidenciar nada

Uma outra proposta híbrida para fusão de dados é apresentada em [38]. Nesta proposta, Yager combina as metodologias possibilística e probabilística, aproveitando os principais pontos fortes de ambas as abordagens individualmente. Enquanto a abordagem possibilística expressa o conhecimento específico sobre um determinado objeto em particular, a abordagem probabilística representa a informação consolidada de uma classe de objetos. Por exemplo, se o interesse é descobrir a altura de uma pessoa, é mais fácil utilizar o modelo possibilístico, uma vez que já existe o pré-conhecimento que esta pessoa é alta.

A Teoria da Evidência de Dempster-Shafer [35] tem sido utilizada em processos de fusão de dados incertos e alienados com o objetivo de suportar a tomada de decisões, entre outros. Uma das principais vantagens da Teoria da Evidência está na sua capacidade de modelar a "falta de informação" sobre algum evento. Por exemplo: de acordo com a clássica Teoria Probabilística, a máxima incerteza de uma variável aleatória de Bernoulli ocorre quando a probabilidade desta variável assumir um valor é igual a $0,5$. Entretanto, de acordo com o conceito de níveis de crença da Teoria da Evidência, atribuir a mesma probabilidade a ambas as hipóteses por si só já representaria uma informação importante na aplicação de fusão de dados. Um outro ponto positivo na Teoria da Evidência é a sua capacidade de refletir melhor o acúmulo e a combinação de evidências, independente da ordem na qual estas evidências surgem no sistema.

Por outro lado, as duas principais desvantagens do modelo matemático de fusão de dados de Dempster-Shafer são:

- Quadro de discernimento: a combinação de possibilidades do conjunto potência ($2^{|\Omega|}$) pode produzir quadros de discernimento muito grandes, dificultando os cálculos.

- Evidências conflitantes [39]: a regra de combinação de Dempster produz resultados incorretos em caso de evidências antagônicos ou conflitantes.

Os estudos sobre fusão de dados encontram-se reaquecidos ultimamente com o lançamento da tecnologia 5G como plataforma básica para alavancar o conceito de Internet das Coisas (IoT). Aplicações críticas como Cidades Inteligentes [40] e Sistemas de Transportes Inteligentes (ITS) [41] requerem a análise de grandes massas de dados provenientes de fontes heterogêneas, distribuídas e imprecisas (sensores). Neste novo contexto, a fusão de dados pode participar em etapas de filtragem e consolidação, produzindo informações refinadas e de alto valor agregado para a tomada de decisões.

2.6 Anúncios de Atualização BGP

De acordo com a RFC 4271 [42], a mensagem de atualização do BGP (*update message*) é utilizada para transferir informações de roteamento entre vizinhos BGP. As informações contidas na mensagem de atualização são usadas para construir o grafo que descreve as relações entre os sistemas autônomos da Internet (AS), adicionando rotas viáveis ou retirando rotas inviáveis. Os roteadores conectados à rede BGP enviam mensagens de atualização sempre a sua tabela *Adj-RIB-Out* muda. Uma atualização anunciada por um determinado roteador conectado à rede BGP é primeiramente recebida e registrada na tabela *Adj-RIB-In* do seu vizinho. Depois de passar por filtros de entrada de prefixos e seleção da melhor rota, as rotas são então registradas na tabela *Loc-RIB* deste segundo roteador, onde podem ser utilizadas nas operações locais de roteamento. Dando continuidade ao processo, se passarem pelos filtros de saída, as rotas são registradas na tabela *Adj-RIB-Out*, para serem novamente anunciadas para a toda a rede BGP. A Figura abaixo mostra graficamente como este processo se desenrola na rede BGP.

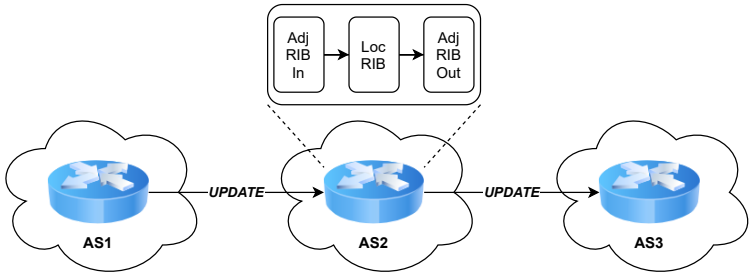


Figura 2.8: O Roteador do AS1 faz um anúncio, que é recebido na tabela *Adj-RIB-In* e gravado na tabela *Loc-RIB* do roteador do AS2. As rotas que passarem pelos filtros de saída de prefixos do roteador do AS2 são então gravadas na tabela *Adj-RIB-out* para serem novamente anunciadas pela rede BGP.

A mensagem de atualização BGP é dividida em vários campos que indicam o seu conteúdo e a forma como estes dados devem ser processados na rede. Um destes campos é o campo “atributos de caminho" que por sua vez é dividido em outros campos, os quais são obrigatórios em todas as mensagens de atualização.

1. *ORIGIN* - Campo mandatório que define a origem da informação do caminho.

    - O valor 0 neste campo indica que a informação que está sendo divulgada (NLRI) provém do próprio AS pelo IGP.
    - O valor igual a 1 indica que o anúncio está repassando a informação (NLRI) aprendido de um outro AS, via EGP.
    - O valor igual a 2 indica que a informação do campo NLRI no anúncio foi aprendida de uma outra forma diferente das duas anteriores.

2. *AS_PATH* - Campo mandatório composto pela sequência de ASs, desde a origem do anúncio até o destino.

    - *AS_PATH_TYPE* - o vetor de ASs pode vir desordenado (*AS_SET*), em caso de agregação de prefixos, ou ordenado (*AS_SEQUENCE*).
    - *AS_PATH_LENGTH* - Número total de ASs que compõem o caminho do anúncio, desde a sua origem até o AS de destino.
    - *PATH_SEGMENT_VALUE* - Contém a lista de ASs envolvidos no anúncio, desde o AS de origem do *AS_PATH*, até o último AS que processou o anúncio.

3. *NEXT_HOP* - Define o endereço IP do roteador que deve ser utilizado no próximo salto para os prefixos de destino anunciados no campo NLRI.

2.7 Aprendizado de Máquina

O conceito de aprendizado de máquina consiste em obter uma representação matemática que modele o comportamento de uma função através de um processo chamado de treinamento. No treinamento, a partir de um banco de amostras conhecido (*dataset*) com múltiplos atributos (*features*), o modelo tem seus parâmetros ajustados de forma a conseguir prever um conjunto inédito de amostras. A correta seleção de atributos é então de vital importância para o sucesso do modelo de aprendizado e corresponde a uma extensa área de pesquisa no campo de aprendizado de máquina [43]. Outro ponto importante é a capacidade do modelo matemático em "compreender" um problema complexo, capacidade esta que deve vir acompanhada de rápida resposta e baixa sensibilidade a variações.

Os algoritmos de aprendizado de máquina podem ser divididos em três classes:

- Supervisionado - O banco de dados de treinamento (*dataset*) é rotulado, ou seja, existem exemplos do mapeamento entre entrada e saída. A presença dos rótulos possibilita que os algoritmos ajustem seus parâmetros para reproduzirem as mesmas saídas caso entradas semelhantes sejam apresentadas.

- Não-supervisionado - O banco de dados não tem rótulos, não existindo um mapeamento entre entradas e saídas. Nesse cenário, os algoritmos buscam relações e características presentes no conjunto de dados que possam ser exploradas para classificar internamente os elementos.

- Reforço - Os algoritmos de aprendizado por reforço se baseiam num modelo de punições e recompensas à medida que o modelo interage com o ambiente onde

está inserido. Assim, ao invés de existir um mapeamento direto entre entradas e saídas, os resultados são obtidos a partir da realimentação (*feedback loop*) entre o sistema de aprendizado e o ambiente.

Apesar resolverem um grande número de problemas, existem situações que podem ser resolvidas diretamente, sem o uso das técnicas de aprendizado de máquina. Por exemplo: quando for possível determinar uma função direta determinística ou estatística, entre os dados de entrada e a resposta do sistema. O uso das técnicas de aprendizado de máquina são especialmente úteis quando:

- A relação entre os dados de entrada e a saída do sistema depende de um grande número de fatores que se sobrepõem ou que precisam ser sintonizados para retratar a função de transferência do sistema.
- As soluções diretas não escalam o suficiente para processar grandes volumes de dados.

Dentro das situações relacionadas acima, os métodos de aprendizado de máquina podem ser utilizados para solucionar vários tipos de problemas, onde os mais comuns são [44]:

- Classificação - Nesta função, o algoritmo deve classificar o dado de entrada numa das k categorias pré-conhecidas, produzindo uma função $f : \mathbb{R}n \rightarrow 1, 2, ..., k-1, k$. Reconhecimento de objetos é um exemplo deste tipo de problema, onde a entrada é uma imagem e a saída pode ser um código, identificando o objeto na imagem.
- Completamento - Em algumas situações os dados de entrada simplesmente podem faltar. Nesta tipo de problema, o algoritmo deve aprender um conjunto de funções, onde cada uma delas classifica o dado faltante, dentro de um conjunto pré-definido. Um exemplo deste tipo de problema é na área de sensores, em que os dados de um determinado sensor não estão disponíveis durante um período de tempo.
- Regressão - A tarefa de predizer um valor numérico n, dada alguma entrada $f : \mathbb{R}n \rightarrow \mathbb{R}$. Por exemplo: determinar o valor de crédito de um indivíduo, com base em dados como: sua idade, se tem casa própria, número de filhos, etc.
- Detecção de Anomalias - Este tipo de algoritmo processa uma sequência de eventos ou objetos e marca aqueles que são considerados anômalos, em relação a uma base de normalidade treinada. A detecção de ataques de negação de serviços é um bom exemplo desta aplicação.

No caso dos anúncios BGP-FlowSpec disseminados pela Internet, o desafio é extrair um conjunto estratégico de atributos das mensagens de atualização (*update*) para montar um *dataset*, cujo rótulo é relacionado à existência ou não de uma intrusão real sinalizada pela mensagem de atualização. O *dataset* rotulado será então utilizado para treinar um algoritmo de aprendizado de máquina, capaz de generalizar e inferir sobre o nível de confiança dos novos anúncios.

2.8 Planos de Controle 4G

O plano de controle do sistema LTE, comumente conhecido como EPC, é uma arquitetura linear, formada por um núcleo multi-acesso baseado em IP, para onde converge o processamento de chamadas de voz e de dados. Isto é, diferentemente do 3G, onde as chamadas de voz ainda são completadas usando a técnica de comutação de circuitos (*circuit switching*), no 4G tanto voz quanto dados são processados utilizando a técnica de comutação por pacotes (*packet switching*). Entre outras várias vantagens em relação ao 3G, o núcleo de processamento baseado em pacotes do 4G permite que as chamadas de voz e de dados ocorram simultaneamente, sem perda de qualidade.

A Figura 2.9 abaixo mostra a arquitetura básica simplificada do sistema 4G, incluindo o EPC e suas interfaces de comunicação com a rede de acesso (RAN).

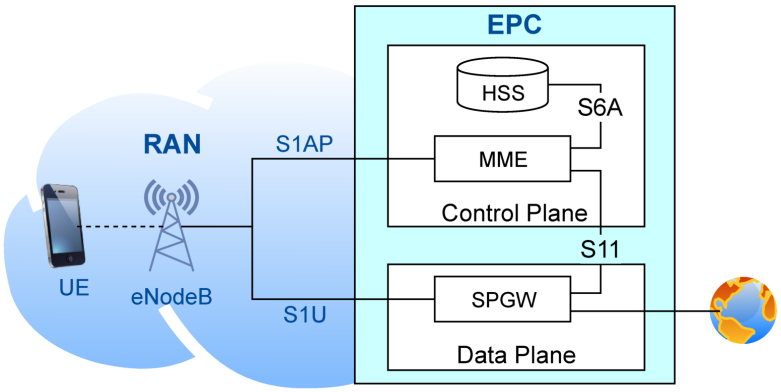


Figura 2.9: Arquitetura 4G simplificada proposta pelo 3GPP, incluindo núcleo (EPC) e rede de acesso (RAN), retirada de [45].

O EPC é composto basicamente por três blocos funcionais distintos: entidade de gerenciamento de mobilidade (*mobility management entity* - MME), base de dados local de usuários (*home subscriber sub-system* - HSS) e servidor gateway de pacotes (*serving packet data network gateway* - SPGW). Estes blocos interagem uns com os outros através de interfaces padronizadas para processar as transações de sinalização do sistema. O MME é responsável por controlar o processo de sinalização entre a rede de acesso (RAN) e o EPC, envolvendo mobilidade e segurança. O HSS funciona

como o banco de dados do sistema, armazenando as informações relacionadas com os usuários do sistema móvel. Por fim, o SPGW lida com a sinalização relacionada com a reserva de recursos (plano de controle) e também encaminha os dados da RAN para as redes externas.

Apesar da significativa evolução em relação ao 3G, a rede 4G ainda herdou alguns legados, misturando o plano de controle com o plano de dados. Esta herança, embora se justifique pela necessidade de adaptação e de manter a compatibilidade com o 3G e 2G, também dificulta os processos de virtualização do plano de controle. Além disto, sua concepção arquitetônica baseada em pacotes com foco na integração com a Internet trouxe novas ameaças de segurança, consubstanciadas com o surgimento de novas vulnerabilidades e oferecendo oportunidades para o desenvolvimento de novos tipos de ataques [46].

2.9 Planos de Controle 5G

Embora o núcleo da plataforma 5G tenha herdado as mesmas macro-funções do seu predecessor 4G, sua arquitetura baseada em serviços (*service-based architecture* - SBA) é organizada em 10 blocos funcionais diferentes. O principal objetivo desta modificação é separar totalmente os planos de dados e de controle, viabilizando a virtualização em nuvem e novas funcionalidades como fatiamento de rede (*network slicing*) e o ambiente de múltiplos locatários (*multi-tenant*). A Figura 2.10 mostra a arquitetura baseada em serviços do núcleo do 5G, com seus blocos funcionais interligados através de interfaces padronizadas (controle e dados) e a rede de acesso (RAN).

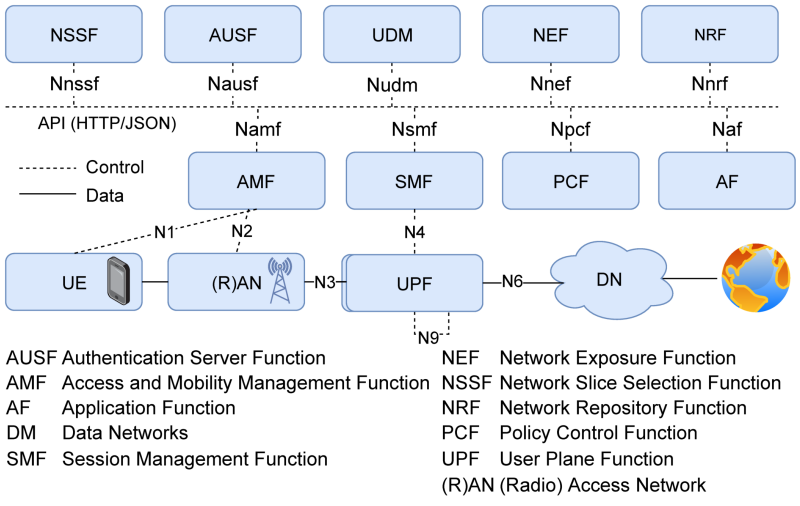


Figura 2.10: Arquitetura baseada em serviços (SBA) do núcleo 5G, incluindo as interfaces padronizadas e as conexões com a rede de acesso (RAN), definida em [47].

Diferentemente do plano de controle 4G, que também já permite a virtualização

de algumas funções, as funções distribuídas no plano de controle 5G já nasceram arquitetadas para funcionarem na nuvem. Fazendo uma comparação básica a título de exemplo, as funções *gateway* de sinalização e gerenciamento de mobilidade, que no 4G eram executadas pelo MME, no 5G são executadas por dois novos blocos funcionais diferentes e nativos na nuvem:

- AMF como função de acesso e gerenciamento de mobilidade (*Access & Mobility Management Function* - AMF).
- SMF como função de controle e políticas de seção *Session Management Function* - SMF)

Entretanto, a constante e intensa troca de mensagens entre os blocos funcionais na nuvem cria um cenário crítico, onde qualquer congestionamento, causado, por exemplo, por um ataque de DoS, poderia deteriorar ou mesmo interromper os serviços de sinalização.

2.10 Ataques de Negação de Serviços

Os ataques de negação de serviços (*Denial of Services* - DoS) ameaçam a disponibilidade dos serviços se caracterizam pela sua furtividade. Ou seja, diferentemente de outros tipos de ataques, o objetivo primário dos ataques de DoS não é roubar alguma informação, mas sim deteriorar ou mesmo interromper os serviços de um determinado alvo. O ataque de DoS se caracteriza principalmente pela presença de um mestre (atacante) que controla remotamente escravos (*bots*), previamente recrutados para atacar um determinado alvo. O ataque de DoS tem por objetivo impedir a utilização de recursos de um determinado alvo por usuários legítimos, exaurindo seus recursos através de demandas sinteticamente intensificadas. A intensificação é possível explorando-se características de amplificação dos sistemas, como, por exemplo, consultas DNS ou NTP.

Uma variante ainda mais ameaçadora de ataque de negação de serviços é a sua versão distribuída (DDoS). Os ataques de DDoS podem ser lançados de *bots* (às vezes centenas de milhares, como em [48]) contra um mesmo alvo, aumentando muito seu poder de causar danos. Quando um ataque de DDoS é lançado a partir de fontes internas, ou com prerrogativas de usuários internos do sistema, os efeitos do ataque de DDoS se tornam ainda mais devastadores, uma vez tendo acesso às áreas mais sensíveis do sistema.

Seguindo a mesma estratégia de disfarce, os ataques de DDoS de sinalização tem por objetivo comprometer ou interromper os serviços do plano de controle por exaustão. Após tendo recrutado um determinado número de *bots*, o atacante aproveita os

processos orientados a estado no plano de controle e o grande número de mensagens necessárias para executar as transações de sinalização, para amplificar o ataque. Um exemplo prático deste tipo de ataque é apresentado no trabalho em [49, 50], onde o agressor se aproveita as de mensagens desencadeadas nas transações de *attach request* e *handover request* [51] como vetores do ataque para inundar o plano de controle. Se muitos *smartphones*, estrategicamente distribuídos em múltiplas estações rádio-bases, são simultaneamente controlados para repetirem tais transações continuamente, o tráfego total de sinalização gerado pela *botnet*[2] pode sem dúvida exaurir o plano de controle como um todo.

2.11 Teoria de Filas

Segundo [52], qualquer sistema que recebe demandas de intervalos e tamanhos aleatórios, sobre um determinado recurso com capacidade limitada, pode ser chamado como "sistema de filas". Principalmente quando estas demandas têm que aguardar em fila dentro deste sistema, enquanto aguardam a disponibilidade do recurso. A Teoria de Filas é um ramo da matemática que estuda como se formam estas filas e como elas funcionam, considerando:

- o processo de chegada de usuários (demandas) no sistema,
- o processo de serviço dos mesmos,
- o número de servidores (recursos),
- a capacidade de armazenamento do sistema e
- o número total de usuários (demandas).

A Teoria de Filas é amplamente utilizada em diversas áreas de estudo, onde se deseja analisar o desempenho de sistemas que podem ser representados usando estruturas do tipo cliente/servidor. A Figura 2.11 abaixo mostra graficamente um sistema de filas típico, com um único servidor com capacidade média de serviço igual a μ usuários/segundo e que recebe usuários (demandas) a uma taxa média de λ usuários por segundo. O espaço total de armazenamento de usuários no sistema é igual a $Q + 1$, onde Q é uma variável aleatória que representa o tamanho da fila.

No caso do modelo M/M/m/K/M proposto para avaliar o desempenho do sistema de mitigação de ataques descritos no Capítulo 6, os usuários, em número finito igual a M, representam as transações de sinalização que entram no plano de controle

[2] *Botnet* é a denominação de um conjunto de bots, que podem ser comandados remotamente pelo mesmo atacante (master)

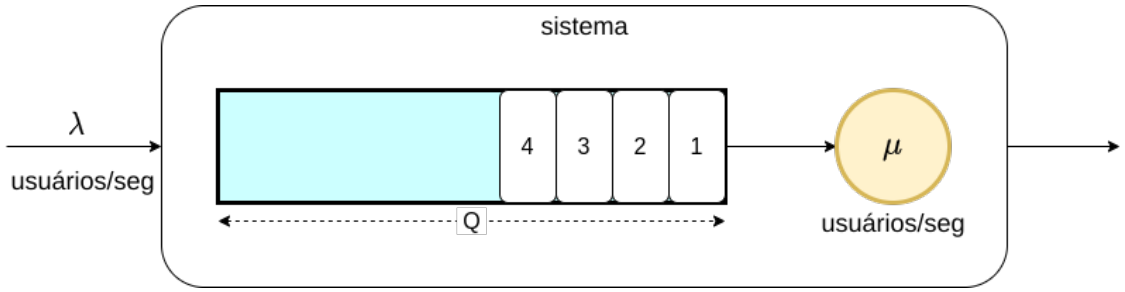


Figura 2.11: Sistema de filas básico com apenas 1 servidor e armazenamento $Q+1$.

do 5G (sistema) segundo uma distribuição de Poisson com taxa λ. Se houver algum servidor ocioso, cada transação segue DIRETO para ser processada por qualquer um dos m servidores idênticos disponíveis (vMMEs) de capacidade fixa igual a μ usuários/segundo. O tempo de processamento das transações é proporcional ao seu tamanho e obedece uma distribuição exponencial.

Um determinado sistema é considerado estável quando a taxa de utilização $\rho = \frac{\lambda}{\mu} < 1$. Ou seja, se a sua capacidade fixa de processamento μ é maior que a taxa de entrada de usuários λ. Se a taxa de entrada de transações (usuários) for maior que a capacidade dos em processá-las, as transações são armazenadas e aguardam na fila até que um dos servidores possa processá-las. Como o sistema tem uma área de armazenamento finita $K = Q+1$, uma transação que chega e encontra o sistema totalmente cheio é descartada sem ser processada, ocorrendo a perda ou bloqueio.

2.12 Teoria de Jogos

Os seres humanos não sobrevivem sem interagirem entre si e em sociedade. Estas interações, que têm o potencial de trazer cooperação e harmonia, também podem desencadear conflitos e guerras. Segundo [53] a Teoria de Jogos estuda justamente as interações estratégicas entre indivíduos em um mesmo grupo, que competem entre si, seguindo determinadas regras por um resultado de interesse comum. Os modelos matemáticos derivados da Teoria de Jogos são largamente utilizados nas mais diversas áreas, incluindo economia, política e biologia para antecipar comportamentos e ajudar na tomada de decisões.

Um jogo estratégico é composto por:

- Um conjunto de jogadores: N.
- Um conjunto de ações: A_i para cada jogador i.
- uma função de ganho: $u_i : A \to \mathbb{R}$ para cada jogador i.

Um modelo clássico dentro da Teoria de Jogos é conhecido como “dilema do prisioneiro". Neste modelo, dois suspeitos de um crime estão presos em celas diferentes

e sem contato nenhum entre eles. Se os dois suspeitos simplesmente ficarem quietos, ambos ficarão presos por dois anos. Se um deles acusar o outro, o primeiro fica livre e o acusado fica preso por quatro anos. Se ambos se acusarem mutuamente, ambos ficarão presos por um ano. A tabela 2.2 abaixo mostra um exemplo de organização deste modelo.

Tabela 2.2: Tabela de estratégia de jogo, demarcando estratégia e recompensas de cada prisioneiro.

		Acusado 1	
		Quieto	Acusar
Acusado 2	Quieto	$(2, 2)$	$(4, 0)$
	Acusar	$(0, 4)$	$(1, 1)$

No exemplo da Tabela 2.2, a recompensa óbvia que cada prisioneiro deseja no jogo é reduzir a função de ganho, que representa o tempo de cada um na prisão u_i. Dentro da Teoria de Jogos, um jogo pode ser definido como a representação completa dos jogadores, do conjunto de regras que condiciona suas estratégias e ações e os pagamentos envolvidos. De acordo com o levantamento proposto em [54], o modelos de jogos podem ser classificados de acordo com a seguinte taxonomia básica:

- Cooperativos - Os modelos cooperativos são caracterizados pela ocorrência de coalisões entre os jogadores, que passam a considerar a possibilidade de colaborarem entre si contra adversário comum.
- Não-ccoperativos - Neste modelos não existem coalisões. Todos são adversários e jogam para atingir o máximo ganho individual ou infligir a máxima pena aos adversários.
  - Estáticos - Os modelos não-cooperativos estáticos se caracterizam por permitirem uma única jogada por vez, por jogador. Os modelos estáticos ainda podem ser classificados com relação à informação sobre os jogadores como Completos ou Incompletos. No caso da modelagem de um ataque de DDoS, onde o defensor a princípio não consegue distinguir sempre um usuário normal de um atacante se passando por um usuário normal, o modelo é de informação incompleta.
  - Dinâmicos - Os modelos dinâmicos permitem mais de uma jogada por jogador por vez, de acordo com o histórico de movimentos (ações) executadas pelos adversários.

2.12.1 Equilíbrio de Nash

O equilíbrio de Nash é o ponto conceitual que descreve o estado estacionário do jogo, onde os jogadores não têm qualquer incentivo para alterarem suas respectivas estratégias, dado que preferem manterem seus prêmios atuais. Assim, encontrar o ponto de equilíbrio de Nash, significa encontrar uma condição de jogo na qual os jogadores estão satisfeitos e com os seus prêmios, e portanto, tendem a se manterem na mesma estratégia de jogo.

2.13 Resumo da Fundamentação

O IDS tem a função de detectar intrusões que possam colocar em risco a integridade, disponibilidade ou confidencialidade de um determinado sistema. O NIDS é um tipo de IDS que utiliza dados da rede (parâmetros do tráfego, cabeçalhos, etc.), para detectar intrusões. Este processo de detecção pode ainda ser através de comparação com padrões pré-configurados (*signature-based*) ou através de análise de comportamento (*anomaly-based*). Enquando os IDSs baseados em assinaturas conseguem melhor desempenho no quesito falso-positivos, os IDS baseados em análise de comportamento conseguem detectar ataques novos (*zero-day attacks*). Os IDS ainda podem operar de forma distribuída (DIDS), auditando diferentes partes de uma rede cooperativamente e correlacionando e combinando os dados coletados para compor informações de detecção mais consistentes e amplas, melhorando o desempenho de detecção do conjunto como um todo.

O BGP FlowSpec é uma funcionalidade no protocolo BGP especificada na RFC 5575. O BGP FlowSpec foi concebido originalmente como uma ferramenta automática de mitigação de ataques de DDoS a partir da divulgação dos fluxos ofensores. Apesar de ser bastante utilizada por ferramentas de mitigação intra-domínio, a utilização do BGP FlowSpec na Internet ainda encontra muitos obstáculos, principalmente relacionados com a confiança nas mensagens recebidas de outros sistemas autônomos, para se decidir por bloquear um determinado tráfego. Na proposta deste trabalho, o BGP FlowSpec entra para formar uma rede sobreposta, permitindo que os alarmes de intrusão emitidos pelos IDSs federados possam chegar de forma normalizada aos seus destinos pela rede BGP. Uma vez chegando ao destino, as mensagens são correlacionadas e combinadas para suportar decisões de mitigação mais acuradas e confiáveis.

O processo de inferência Bayesiana consiste na utilização da Equação de Bayes 2.1 para inferir sobre características desconhecidas de um determinado sistema, a partir de amostras associadas à experimentos deste sistema (verossimilhança) e algum conhecimento prévio sobre o mesmo (probabilidade à priori). A inferência Bayesiana é

fundamental na composição de métodos computacionais relacionados à inteligência artificial, mineração e otimização de dados, com aplicações variadas nos mais diversos domínios. Na inferência Bayesiana, a verossimilhança é função dos estados da variável de interesse, correspondentes aos dados fixos observados no experimento. A probabilidade à priori representa o conhecimento prévio sobre a variável de interesse, antes das observações. A definição deste conhecimento prévio requer dados e/ou conhecimentos específicos de especialistas, o que nem sempre é possível (subjetivos). Além disto, a definição de probabilidades à priori muito informativas e dissonantes com o sistema poder gerar resultados incorretos ou sem sentido. Uma forma de contornar estes problemas é estimar o valor da probabilidade à priori objetivamente, valorizando os dados coletados no processo. A função de distribuição Beta é muito utilizada para representar probabilidades à priori em virtude das suas propriedades de conjugada e seus múltiplos formatos, em função dos parâmetros α e β. hA probabilidade a posteriori, por sua vez, é a probabilidade de ocorrência de um determinado evento, considerando as evidências observadas e o conhecimento prévio antes das evidências.

A fusão de dados consiste num processo automático ou semi-automático de correlação e combinação de dados provenientes de uma única ou de múltiplas fontes com o objetivo de consolidar uma determinada informação e sua significância. O método de fusão de dados proposto por Dempster-Shafer é baseado na teoria da crença evidencial. Isto é, para cada hipótese é atribuída uma massa de crença, de acordo com a(s) evidência(s) que a suporta(m). A utilização deste método de fusão de dados oferece vantagens importantes como a modelagem explícita da "falta de informação" e a capacidade de refletir os acúmulos e a combinação de crenças, independentemente da ordem das evidências. As principais desvantagens do método de fusão de dados proposto por Dempster-Shafer recai sobre a combinação de quadros de discernimento muito grandes e na ocorrência de evidências conflitantes.

O IDS tem a função de detectar intrusões que possam colocar em risco a integridade, disponibilidade ou confidencialidade de um determinado sistema. O NIDS é um tipo de IDS que utiliza dados da rede (parâmetros do tráfego, cabeçalhos, etc.), para detectar intrusões. Este processo de detecção pode ainda ser através de comparação com padrões pré-configurados (*signature-based*) ou através de análise de comportamento (*anomaly-based*). Enquando os IDSs baseados em assinaturas conseguem melhor desempenho no quesito falso-positivos, os IDS baseados em análise de comportamento conseguem detectar ataques novos (*zero-day attacks*). Os IDS ainda podem operar de forma distribuída (DIDS), auditando diferentes partes de uma rede cooperativamente e correlacionando e combinando os dados coletados para compor informações de detecção mais consistentes e amplas, melhorando o desempenho de detecção do conjunto como um todo.

O BGP FlowSpec é uma funcionalidade dentro do protocolo BGP especificada na RFC 5575. O BGP FlowSpec foi concebido originalmente como uma ferramenta automática de mitigação de ataques de DDoS a partir da divulgação dos fluxos ofensores. Apesar de ser bastante utilizada por ferramentas de mitigação intra-domínio, a utilização do BGP FlowSpec na Internet ainda encontra muitos obstáculos, principalmente relacionados com a confiança nas mensagens recebidas de outros sistemas autônomos, para se decidir por bloquear um determinado tráfego. Na proposta deste trabalho, o BGP FlowSpec entra para formar uma rede sobreposta, permitindo que os alarmes de intrusão emitidos pelos IDSs federados possam chegar de forma normalizada aos seus destinos pela rede BGP. Uma vez chegando ao destino, as mensagens são correlacionadas e combinadas para suportar decisões de mitigação mais acuradas e confiáveis.

O processo de inferência Bayesiana consiste na utilização da Equação de Bayes 2.1 para inferir sobre características desconhecidas de um determinado sistema, a partir de amostras associadas à experimentos deste sistema (verossimilhança) e algum conhecimento prévio sobre o mesmo (probabilidade à priori). A inferência Bayesiana é fundamental na composição de métodos computacionais relacionados à inteligência artificial, mineração e otimização de dados, com aplicações variadas nos mais diversos domínios. Na inferência Bayesiana, a verossimilhança é função dos estados da variável de interesse, correspondentes aos dados fixos observados no experimento. A probabilidade à priori representa o conhecimento prévio sobre a variável de interesse, antes das observações. A definição deste conhecimento prévio requer dados e/ou conhecimentos específicos de especialistas, o que nem sempre é possível (subjetivos). Além disto, a definição de probabilidades à priori muito informativas e dissonantes com o sistema poder gerar resultados incorretos ou sem sentido. Uma forma de contornar estes problemas é estimar o valor da probabilidade à priori objetivamente, valorizando os dados coletados no processo. A função de distribuição Beta é muito utilizada para representar probabilidades à priori em virtude das suas propriedades de conjugada e seus múltiplos formatos, em função dos parâmetros α e β. hA probabilidade a posteriori, por sua vez, é a probabilidade de ocorrência de um determinado evento, considerando as evidências observadas e o conhecimento prévio antes das evidências.

A fusão de dados consiste num processo automático ou semi-automático de correlação e combinação de dados provenientes de uma única ou de múltiplas fontes com o objetivo de consolidar uma determinada informação e sua significância. O método de fusão de dados proposto por Dempster-Shafer é baseado na teoria da crença evidencial. Isto é, para cada hipótese é atribuída uma massa de crença, de acordo com a(s) evidência(s) que a suporta(m). A utilização deste método de fusão de dados oferece vantagens importantes como a modelagem explícita da "falta de

informação" e a capacidade de refletir o acúmulos e a combinação de crenças, independentemente da ordem das evidências. As principais desvantagens do método de fusão de dados proposto por Dempster-Shafer recai sobre a combinação de quadros de discernimento muito grandes e na ocorrência de evidências conflitantes.

O desenvolvimento do sistema 4G trouxe significativas melhorias em relação ao 3G, aperfeiçoando as tecnologias de acesso, controle de mobilidade e aumentando as bandas. O plano de controle do 4G também sofreu importantes modificações para suportar as novas demandas. Sua arquitetura linear, formada por entidades com funções bem delineadas (Figura 2.9), consegue processar uma diversidade de protocolos de sinalização, incluindo: informações do sistema, atualização de configuração de mobilidade, controle de acesso, e muitos outros. Esta diversidade de protocolos, que deve ser processada e respondida em tempo real com a mínima latência, têm entre si uma relação de interdependência, que torna o ambiente do plano de controle extremamente complexo. Entretanto, a mesma evolução tecnológica que serviram de base para o surgimento das novas funcionalidades do 4G, também favorecem no desenvolvimento de novos ataques.

O plano de controle do 5G, conhecido como SBA, foi desenvolvido a partir das mesmas premissas de arquitetura do 4G. Entretanto, para viabilizar o fatiamento da rede para diferentes serviços e múltiplos usuários, teve as suas funções básicas divididas e espalhadas para garantir a separação total entre os planos de dados e controle. Significa dizer que, além da complexidade na interação entre as muitas diferentes funções e protocolos, o plano de controle precisa processar ainda mais rapidamente para garantir os desafios de banda ultra-alta e latência ultra-baixa do 5G. Fora isto, vale a pena observar que o fato de gerenciar fatias de infraestrutura para múltiplos usuários (*tenants*), torna a disponibilidade uma das mais importantes questões na segurança do 5G.

Os ataques de negação de serviços (DoS)tem como principal objetivo interromper ou prejudicar a experiência de usuários legítimos na utilização de um determinado sistema. Este objetivo é geralmente atingido através de métodos diretos ou indiretos, causando exaustão de recursos ao sistema. Se o ataque de negação de serviços utilizar múltiplas fontes para atingir os seus objetivos, o ataque é categorizado como distribuído (DDoS). Num ataque distribuído, o atacante mestre recruta agentes escravos distribuídos e os controla de forma coordenada contra um alvo previamente escolhido. Devido ao seu caráter furtivo e disfarçado, os ataques de negação de serviços são de difícil detecção e mitigação. Portanto, pode-se afirmar que os ataques de DDoS contra o plano de sinalização representam uma séria ameaça à disponibilidade dos serviços no plano de controle 5G.

A Teoria de filas é uma ferramenta matemática de modelagem largamente utilizada para avaliar o desempenho de sistemas do tipo cliente/servidor. Um sistema

é considerado estável, se a sua capacidade de processar as demandas μ é maior que a taxa de entrada das mesmas ao sistema λ. Se esta relação não se mantém, o sistema tende a utilizar sua área de armazenamento. Entretanto, se um usuário entra no sistema e o encontra completamente cheio, esta demanda é descartada sem ser processada, ocorrendo a perda ou bloqueio.

A Teoria de Jogos é pode ser definida como um arcabouço de modelos matemáticos utilizados para estudar o comportamento dos jogadores e suas estratégias para obterem o máximo ganho em condições de conflito. Uma das aplicações mais comuns dos modelos de Teoria de Jogos é no estudo de ataques cibernéticos, onde o atacante e o defensor competem entre si com objetivos opostos (jogo estático não-cooperativo). Neste jogo, cada jogador tem direito a uma única jogada por vez, e as consequências de cada jogada interferem diretamente na estratégia do seu oponente. O ponto de equilíbrio de Nash é uma determinada condição durante o jogo, onde os jogadores estão satisfeitos com os seus ganhos e portanto, tendem a manter suas estratégias.

Capítulo 3

Revisão Bibliográfica

Neste Capítulo é feita uma extensa e exaustiva revisão bibliográfica de toda a tese, separada em capítulos.

3.1 Trabalhos Relacionados ao Capítulo 4

O sistema de imunização do corpo humano (HIS) tem inspirado vários trabalhos voltados para detecção de intrusão [22, 55–57]. O HIS se caracteriza principalmente pela sua natureza distribuída e pela capacidade de aprender para se adaptar. Além destas características, que atuam conjuntamente para melhorar o desempenho de detecção, o HIS também possui mecanismos de proteção especias para evitar que ele próprio seja utilizado como um vetor de ataque [55]. A aplicação dos conceitos do HIS para melhorar os sistemas artificiais de imunização (AIS) se concentra na capacidade de distinguir um evento legítimo (*self*) de um evento malicioso (*non-self*), utilizando um algorítimo seleção negativa (NSA).

Os sistemas distribuídos de detecção de intrusão (DIDS) surgiram como uma proposta para reduzir as taxas alarmes Falso-Positivos e Falso-Negativos dos sistemas de detecção, baseada no sistema de imunização humano. Segundo os critérios propostos em [22], a natureza distribuída e independente dos seus IDSs membros garante sistema: (*i*) robustez; a falta de um dos elementos IDS pode ser absorvida pelos demais, (*ii*) configurabilidade; o processo de detecção depende exclusivamente do local que o hospeda para funcionar e (*iii*) escalabilidade; os dados são colhidos e processados localmente em cada IDS membro, mas ajudam o sistema exponencialmente. Uma outra vantagem dos sistemas distribuídos faz referencia à detecção de ataques novos, cuja assinatura ainda não existe na rede (*zero-day attack*). Segundo Igbe *et al.* em [57], considerando uma rede espalhada numa grande região geográfica, um ataque novo num determinado IDS membro, pode não sê-lo para um outro IDS membro geograficamente distante.

A proposta de se correlacionar informações de intrusão provenientes de múltiplos membros distribuídos não é nova, mas vem evoluindo ao longo do tempo. No primeiro trabalho referenciado em [58] propõe-se uma arquitetura híbrida de detecção de intrusões. A arquitetura proposta em [58] recebe e agrega informações de auditoria de múltiplos agentes, localizados de forma distribuída em computadores hospedeiros (HIDS) e nas redes LAN (NIDS). As informações de auditoria chegam até uma entidade central chamada DIDS *director* através de um protocolo chamado CMIP, onde são agrupadas e processadas para suportar ações de defesa. Snapp *et al.* [58] ainda propõem um mecanismo de identificação única de usuário, baseada na tupla *session start, user-id, host-id, time*, para facilitar a detecção, quando um mesmo atacante utiliza múltiplos *logins* diferentes. A principal desvantagem desta proposta é a existência de uma entidade centralizada, que limita a escalabilidade da solução e representa um único ponto de falha. Adotando uma abordagem similar, os sistemas de detecção de intrusões propostos em [59–61] também se baseiam na utilização de agentes de monitoração distribuídos na rede. Na proposta apresentada em [59] estes agentes, diferentemente da proposta anterior, comunicam-se uns com os outros cooperativamente. Por exemplo, se um agente de monitoração de rede levanta uma suspeita sobre uma determinada conexão, ele envia uma mensagem aos demais agentes, que por sua vez, aumentam o seu nível de alerta sobre uma possível intrusão. A arquitetura hierárquica proposta em [60] possui três diferentes entidades: agentes, transceptores e monitores. Os agentes, como nas propostas anteriores, monitoram pontos específicos de um *host* e reportam aos transceptores. Estes, por sua vez, submetem as informações a um processo de redução e reportam para um ou mais monitores, que fazem a correlação das informações recebidas. A arquitetura de detecção proposta em [61], chamada de MAIDA, baseia-se na utilização de agentes inteligentes distribuídos na rede. Os agentes inteligentes são capazes de aprender as características do ambiente onde estão inseridos, se comunicarem cooperativamente com outros agentes e tomar decisões baseadas em auto-aprendizagem.

Centralizar o processamento dos dados vindos de IDSs distribuídos é um processo custoso e muitas vezes inviabiliza sistemas muito grandes, principalmente considerando um conjunto heterogêneo de IDSs. O trabalho proposto em [62] apresenta uma estrutura para abstrair os detalhes de ataques de abuso (baseado em assinaturas) a fim de viabilizar a cooperação entre IDSs heterogêneos e reduzir a carga de processamento no centro de dados. Uma outra vantagem desta abordagem é que, uma vez abstraindo corretamente um determinado tipo de assinatura de ataque, variantes desconhecidas deste tipo de ataque também ser detectadas.

Muitas vezes uma determinada intrusão é apenas uma parte de um plano maior de ataque. As informações de detecção oriundas de um único IDS não oferece uma visão global que permite descobrir este plano. O sistema proposto em [63] tem como

objetivo implementar cooperação entre diferentes IDSs, agrupando, enriquecendo e correlacionando alarmes para gerar um novo alarme sintético, que possa suportar uma decisão de defesa mais completa. Como apenas um módulo de uma plataforma maior de detecção de intrusão (MIRADOR), o sistema proposto (CRIM) não enfrenta os problemas de IDSs heterogêneos tecnologicamente.

Um dos principais desafios no compartilhamento de informações entre os IDSs distribuídos é a infraestrutura de rede. Além de proporcionar conectividade, é fundamental que a comunicação entre elementos aconteça de forma segura. A arquitetura proposta em [64] define uma estrutura hierárquica e sobreposta, composta por quatro entidades distintas: nós centrais; responsáveis pelo compartilhamento das informações de intrusões, nós satelitais; responsáveis pela coleta e envio de dados confiáveis para os nós centrais, colaboradores terrestres; responsáveis pela coleta e envio de dados não confiáveis para os nós centrais, mensagens XML; responsáveis pela interoperabilidade das entidades.

A abordagem apresentada em [65] se baseia no funcionamento de programas (*deamons*) distribuídos nos diversos computadores de uma rede. Além de detectarem eventuais intrusões, estes programas também enviam mensagens de alarme uns para os outros, a fim de avisar seus vizinhos sobre uma possível ameaça. Para a segurança na comunicação direta entre os programas de detecção *Peer-to-Peer* (P2P), o sistema INDRA propõe a utilização de ferramentas de autenticação e criptografia como o *Pretty Good Privacy* (PGP).

Os ataques de negação de serviço (DoS) se tornam cada vez mais perigosos à medida que as redes se tornam mais densas e conectadas. A arquitetura proposta em [66] (CATS) tem como objetivo melhorar a performance de detecção de ataques de negação de serviços (DoS), através da cooperação entre sistemas de detecção autônomos e distribuídos. Partindo de uma arquitetura interna dividida em camadas, o artigo propõe interfaces padronizadas em cada camada, de forma que as mesmas camadas em sistemas de detecção distintos possam se comunicar livremente. A cooperação entre os sistemas de detecção distribuídos pode acontecer em todas as camadas, desde a camada de monitoração de pacotes (baixo nível) até informações de ataques já enriquecidas (alto nível).

Sabe-se que assim como a Internet evolui, os ataques cibernéticos também ficam mais sofisticados. Uma destas sofisticações, conhecida como ataques evasivos, é justamente no sentido de se evitar sua detecção. Os ataques evasivos se aproveitam de eventuais falhas nos sistemas de detecção de intrusão para não serem detectados por estes. Este tema é estudado no artigo em [67], que também propõem a utilização de um sistema distribuído para detecção destes tipos de ataques. O sistema proposto se baseia na arquitetura cliente-servidor para a coleta de dados via WinPCap [68], normalização e processamento integrado dos dados, a fim de gerar alarmes de segurança

para os administradores.

O trabalho apresentado em [69] utiliza o conceito de computação suave (*soft computing*) para propor um sistema de classificação e inferência voltado para a detecção de intrusões num ambiente distribuído. Os dados de intrusão são enviados pelos agentes inteligentes para uma árvore de controladores, organizados hierarquicamente, até chegarem a um controlador central, no topo da árvore. O sistema proposto é modelado utilizando conceitos de lógica *fuzzy*, combinados com diferentes técnicas de aprendizado de máquina. Os resultados obtidos a partir de um conjunto de dados de teste são comparados em relação a outros métodos.

O surgimento e a evolução das redes da próxima geração (NGN) têm alavancado o desenvolvimento de novas tecnologias de processamento, tais como: processadores com múltiplos núcleos, processadores de rede e FPGA. A proposta apresentada em [70] aproveita estas novas tecnologias de processamento para propor um sistema de detecção de intrusão distribuído, utilizando os recursos dos próprios módulos de interface (NIC) dos servidores protegidos. O sistema se baseia na análise de expressões regulares em cada servidor para diminuir a carga de processamento de IDSs mais gerais. Uma outra vantagem desta abordagem é a possibilidade de administrar regras mais específicas, de acordo com as aplicações em cada um dos servidores protegidos.

Ataques baseados em *downloads* e os ataques direcionados por buscas são exemplos de ataques que são invisíveis aos sistemas de detecção usuais. Nestes tipos de ataques, muitas vezes a vítima nem percebe que foi atacada. Os ataques baseados em *downloads* se aproveitam de engenharia social (curiosidade) para induzir uma vítima para um ambiente inseguro, onde algum tipo de software nocivo (*malware*) é instalado no seu sistema. O ataque direcionado por busca se aproveita de informações sensíveis, tornadas públicas através de uma busca maliciosa. A estrutura de análise apresentada em [71] propõe 4 níveis de classificação de ataques, com base na visibilidade destes ataques por suas vítimas. Para os ataques invisíveis, o referido artigo propõe uma estrutura de compartilhamento seguro de informações distribuídas de ataques, que podem ser utilizadas para proativamente aumentar o nível de segurança dos sistemas de proteção.

Um dos principais fatores para o sucesso dos ataques cibernéticos atualmente é a ineficiência dos sistemas de defesa. Neste ponto, ter um sistema de defesa ruim é muitas vezes pior do que não tê-lo. O trabalho apresentado em [72] considera eventuais problemas de atualização e falhas na configuração dos sistemas de defesa para propor um sistema de compartilhamento de informações de ataques. As informações com os registros dos ataques, vindas de IDSs distribuídos, são analisadas por uma entidade central para identificar problemas relacionados com desatualização e falhas de configuração.

O desenvolvimento de novos conceitos tecnológicos básicos para a viabilização da quinta evolução tecnológica das redes de comunicação 5G, como computação ubíqua, computação em nuvem, etc., trouxe também preocupações adicionais com a segurança dos dados. Considerando um ecossistema onde as trocas de dados entre os dispositivos inteligentes e a nuvem definem os serviços oferecidos, um processo de detecção de intrusão que consiga processar tamanha diversidade e volume é absolutamente desafiador. A arquitetura proposta em [73] considera particularmente os novos ambientes de computação ubíqua para conjecturar sobre os principais desafios na detecção de intrusões. A partir destes desafios é apresentada uma proposta baseada no agrupamento e na autenticação dos nós de rede para detecção de anomalias relacionadas a uma eventual ameaça.

A evolução das redes de comunicação para a sua quinta geração tecnológica (5G) traz avanços significativos com relação às taxas de transmissão (de 10 a 100 vezes maior que no LTE), diminuição de latência (menor que 1 ms) e o fatiamento da rede para diferentes locatários. Estes avanços, embasados em soluções tecnológicas como redes definidas por software, virtualização e computação móvel de borda (*Mobile Edge Computing* - MEC), trazem também preocupações adicionais com a disponibilidade dos serviços. Muito mais que simplesmente detectar uma eventual intrusão, as aplicações críticas suportadas pelo 5G requerem antecipação a elas. A proposta apresentada em [74] se baseia nos conceitos de detecção centrada no usuário e *deep learning* para detectar anomalias a partir da análise de fluxos, coletados diretamente dos sistemas de computação móveis de borda. Os dados coletados são então combinados e o resultado da combinação é comparado com sintomas previamente aprendidos.

3.1.1 Estado da Arte

Analisando os principais trabalhos relacionados apresentados na seção anterior, é possível perceber que o tema dos sistemas distribuídos para detecção e intrusões não é novo. Motivados pelos benefícios na utilização deste tipo de arquitetura - principalmente na ampliação do escopo de detecção, robustez, escalabilidade e aumento no desempenho geral de detecção - um grande número de propostas surgiram e continuam surgindo todos os anos. Entretanto, apesar do grande número de propostas e da diversidade de abordagens, ainda não se pode dizer que existe um sistema de detecção distribuído "de facto", capaz de se transformar até num serviço de Internet, como é o caso do serviço de resolução de nomes (*Domain Name Service* - DNS). As razões para isto estão relacionadas principalmente aos desafios tecnológicos de se criar uma rede de conectividade ampla e simples o suficiente para escalar e permitir a plena cooperação entre os membros.

A grande maioria dos trabalhos relacionados tem como base a comunicação entre agentes distribuídos, que, de forma cooperativa ou não, trocam informações numa estrutura hierárquica para refinar ou enriquecer as informações de detecção de ameaças. Os agentes distribuídos, inteligentes ou não, são responsáveis em monitorar e detectar as intrusões, de acordo com os seus respectivos métodos de detecção, que podem variar em relação ao tipo de dado que é analisado (dados do sistema operacional - HIDS, ou dados de rede - NIDS), até a forma de analisar uma eventual ameaça, comparando padrões pré-configurados (assinaturas) ou anomalias de comportamento. De forma geral, as propostas existentes criam novos IDSs (na figura dos agentes) e os interligam através de redes sobrepostas para poderem interagir, até se poder chegar a uma informação mais confiável em relação a uma ameaça. Este tipo de abordagem cria, na verdade, uma estrutura à parte, com recursos e padrões específicos, destinadas à detecção de intrusões. Além de trazer dificuldades na operação e atualização dos agentes, dependendo do tamanho da estrutura de detecção, este tipo de abordagem dificulta que a heterogeneidade intrínseca da Internet seja um fator para aumentar o escopo de detecção e facilitar a escalabilidade da rede de detecção.

A abordagem proposta neste trabalho de doutorado difere dos trabalhos relacionados no sentido de utilizar a própria estrutura de conectividade da Internet para aumentar a probabilidade de detectar um potencial ataque cibernético contra um determinado alvo. A rede sobreposta que possibilita o ambiente federativo, composto por IDSs autônomos que cooperam entre si para detectar intrusões, é baseada numa estrutura bastante conhecida e de fácil implementação (FlowSpec BGP). Estas características facilitam o escalonamento dos membros federados e ampliam a superfície de detecção da plataforma, viabilizando o sistema distribuído como uma alternativa para aumentar o desempenho de detecção de intrusões e reduzir a vulnerabilidade dos sistemas monolíticos tradicionais. Os resultados obtidos através de modelos analíticos baseados na combinação da Teoria Bayesiana com o método de fusão de dados proposto por Dempster-Shafer mostram uma significante melhoria nas métricas de desempenho de detecção, condicionada à precisão dos IDSs membros (PPV_i) e ao número de IDSs federados. Esta proposta de doutorado encontra-se publicada em dois artigos científicos [75, 76]. A Tabela 3.1 abaixo mostra de forma resumida as principais características dos trabalhos relacionados com a proposta deste capítulo.

3.2 Trabalhos Relacionados ao Capítulo 5

O ecossistema BGP já sobrevive há mais de três décadas e é considerado atualmente uma referência global na questão de redundância, capaz inclusive de sobreviver a

Tabela 3.1: Tabela comparativa dos principais trabalhos relacionados na Seção 3.1.

Trabalho	Abordagem	Arquitetura	Vantagens	Limitações
Snapp *et al.*[58]	Sistema híbrido (HIDS + NIDS) que se comunicam com o *director* pelo protocolo CMIP	Tipo estrela, como um elemento processador centralizado	Comunicação normalizada	Contém um único ponto de falha
Crosbie *et al.*[59]	Programas agentes autônomos e leves que cooperam entre si	Distribuída *full-mesh*	Escalável e descentralizada	Escopo de uso limitado a uma LAN
Balasubra. *et al.*[60]	Agentes autônomos organizados hierarquicamente enviam infromações para os monitores	Distribuída hierárquica	Escalável e com processo de redução de dados	Escopo de uso limitado a uma LAN
Labiod *et al.*[61]	Agentes autônomos e inteligentes distribuídos em domínios	Distribuída com processamento centralizado	Auto-aprendizado implementado nos agentes	Muito complexo e de difícil implementação
Ning *et al.*[62]	Abstração de assinaturas para normalizar comunicação	Distribuída e hierárquica	Pode detectar ataques novos	Depende de pré-conhecimento para montar as assinaturas
Cuppens *et al.*[63]	Clusterização e correlação de alarmes	Distribuída e hierárquica	Reduz o número de alarmes	Depende de uma padronização de comunicação dos IDSs membros
Yegneswaran *et al.*[64]	Monitoração de IPs inválidos (*spoofing*) para criação de *blacklists*	Distribuída e hierárquica	Amplitude global	Depende de uma rede sobreposta específica
Janakiraman *et al.*[65]	Agentes de software nos computadores que compartilham informações de intrusões	*Peer-to-peer*	Comunicação segura entre os agentes	Amplitude restrita
Dressler *et al.*[66]	Padronização da taxonomia de assinaturas	Autônoma e cooperativa	Permite rastrear a fonte do ataque	Amplitude global
Basicevic *et al.*[67]	Em estrela	Cliente-servidor	Permite detectar ataques evasivos	Restrito à LAN
Silva *et al.*[75, 76] - proposta do doutorado	Utiliza a autonomia e a conectividade da Inernet para aumentar a superfície de detecção	Distribuída *full-mesh*	Fácil implementação global	Necessidade do protocolo FlowSpec

um ataque nuclear [77]. Entretanto, apesar dos avanços tecnológicos durante este período, este ecossistema ainda apresenta alguns riscos, decorrentes de vulnerabilidades importantes, como vazamento de rotas e sequestro de prefixos [78]. A utilização de técnicas de inteligência artificial e aprendizado de máquina para endereçar problemas relacionados com o protocolo BGP têm crescido nos últimos 10 anos, em especial para enfrentar as ameaças de segurança [79]. A maioria destes trabalhos tem por objetivo detectar e classificar anomalias utilizando algoritmos de aprendizado supervisionado, que necessitam passar por uma fase de treinamento, utilizando conjuntos de dados (*datasets*) rotulados [80–83].

O modelo de aprendizado proposto em [80] compara *Support Vector Machines* (SVM) e *Hiden Markov Model* (HMM) para detectar anomalias na rede BGP. Os modelos propostos são treinados com um *dataset* construído a partir de 37 atributos diretos e de volume, extraídos das mensagens BGP de coletores mundiais, como o RRC04 [84]. Este conjunto de dados passar por um processo de classificação

de atributos, baseado no algoritmo de Mínima Redundância, Máxima Relevância (mRMR) para selecionar os atributos mais relevantes. Os resultados dos testes apresentados, considerando as métricas *f-score* e acurácia, demonstram a eficiência da proposta para detectar ataques do tipo *worm*.

A estrutura de detecção de anomalias proposta em [81] consiste em 3 fases: extração seleção e geração de atributos, utilizando SVM como modelo de aprendizado. O conjunto de dados utilizado para treinar o modelo, composto por 18 atributos de volume, foi rotulado a partir de registros de mensagens BGP colecionadas durante eventos importantes de ataques, como: Nimda (18 de Setembro de 2001), Slammer (25 de Janeiro de 2003) e Codered (19 de Julho de 2001). Na fase de extração, foi utilizado um modelo de normalização baseado em na média e no desvio padrão de cada atributo, para resolver problemas de escala. Os resultados apresentados, comparando as taxas de verdadeiro-positivos e falso-positivos, obtidas utilizando-se (*i*) apenas 6 atributos e (*ii*) todos os atributos, mostram uma grande vantagem da opção parcial (*i*), ao invés de utilizar todos os atributos.

O conjunto de dados e o processo de seleção de atributos afetam de forma significativa o desempenho dos modelos de aprendizado. Com base nesta afirmação, o trabalho proposto em [82] extrai um conjunto de 37 atributos diretos e volumétricos, significativos física e estatisticamente, a partir de dados brutos de mensagens BGP. Considerando o baixo desempenho do SVM em processar *dataset* com um grande número de atributos (dimensões), o artigo propõe um algoritmo baseado em Fisher/Markov para selecionar os melhores atributos, aumentando as distâncias entre as aglomerações e diminuindo as distâncias dentro das aglomerações. Da mesma forma utilizada em [81], o processo de normalização proposto também considera a média e o desvio padrão. Os resultados apresentados mostram valores de acurácia e e *F1-Score*[1] próximos de 92% e 96%, respectivamente.

Aproveitando a própria semântica do protocolo BGP, que utiliza mensagens de atualização BGP condensadas para modificar a aumentar (*A-updates*) ou encolher (*W-updates*) a topologia da Internet, o trabalho apresentado em [83] propõe um modelo de aprendizado baseado em rede neural para detecção de anomalias causadas por ataques cibernéticos em curso na Internet. O *dataset* de treinamento utilizado na proposta [83] possui apenas 8 atributos volumétricos, extraídos de bases de dados globais como em [84], que considera apenas aspectos de alto nível do protocolo, como: número de mensagens de *update* que só aunciam prefixos, ou que excluem pelo menos um prefixo. Os resultados dos testes mostram níveis de acurácia para

[1] A métrica *F1-Score* é calculada a partir da soma ponderada de PPV e TPR,

$$F1 = 2 \times \frac{PPV \times TPR}{PPV + TPR}$$

.

detectar os ataques Codered, Mimda e Slammer próxima de 98%.

Por fim, além de propor uma abordagem baseada em Teoria de Grafos para detectar anomalias de tráfego baseada em mensagens BGP, o trabalho proposto em [85] compara as principais técnicas de aprendizado de máquina com esta finalidade. Assim como todas as outras propostas descritas nesta Seção 3.2, o conjunto de dados utilizado no trabalho em [85] também foi obtido em [84], considerando a data e o período de cada evento investigado. Atributos como: centralidade, grau, intermediação (*betweeness*), proximidade (*closeness*), entre outros, foram utilizados para montar um (*dataset*) rotulado e selecionado, utilizado para treinar os modelos testados. O processo de seleção de atributos também utilizou algoritmos da família mRMR, combinado com ordenação de Fisher [86]. O resultado da comparação dos diferentes modelos de aprendizado de máquina mostram que o algoritmos *Multi-Layer Perceptron* (MLP) alcançou os melhores resultados em termos de acurácia, chegando em $99,01\%$.

A tese de doutorado proposta em [87] endereça dois problemas clássicos em se tratando de classificação de anomalias com base em dados de tráfego BGP:

1. A ausência de um único *dataset* rotulado para diferentes classes de anomalias.

2. A ausência de uma estrutura padronizada para detectar e classificar anomalias a partir do *dataset*.

Apesar de se basear numa ideia similar aos trabalhos já apresentados nesta Seção 3.2, o conjunto de dados proposto pelo autor em [87] é público, com uma maior dimensão e mais completo. Contando com 66 atributos, diretos, volumétricos e estatísticos, o conjunto de dados pode ser construído através de uma ferramenta também proposta pelo autor, a partir de dados brutos públicos bem conhecidos como em [84]. A estrutura de aprendizado proposta explora as principais vantagens das diferentes arquiteturas de redes neurais *Long Short-Term Memory* (LSTM) para ampliar o espectro de anomalias detectáveis com acurácia de 87,4% em tempo real.

3.2.1 Estado da Arte

A Tabela 3.2 abaixo mostra de maneira resumida as principais similaridades e diferenças das propostas apresentadas na Seção 3.2.

Apesar das grandes similaridades concepcionais nos trabalhos apresentados na Seção 3.2, as diferenças sutis e estratégicas dos modelos de aprendizado, principalmente na construção dos seus *datasets* rotulados, ajudaram a compor o arcabouço central desta proposta, considerando as suas peculiaridades de utilização:

Tabela 3.2: Tabela comparativa dos trabalhos relacionados na Seção 3.2.

Trabalho	Modelo de Aprendizado	Conjunto de Dados	Atributos	Escopo	Tempo Real
BGP Anomalies [80]: 37 atributos	SVM e HMM	Privado	37 atributos diretos (AS_PATH) e de volume de mensagens BGP	Amplo	Não
Framework [81]: 18 atributos	SVM com seleção de atributos	Privado	18 atributos volumétricos de mensagens BGP	Mediano	Sim
Application [82]: 37 atributos	SVM com seleção de atributos Fisher/Markov	Privado	37 atributos diretor e de volume de mensagens BGP	Mediano	Não
Neural Network [83]: 8 atributos	MLP	Privado	8 atributos de volume de mensagens BGP	Mediano	Sim
Deep Learning [87]: 66 atributos	Redes neurais LSTM	Público	66 atributos diretos, volumétricos e estatísticos	Amplo	Sim

- Impossibilidade de lidar com atributos volumétricos em decorrência do baixo número de mensagens BGP-FlowSpec correlacionáveis para serem combinadas no AS de destino, eventual alvo de um ataque cibernético.
- A necessidade de se obter resultados em tempo real a partir das informações básicas do cabeçalho da mensagem, já que seu conteúdo já é utilizado na parte de correlação, antes do processo de combinação de massas de crença.

3.3 Trabalhos Relacionados ao Capítulo 6

Apesar dos efeitos, que podem ser devastadores, o caráter mimético dos ataques de DDoS dificultam enormemente o processo de detecção. Portanto, um sistema de defesa que parte precipitadamente para ações de bloqueio pode comprometer tráfego legítimo e deliberadamente efetivar o sucesso do ataque. Neste sentido, a adoção da estratégia *defense-in-depth* com medidas para frear ou enfraquecer o ataque pode ser uma alternativa interessante no processo de mitigação, minimizando os efeitos colaterais. O escalonamento de recursos do alvo para absorver os efeitos do ataque é uma das medidas de proteção que fazem parte desta estratégia. Os levantamentos apresentados em [88, 89] descrevem e classificam estas medidas em taxonomias próprias, relacionando-as a outros mecanismos de mitigação mais complexos.

Apesar dos desafios de segurança inerentes ao novo cenário de virtualização [90, 91], os avanços tecnológicos na área de computação e conectividade em nuvem também abrem uma série de oportunidades para melhorar os processos de desempenho dos sistemas de comunicação móvel. O trabalho apresentado em [92],

aproveita o ambiente virtualizado do novo plano de controle LTE para propor SCALE, uma estrutura de escalonamento de vMMEs transparente e flexível, com o objetivo corrigir desvios e manter compromissos de desempenho para o tráfego IoT. SCALE [92] endereça o balanceamento de carga entre múltiplas máquinas virtuais dividindo o MME em dois novos blocos funcionais, para processamento (MMP - MME *processing*) e balanceamento de carga (MLB - MME *load balance*). Adotando uma abordagem similar, a proposta apresentada em [93] adota o escalonamento de MMEs como uma alternativa para controlar sobrecargas no plano de controle 5G. Na abordagem descrita em [93], o MME é decomposto é vários micro-serviços e o gerenciamento dos estados de processamento do tráfego é mantido centralizado numa base de dados.

Alguns anos antes do desenvolvimento do sistema 5G (2016), já se notava um incômodo na academia com as vulnerabilidades do plano de sinalização do sistema móvel celular. Segundo pesquisa citada no trabalho em [94], 51% das operadoras móveis permitem que os equipamentos conectados sejam escaneados a partir da Internet, o que abre caminho para ataques contra o plano de controle, explorando transações como *paging*, requerimento de serviços (*service requests*) e RRC como amplificadoras de sinalização. Escudero-Andreu *et al.* [94] ainda argumenta que este problema tende a ser ainda mais grave no 5G, afetando principalmente as novas aplicações que requerem baixa latência e altos volumes de tráfego. No trabalho apresentado em [94], os autores propõem um sistema de detecção e mitigação de ataques de DDoS contra o plano de controle do 5G baseado no rastreamento das transições de *timeout* nas transações de sinalização. Elevadas taxas de *timeouts* pode ser um comportamento comum, que caracteriza um ataque de negação de serviços. Através de modelos analíticos baseados em análise estocástica, o artigo propõe a desconexão temporária do móvel que apresenta este tipo de mau comportamento como mecanismo de mitigação.

A análise proposta em [95] aborda o grande número de mensagens de controle de recursos de rádio (RRC - *radio resourses control*) trocadas entre a rede de acesso e o núcleo para negociar e estabelecer canais lógicos entre o equipamento do usuário (UE - *user equipment*) e a rede de dados. Nesta análise, Ettiane *et al.* propõe a sintetização desta transação como uma forma de amplificar um possível ataque de negação de serviços contra o plano de controle. Os autores ainda comparam três diferentes métodos de proteção, elegendo a randomização do alvo como o método de proteção mais eficiente contra este tipo de ataque. Quase na mesma linha, porém endereçando um problema mais específico, o trabalho apresentado em [96] propõe um ataque de negação de serviços de sinalização contra o EPC baseado no elevado número de mensagens para estabelecer portadoras dedicadas (*dedicated bearer*) entre o UE e a rede de dados. Após estabelecida uma ou mais portadoras dedicadas

(no máximo 8), o atacante simplesmente se desconectava, fazendo com que o EPC fizesse a desalocação dos recursos depois de um tempo (20 segundos). Este processo de repetia continuamente após este tempo, provocando a saturação do plano de controle. No mesmo artigo, Bassil *et al.* apresentam um modelo de simulação para comprovar a efetividade do ataque e também para propor um método de detecção baseado na análise estatística de tráfego das portadoras dedicadas.

Uma das principais contribuições da arquitetura baseada em IP do 4G é a sua capacidade de permitir a mobilidade entre diferentes tecnologias de acesso, tais como WiMax (*Worldwide Inter-operability for Microwave Access*), Wifi (*Wireless Fidelity*), GSM (*Global System for Mobile communications*) e UMTS (*Universal Mobile Telecommunications System*). Entretanto, o processo de identificação dos móveis migrando de outras tecnologias de acesso e a integração nativa das redes 4G com a Internet cria também viabiliza ataques de negação de serviços de sinalização contra a camada de controle de recursos de rádio do 4G (RRC - *Radio Resources Control*). O artigo apresentado em [97] desenvolve um sistema de filas genérico para modelar a complexa troca de protocolos na camada RRC, incluindo diferentes tipos de usuários e comportamentos. Aproveitando o modelo de simulação, Pavloski *et al.* também propõem dois mecanismos de detecção baseados no número de alocações de canal e no monitoramento da banda utilizada. Quase na mesma linha, a proposta apresentada em [98] descreve um sistema de detecção de ataques contra a camada RRC, que visa esgotar os recursos do plano de controle. O sistema de detecção propõe a criação de métricas específicas de consumo de recursos, que são combinadas de diferentes lugares da rede através da Teoria da Fusão de Dados de Dempster-Shafer para se chegar numa posição consolidada sobre a ameaça.

Mais recentemente, considerando a funcionalidade de múltiplos locatários do 5G compartilhando uma mesma infraestrutura, os autores em [99] analisam o impacto mútuo entre os domínios de um ataque de DDoS. Apesar de serem logicamente isolados e dependendo da intensidade do ataque, a análise comprovou que os ataques de negação de serviços de sinalização podem comprometer domínios adjacentes. Nesta análise, Sattar *et al.* ainda propõem um modelo matemático para otimizar a alocação das funções virtuais de rede (VNF - *Virtual network function*) para aumentar o isolamento intra e inter-domínio.

3.3.1 Estado da Arte

Os principais trabalhos relacionados com a mitigação dos efeitos de ataques de DDoS de sinalização apresentados na seção anterior têm como base a implementação de contra-medidas de bloqueio ou restrições. A não ser os trabalhos apresentados por Ettiane *et al.* [95] e Sattar *et al.* [99], que propõem a randomização dos alvos e o

isolamento de domínios de abstração, respectivamente, como mecanismos de proteção contra ataques de sinalização, a maioria das propostas já parte para ações mais duras e definitivas, a partir da caracterização de um ataque e o seu alvo. Isto é, os efeitos colaterais destas medidas podem comprometer tráfego legítimo. Já as abordagens baseadas em escalonamento de recursos apresentadas nos trabalhos de Banerjee *et al.* [92] e Amoch *et al.* [93] não têm como objetivo explícito e direto a mitigação dos efeitos de um ataque de negação de serviços de sinalização. Apesar de não fazer parte o escopo da proposta apresentada neste Capítulo 6, a parte de detecção deste tipo de ataque também é crucial para o sucesso das medidas de proteção a serem reativamente implementadas. Alguns trabalhos apresentados na seção anterior ainda propõem sistemas de detecção de ataques de sinalização [94, 96, 97]. Considerando o ecossistema distribuído e a essência furtiva dos ataques de DDoS, a parte de detecção é endereçada no Capítulo 6, como um elemento disparador do jogo de mitigação. Ainda que reativa, uma proposta de mitigação que adota uma estratégia proativa, mantendo recursos extra ociosos para serem utilizados em caso de um ataque, apresenta uma nova linha de ação, valendo-se do ambiente virtualizado do plano de controle 5G.

Diferentemente dos trabalhos relacionados, a abordagem proposta no Capítulo 6 [45] se baseia no conceito utilizado pelas companhias seguradoras para mitigar um ataque, através da ampliação na capacidade de processamento do plano de controle. Antecipando o escalonamento de recursos extra, que ficariam ociosos durante o regime normal de operação, a estratégia baseada no conceito DiD atua sobretudo para “enfraquecer" ou “frear" o ataque, sem comprometer o tráfego legítimo. Do ponto de vista do defensor, a ideia é escalonar recursos de forma inteligente, de acordo com a análise do comportamento esperado do agressor.

O desempenho da estratégia proposta é avaliado através de modelos analíticos, combinando a análise do comportamento dos jogadores (atacante e defensor) num jogo não-cooperativo, com os efeitos da mitigação sobre ataques de DDoS de diferentes escalas. A análise comportamental dos jogadores captura os pontos de equilíbrio durante o ataque (Equilíbrio de Nash [100]), nos quais nem o atacante, nem o defensor sentem-se estimulados a mudar suas respectivas estratégias de jogo. A análise dos efeitos disruptivos do ataque, bem como da efetividade do processo de mitigação, é feita através da extrapolação dos resultados do protótipo de testes, combinando-se diferentes cenários do número de *bots* com vMMEs disponíveis no plano de controle. Em resumo, os principais objetivos do sistema proposta são:

- Oferecer um tempo precioso para que os times de segurança possam melhorar os sistemas de defesa, sem comprometer o tráfego legítimo.

- Desestimular o atacante, e esperar por uma eventual desistência do mesmo,

aumentando o custo/benefício da sua tentativa.

Adicionalmente, manipulando o nível de atração de tráfego dos vMMEs (*weight factor*) através do parâmetro de capacidade relativa (*relative capacity* [51]), é possível implementar um sistema de balanceamento de carga, dispensando sistemas complexos de sincronização de estados ou desintegrando a arquitetura do plano de controle. A eficiência e a eficácia da manipulação do fator de peso dos vMMEs são demonstradas através de testes de carga num protótipo, constituído por 3 simuladores de *enobeBs*, cada um deles com um certo número de *smartphones*, conectadas a um EPC. A Tabela 3.3 logo a seguir mostra um panorama comparativo dos trabalhos relacionados.

Apesar da desvantagem explicitada na Tabela 3.3, o sistema de mitigação proposto neste trabalho, referenciado no artigo [45], pode ser uma importante opção para frear o ataque, seguindo a estratégia de segurança DiD. Maiores detalhes sobre a arquitetura do sistema serão apresentados na próxima seção.

Tabela 3.3: Tabela comparativa dos trabalhos relacionados na Seção 3.3.

Trabalho	Abordagem	Estratégia de defesa	Metodologia de teste	Vantagens	Desvantagens
SCALE [92]: Virtualização de MME	Garantia dos níveis de desempenho requeridos pelo tráfego de sinalização	Separando o controle das interfaces padrão 3GPP e o endereçamento dos equipamentos de usuários do processamento MME	Protótipo de testes e modelos probabilísticos	Balanceamento de carga entre os vMMEs	Altera consideravelmente a arquitetura do vMME
CNS-MME [93]: Escalonamento automático baseado do MME baseado na nuvem	Garantindo os níveis de serviço para o tráfego de sinalização 5G	Separando as funcionalidades do vMME em micro-serviços baseados em conteiners	Simulador NFV-LTE-EPC	Serviços de baixo consumo de recursos	Altera consideravelmente a arquitetura do vMME
Mecanismos de proteção para o 3G [95]: Análise comparativa	Detecção e mitigação de ataques de DoS contra o plano de controle 3G	Randomização, CUSIN e inspeção do pacote IP	N/A	N/A	N/A
Detecção e mitigação de ataques de sinalização [94]: Transições de sinalização	Detecção e mitigação de ataques de DoS contra o plano de controle 5G	Contagem das transições sucessivas de sinalização que são desconectadas por inatividade	Modelos analíticos baseados em modelos estocásticos	Não requer alteração de arquitetura	Requer a implementação de um mecanismo de contagem das transições. Além disto, pode afetar tráfego legítimo
Detecção de ataques de sinalização LTE [96]: explorando as portadoras lógicas	Detecção de um tipo de ataque de sinalização que pode interromper os serviços do plano de controle LTE	Detecção baseada em análise semântica	Modelo de simulação OPNET	Não requer alteração de arquitetura	Detecta apenas os ataques de portadoras lógicas
Fatia de rede segura [99]: Efeitos mútuos dos ataques de DDoS sobre as fatias de rede	Configuração de recursos para minimizar os efeitos mútuos intra e inter-fatias em caso de ataques de DDoS de sinalização	Isolamento de fatias de rede	Simulador NS3 e modelos de otimização	Não requer alteração de arquitetura	Resource-intensive VNFs in the same host might reach their maximum load at similar times
REPEL [45]: Escalonamento inteligente de recursos no plano de controle	Garantia da disponibilidade dos serviços do plano de controle	Escalonamento inteligente de vMMEs para desencorajar o ataque	Protótipo de EPC baseado na plataforma OpenAirInterface, modelos analíticos baseados em Teoria de jogos e filas	Não requer mudança de arquitetura, nem compromete tráfego legítimo	O atacante pode não parar de atacar, mesmo não obtendo os resultados esperados

Capítulo 4

Sistema Distribuído Federativo para Detecção de Intrusões

Apesar da grande diversidade de dados estatísticos e de estudos envolvendo ataques cibernéticos, ainda é possível notar um incômodo consenso em relação à ameaça crescente que isto representa para a evolução tecnológica da Internet. Informações contidas no relatório em [101] mostram que o número de ataques cibernéticos cresce com taxas maiores ano após ano. Neste preocupante contexto, onde a conectividade destaca-se como insumo básico da evolução tecnológica, o ataque de negação de serviços (DoS) surge como uma importante ameaça à disponibilidade dos sistemas e como vetor de outros ataques. Considerado um dos ataques mais difíceis de se detectar, e consequentemente de se proteger, o ataque distribuído de negação de serviços (DDoS) é um dos que mais cresce atualmente. Informações contidas em [102] mostram um número superior a 900 eventos num só dia. Analisando a geografia destes ataques, sabe-se que mais de 95% dos ataques direcionados à alvos do Brasil provêm de fora do país [103], sendo 66% destes com origem na Ásia. Pode-se então especular que a mesma conectividade que fundamentou o projeto original da Internet [104] e que foi capital para sua universalização [105], agora também ajuda os atacantes para iniciarem um ataque para qualquer alvo conectado, partindo de qualquer parte do globo.

O perigo atual dos ataques cibernéticos para a sociedade de modo geral é particularmente importante para os provedores de acesso à Internet (ISPs). Os provedores cumprem hoje um importante papel na universalização da Internet, distribuindo o acesso aos usuários através das diversas tecnologias de comunicação fixa ou móvel. Atualmente a Internet conta com cerca de 60.000 sistemas autônomos (ASs), destes, cerca de 10.000 são de provedores de acesso. Os ataques cibernéticos são particularmente perigosos para os provedores de acesso, pois além de terem seus próprios bens ativos ameaçados, ainda podem ser acusados de transportar os ataques para os seus próprios clientes. Para enfrentar esta ameaça, os provedores de acesso investem em

sistemas de detecção de intrusão (IDS).

O IDS pode ser definido como um sistema automatizado de segurança e defesa, que detecta atividades consideradas suspeitas dentro de um determinado perímetro de proteção, e envia alarmes aos sistemas de defesa. O perímetro de proteção pode variar desde um único computador (*Host-based Intrusion Detection Systems* - HIDS) até uma rede inteira com vários elementos (*Network-based Intrusion Detection Systems* - NIDS). Os sistemas de detecção de intrusão variam bastante em relação à metodologia de detecção [20]. A metodologia baseada em detecção anomalias monitora o comportamento de parâmetros da rede como: tráfego, variáveis de sistema operacional, acessos, etc. e compara com perfis previamente programados e/ou aprendidos. Variações consideradas anormais entre o comportamento monitorado e o perfil de referência são alarmados como potenciais intrusões. Os sistemas baseados em assinaturas monitoram continuamente os parâmetros de rede, comparando o comportamento monitorado com padrões pré-programados, conhecidos como *assinaturas*. Neste caso um alarme é enviado sempre que o comportamento monitorado se conformar como uma determinada assinatura.

Mesmo contando com toda esta diversidade e as muitas melhorias tecnológicas atuais, como a utilização de inteligência artificial e técnicas de processamento em tempo real, os IDSs ainda não são capazes de antecipar informações de intrusões ou detectar ataques novos, conhecidos como "ataques de dia zero" (*zero-day attacks*) [106]. Para se protegerem e protegerem os seus clientes destes ataques, os provedores de acesso investem esforços na mineração e correlação de dados provenientes de redes sociais, fóruns e sistemas privados de avisos ou Equipes de Resposta às Emergências Cibernéticas (CERT). Entretanto, como se trata de uma tarefa lenta e complexa, as informações nem sempre chegam rápido o bastante para se construir uma estratégia de defesa adequada.

Além da ameaça dos ataques novos, o desempenho de detecção continua sendo um desafio, mesmo para os sistemas de detecção mais modernos e poderosos. O desempenho de detecção de um IDS pode ser aferido tomando como base métricas específicas, extraídas de testes de confrontação. Nestes testes, uma base de dados de eventos previamente rotulados (*dataset*) é submetida ao IDS e os resultados obtidos são então organizados numa tabela conhecida como matriz de confusão. Métricas como precisão (PPV)
abbrevPPV*Positive Prediction Value*, taxa de alarmes verdadeiro-positivos (TPR), falso-positivos (FPR), verdadeiro-negativos (TNR) e falso-negativos (FNR) entre outras, são úteis para avaliar o desempenho de detecção do IDS. Enquanto as métricas verdadeiras inferem sobre a eficiência do sistema no processo de detecção, as métricas falsas permitem avaliar os custos e os riscos inerentes a uma falha da detecção [107].

Dois outros pontos que também devem ser bem endereçados no projeto de segurança de um provedor de acesso se referem ao perímetro de proteção e a segurança do próprio IDS. Os sistemas de detecção comumente utilizados operam monoliticamente nas bordas de uma dada região que se deseja proteger. Isto é, além da possibilidade deles próprios serem os alvos do ataque, uma vez detectada a intrusão, esta já pode estar perigosamente perto demais do seu alvo. A ampliação do perímetro de detecção é um dos objetivos do trabalho proposto em [108], que estuda a localização dos sensores de detecção dentro da rede de modo a maximizar o desempenho de detecção.

Os sistemas distribuídos de detecção de intrusões (DIDS) surgiram na década de 90 como uma solução para resolver o problema da arquitetura monolítica dos IDSs e para aumentar o seu desempenho de detecção, com base na colaboração entre os membros e na ampliação da superfície de detecção. Os sistemas distribuídos de detecção de intrusões funcionam intercambiando informações cooperativamente de modo a melhorar a acurácia da detecção, reduzir a taxa de alarmes falso-positivos, reduzir a probabilidade de falhas e principalmente aumentar a probabilidade de se detectar ataques de dia zero. Inspirados, na sua maioria, no princípio básico de funcionamento do sistema de imunização do corpo humano [22, 55, 56], muito esforço têm sido empenhado na busca de uma arquitetura aberta, capaz de se transformar num padrão [58–67]. No entanto, apesar do grande número de propostas, ainda não é possível dizer que há uma arquitetura viável, capaz de suprir integralmente as necessidades de segurança de hoje para a Internet. Desafios relacionados com o *overhead* de comunicação entre os membros, escala, diversidade tecnológica e vulnerabilidades de segurança decorrentes da própria arquitetura distribuída dificultam enormemente a concepção de um sistema global de detecção de intrusões.

Segundo a análise de Kim *et al.* em [22], um sistema distribuído de detecção de intrusões precisa juntar três características básicas para ser considerado eficiente, eficaz e viável:

- Autoconfigurável - o sistema precisa se autoconfigurar para continuar “vivo"e operacional, mesmo após a entrada ou saída de agentes do sistema.

- De fácil implementação - um sistema leve e de fácil implementação conseguirá rapidamente escalar, agregando membros para se tornar suficientemente amplo e distribuído.

- Distribuído - quanto mais distribuídos e heterogêneos forem seus membros, maiores as chances de detecção de ataques. Mesmo um ataque de dia zero numa determinada localidade, pode não mais o ser numa outra localidade geograficamente distante.

Com base nas características mencionadas acima e nas principais propostas de sistemas distribuídos de detecção de intrusões, é apresentada uma arquitetura baseada numa federação de IDSs membros, que embora heterogêneos, cooperam entre si, disseminando informações de fluxos maliciosos que atravessam seus sistemas autônomos (ASs). Fundamentada na estratégia de defesa DiD, a abordagem proposta tira proveito da diversidade dos IDSs membros e da ampla conectividade da Internet, a mesma que em tese ajuda um atacante a iniciar um ataque de qualquer parte do globo, para ampliar a superfície de detecção, criando uma linha de frente de proteção contra ataques cibernéticos. As informações do fluxo considerado malicioso (*flow features*) são embutidas nas mensagens padronizadas de atualização do protocolo *FlowSpec* no campo NLRI e disseminadas para toda a Internet, tirando proveito da amplitude e permeabilidade globais do BGP (*Border Gateway Protocol*). Quanto maior for o número de ASs atravessados pela fluxo ofensor, potencialmente maior será o número de mensagens BGP recebidas no AS alvo e maiores as chances de detecção do sistema, considerando a existência de pelo menos um IDS federado em cada AS atravessado.

Apesar da heterogeneidade e alienação dos IDSs federados, consequência da premissa de autonomia proposta para os membros federados, a cooperação entre os IDSs membros é viabilizada através da estrutura de comunicação normalizada, proporcionada pelo protocolo BGP Flowspec. Esta normalização facilita o processo de correlação e fusão no AS de destino. A política de correlação, incluindo sistemas de credibilidade atribuída às mensagens BGP que chegam, podem ser definidas independentemente no AS de destino, de acordo com as suas próprias políticas de segurança. Ou seja, as ações correspondentes aos níveis finais de credibilidade, o número mínimo de mensagens para serem combinadas, o intervalo de tempo para correlação, a credibilidade dos ASs que originam as mensagens e o caminho BGP percorrido pelas mensagens (*AS path*) poderão ser utilizados para determinar o nível de credibilidade da informação combinada e suportar decisões de defesa, dependendo da política de segurança de cada sistema autônomo.

Com o objetivo de validar a eficiência do sistema de detecção e a viabilidade técnica da arquitetura, são propostos um modelo analítico e um modelo de experimental. Considerando as características de incerteza e de alienação dos IDSs federados, resultado da premissa de autonomia, propõe-se um processo de fusão de dados para avaliar o nível de credibilidade da hipótese de intrusão, a partir das mensagens recebidas num determinado alvo. Combinando a Regra de Bayes com a Teoria da Evidência de Dempster-Shafer, avalia-se a eficiência do sistema de detecção proposto a partir de métricas de desempenho amplamente conhecidas, definidas a partir da matriz de confusão do sistema. Neste modelo, o valor da verossimilhança da equação de Bayes é obtido utilizando-se a fusão dos dados correlatos que che-

gam ao AS alvo. A probabilidade à priori no mesmo modelo é definida por uma função de distribuição *Beta*, a partir de dados empíricos. A utilização desta distribuição tem como objetivo principal aumentar generalidade dos resultados obtidos empiricamente, modulando-os com as hipóteses de detecção testadas. A viabilidade técnica do sistema de detecção é discutida com base nos resultados de um modelo experimental, construído com o *software* GNS, demonstrando o funcionamento do protoclo FlowSpec dentro do contexto de detecção proposto na federação.

Os resultados obtidos a partir do modelo analítico descrito no parágrafo anterior mostram um aumento de desempenho significativo em comparação com um sistema de detecção normal, de código aberto (SNORT), utilizando as mesmas métricas. O modelo experimental, mostrou que as mensagens FlowSpec BGP manipuladas num ponto da rede (IDSs membros) foram recebidas de forma normalizada e de acordo com o esperado num outro ponto da rede, representando um eventual alvo do ataque.

4.1 Arquitetura Proposta

Como já mencionado, o conceito de detecção distribuída não é novo e o número de propostas da academia neste sentido é relativamente grande. Apesar de terem sido, em sua maioria, inspiradas no sistema de imunização do corpo humano, as propostas apresentadas no Capítulo 3 mostram uma diversidade de abordagens, principalmente com relação à autonomia dos agentes, sua arquitetura distribuída e à estratégia de cooperação. Entretanto, mesmo considerando o grande número e a diversidade de propostas, percebe-se que todas compartilham os mesmos desafios, principalmente relacionados à infraestrutura de comunicação e à heterogeneidade dos agentes distribuídos [109]. No caso particular da infraestrutura de comunicação, a arquitetura deve necessariamente ser leve e de fácil implementação, de modo a encorajar a adesão de novos membros e fomentar o crescimento da federação.

A infraestrutura de comunicação deve ser capaz de interligar todos os agentes distribuídos plenamente, permitindo que todos possam compartilhar informações de forma ubíqua. A proposta de se criar uma estrutura de comunicação específica ou sobreposta esbarra na complexidade de adaptação dos agentes e no desperdício de recursos de banda com cabeçalhos (*overhead*), podendo comprometer o processo de adesão de novos membros.

A heterogeneidade dos agentes é um fator que colabora na amplitude de detecção e a melhor forma de conseguir esta heterogeneidade é através da autonomia dos agentes. Entretanto, esta mesma autonomia também dificulta a comunicação entre os agentes, e por consequência, dificulta o estabelecimento do ambiente cooperativo desejado. Uma forma de minimizar estes problemas é utilizando uma linguagem uniformizada entre os agentes, normalizando a quantidade de informações das men-

sagens, sem contudo comprometer a utilidade dos alarmes.

A arquitetura proposta não dispensa as soluções típicas de detecção de intrusões já estabelecidas e em funcionamento nos ambientes restritos de segurança. Outrossim, tem como principal objetivo aumentar a superfície de detecção ao estabelecer uma linha de defesa avançada para proteção dos provedores de acesso contra os ataques cibernéticos. Trata-se basicamente da criação de uma federação de agentes de detecção, que cooperam entre si, enviando alarmes de possíveis intrusões, utilizando a ubiquidade do protocolo BGP e a estrutura do padrão FlowSpec para normalizar a comunicação. Quanto maior for o número de ASs atravessados pelo fluxo malicioso, maior o número de alarmes enviados para o mesmo fluxo. Quanto maior o número de alarmes correlacionados com um mesmo destino alvo, maior a crença na hipótese de uma intrusão real e melhores serão as possibilidades de proteção no destino. Portanto, pode-se dizer que os 4 pilares de funcionamento da arquitetura proposta são:

- A utilização do BGP como infraestrutura de comunicação comum entre todos os agentes.
- A utilização do tamanho do caminho percorrido pelo fluxo ofensor (número de ASs) para aumentar a possibilidade de detecção e para aumentar a eficiência das medidas de proteção no destino alvo.
- A criação de uma ampla federação de agentes autônomos para ampliar a superfície de detecção através da heterogeneidade dos IDSs membros.
- A utilização da estrutura do padrão FlowSpec para normalizar as mensagens e uniformizar a comunicação entre os agentes federados.

Na arquitetura proposta, cada IDS federado é capaz de detectar um fluxo malicioso que atravessa o seu AS, independentemente do alvo. Ao detectar um fluxo malicioso, o IDS junta as informações básicas do fluxo ofensor e divulga uma mensagem de atualização FlowSpec, incluindo os atributos do fluxo no campo NLRI, conforme explicado na Seção 2.2. Quanto mais heterogênea for a rede de IDSs federados, maior a chance de detectar uma nova intrusão. No destino, deve haver um processo de monitoração de mensagens FlowSpec para correlacionar as mensagens de interesse e disparar ações de proteção, de acordo com os dados recebidos e com a confiabilidade da informação. A confiabilidade da informação combinada é proporcional ao número de mensagens correlacionáveis recebidas no destino. A Figura 4.1 mostra graficamente como funciona o processo.

O processo de monitoração e correlação das mensagens BGP, bem como os limites de confiabilidade para se tomar uma decisão qualquer de proteção no destino,

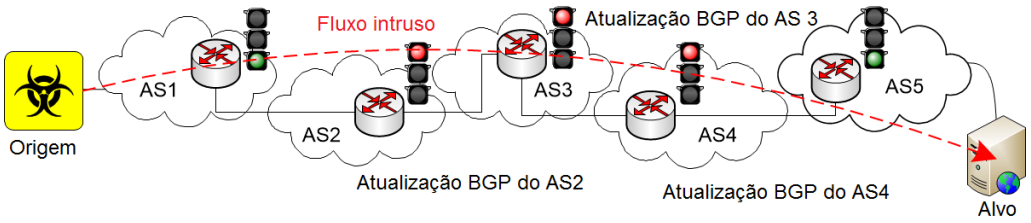


Figura 4.1: Um atacante no AS1 inicia um ataque contra um determinado alvo no AS5. Os IDSs federados nos ASs 2, 3 e 4 detectam o fluxo malicioso e divulgam uma mensagem de atualização BGP FlowSpec para a rede com as informações do fluxo para alertar o AS 5 sobre esta ameaça.

dependem unicamente das premissas de segurança de cada AS federado. Como uma primeira linha de defesa, uma determinada informação de intrusão, dependendo do seu nível de confiabilidade, pode servir de base para um bloqueio imediato ou uma monitoração mais detalhada, concentrada nos dados do fluxo ofensor informado.

As regras que determinarão a confiabilidade sobre cada mensagem BGP que chega ao AS alvo do ataque dependem unicamente das políticas de segurança utilizadas neste AS. Considerando que não há qualquer restrição quanto ao processo de detecção de cada membro federado, pode-se assumir que as mensagens recebidas e correlacionadas no destino apresentam níveis de confiabilidade diferentes, decorrentes de sistemas de detecção diferentes. Por exemplo: um sistema de detecção de intrusões baseado em rede (NIDS) muito simples pode apresentar uma alta taxa de alarmes falso-positivos. Isto certamente contribui negativamente para a composição do nível de crença da informação combinada. Entretanto, mesmo considerando mensagens vindas sistemas federados com baixo desempenho de detecção ou de ASs de menor credibilidade, o número de mensagens combinadas no destino ainda pode garantir níveis de crença suficientes para a adoção de medidas menos intrusivas de proteção no alvo.

O sistema também permite o rastreamento parcial da origem do ataque, uma vez que a mensagem FlowSpec BGP possui o registro dos ASs no caminho, desde o primeiro AS que detectou o fluxo malicioso até o AS alvo. Aliás, este registro dos ASs no caminho das mensagens também pode ser utilizado como um fator de credibilidade, uma vez que múltiplas mensagens recebidas de IDSs no mesmo caminho reforçam a crença na hipótese de uma intrusão verdadeira. A simples presença do número do AS de origem no AS-PATH garante que este AS foi efetivamente atravessado pelo fluxo considerado malicioso, evitando mensagens falsas (*spoofing*). Uma outra estratégia pode ser através de um *ranking* de credibilidade criado no AS de destino, pontuando os ASs que mais acertaram na detecção e intrusões reais ao AS de destino.

4.2 Modelagem

Como mencionado na Seção 2.5, o conceito de fusão de dados pode ser utilizado em diversas áreas, incluindo: diagnósticos médicos, reconhecimento de alvos militares, definição de imagens de satélite, detecção de ameaças de segurança, etc. O próprio ser humano é o principal exemplo de fusão de dados. O cérebro humano toma decisões a partir da fusão de dados provenientes dos nossos sentidos: visão, paladar, olfato e tato.

Particularmente na área de segurança cibernética, as técnicas de fusão de dados podem ser empregadas nos sistemas de detecção de intrusões para aumentar a taxa de alarmes verdadeiro-positivos e reduzir os falso-positivos, melhorando assim o desempenho do sistema. Um exemplo neste sentido é o trabalho apresentado em [110], que propõe alterações na regra de fusão de Dempster-Shafer [35] para melhorar o desempenho de um sistema de detecção distribuído, reduzindo o problema das evidências conflitantes.

Neste trabalho de doutorado, o conceito e as técnicas de fusão de dados são explorados para modelar as métricas e avaliar o desempenho de detecção do sistema distribuído de detecção de intrusões proposto no Capítulo 4. As evidências de intrusão que chegam a um determinado destino são combinadas de acordo com parâmetros de correlação dos fluxos alarmados. O resultado da fusão entra como parâmetro de verossimilhança na equação de Bayes para compor métricas de desempenho do sistema distribuído de detecção de intrusões proposto.

Com o objetivo de aumentar a generalidade do modelo e melhorar sua aderência aos sistemas reais, propõe-se uma modelagem objetiva semi-informativa para a probabilidade à priori, utilizada nas equações Bayesianas das métricas de desempenho. Neste tipo de modelagem, ao invés de usar somente informações subjetivas ou empíricas, o conhecimento inicial é ajustado pela probabilidade à priori calculada objetivamente com base nos dados a favor (α) e contrários (β), à medida que estes forem testados nos cenários hipotéticos de detecção (verossimilhança).

4.2.1 Métricas de Desempenho de Detecção

Uma das formas mais comumente utilizadas para avaliar o desempenho dos sistemas de detecção é analisar a sua capacidade de detectar situações ameaçadoras e diferenciá-las daquilo que é considerado tráfego normal. Esta análise pode ser feita através de métricas específicas, extraídas de uma "matriz de confusão" [111]. A matriz de confusão é uma forma organizada de apresentar os resultados dos testes do sistema de detecção, tendo como entrada um banco de dados rotulado, ou *dataset*.

O banco de dados rotulado contém um conjunto de dados brutos de tráfego, composto por fluxos normais e fluxos intrusos. O termo rotulado se refere à identificação

de cada tipo de fluxo através de marcas especiais para diferenciá-los (metadados). O banco de dados rotulado pode ser construído de forma sintética, contento fluxos intrusos fabricados sinteticamente, ou de forma natural, simplesmente coletando tráfego real durante um determinado intervalo de tempo. A propósito, apesar da grande variedade de bancos de dados rotulados disponíveis na Internet, existem poucos trabalhos estudando os detalhes e as metodologias para se montar novos *datasets*. A proposta apresentada em [112] é motivada pela dificuldade de se encontrar *datasets* reais e oportunos na avaliação de sistemas de detecção de intrusões. Além de relacionar e descrever os principais *datasets* que se encontram disponíveis atualmente, o artigo de Rhuyan *et al.* propõe uma metodologia para gerar *datasets* de tráfego real, imparciais (sem tendência) e suficientemente grandes.

A matriz de confusão M_{ij} mostrada na tabela 4.1 é formada a partir dos testes de desempenho de um determinado sistema de detecção (IDS_i) para cada fluxo dentro do *dataset j*. Importante enfatizar que quanto maior e mais diverso for o *dataset*, mais precisa e aderente será a avaliação das métricas de desempenho de detecção do sistema. Testes de desempenho de detecção mais elaborados envolvem muitas vezes mais de um *dataset*.

Tabela 4.1: Na matriz de confusão M_{ij}, TP_{ij} é o número de intrusões verdadeiras, corretamente detectadas. FN_{ij} se refere ao número de intrusões verdadeiras, que são incorretamente ignoradas. TN_{ij} é o número de fluxos normais corretamente ignorados. FP_{ij} é o número de fluxos normais que foram incorretamente alarmados como intrusões.

IDS_i	**Intrusão**	**Normal**
Alarmou	TP_{ij}	FP_{ij}
Não alarmou	FN_{ij}	TN_{ij}

A partir da matriz de confusão de um determinado *dataset* ou de múltiplos *datasets*, podem ser extraídas diversas métricas importantes para avaliar o desempenho de um sistema de detecção qualquer. Uma métrica bastante utilizada para avaliar a precisão positiva do sistema de detecção é conhecida como valor de predição positiva (PPV). O PPV_i mede a precisão do IDS_i em acertar nos alarmes emitidos que realmente representam ameaça. O cálculo do PPV_i de um determinado IDS_i a partir da matriz de confusão é a razão entre o número de alarmes de intrusão corretamente emitidos, sobre a soma de todos os alarmes emitidos pelo IDS_i, de acordo com a Equação 4.1.

$$PPV_i \triangleq \frac{TP_i}{TP_i + FP_i} \tag{4.1}$$

A taxa de alarmes verdadeiro-positivos (TPR_i) e verdadeiro-negativos (TNR_i) de um determinado IDS_i representam respectivamente a sua sensibilidade em detectar uma intrusão real e a sua especificidade de não alarmar quando não for o caso. Estas métricas podem ser calculadas da seguinte forma:

$$TPR_i \triangleq \frac{TP_i}{TP_i + FN_i} \tag{4.2}$$

$$TNR_i \triangleq \frac{TN_i}{TN_i + FP_i} \tag{4.3}$$

A taxa de alarmes falso-negativos (FNR_i) e falso-positivos (FPR_i) de um determinado IDS_i representam a taxa de perda e de excesso respectivamente. A perda se refere ao número de alarmes que o IDS_i deixou de enviar para intrusões reais. O excesso, por sua vez, se refere aos alarmes enviados desnecessariamente pelo IDS_i. Apesar de ambos representarem problemas de desempenho, uma intrusão real não alarmada tende a ser mais grave do que alarmes enviados desnecessariamente. Desta forma, muitas operações de segurança preferem "pecar pelo excesso", aumentando a sensibilidade do IDS e depois criando filtros para facilitar o pós-processamento do volume maior de alarmes.

$$FNR_i \triangleq \frac{FN_i}{TP_i + FN_i} = 1 - TPR_i \tag{4.4}$$

$$FPR_i \triangleq \frac{FP_i}{TN_i + FP_i} = 1 - TNR_i \tag{4.5}$$

4.2.2 Modelagem Matemática

A modelagem matemática apresentada nesta seção tem por objetivo calcular as métricas de avaliação de desempenho de detecção da plataforma distribuída, proposta na Seção 4.1. Para facilitar o desenvolvimento do modelo, bem como para permitir uma melhor visualização das hipóteses assumidas, é apresentado um esquema gráfico simplificado da arquitetura do sistema distribuído de detecção na Figura 4.2.

A Figura 4.2 mostra a hipotética ocorrência de uma determinada intrusão contra um determinado alvo na Internet num determinado instante de tempo t.

À medida que detecta a passagem do fluxo suspeito pelos seus respectivos ASs (*Autonomous System*), cada IDS_i federado divulga uma mensagem de atualização BGP (*Border Gateway Protocol*) FlowSpec U_i, com os atributos básicos do fluxo. Quando chegam ao AS alvo no intervalo de tempo $[t; t+\Delta t]$, as mensagens FlowSpec BGP (U_i) são correlacionadas com relação aos atributos que carregam. As massas de crença das mensagens correlacionadas são combinadas matematicamente (*data fusion*) para gerar a massa de crença total sobre a informação binária de intrusão.

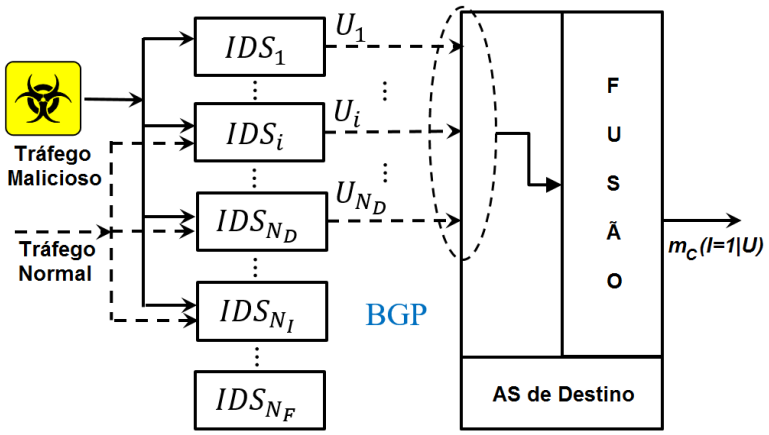


Figura 4.2: Número N_D evidências de intrusão paralelas sendo fundidas no AS de destino de acordo com um atributo de fluxo comum.

Esta informação pode ser útil na seleção das medidas de defesa, que podem ser eventualmente adotadas no AS alvo para se protegerem de um possível ataque cibernético. Quanto maior for a massa de crença combinada, mais séria e ameaçadora a hipótese de uma intrusão real em curso contra aquele AS.

A arquitetura é composta por um total de N_F IDSs federados, que são todos independentes entre si e com a mesma relevância. O tráfego normal e o tráfego malicioso passam por um número N_I ($N_I \leq N_F$) de IDSs até alcançarem seu destino. Como são autônomos e independentes, cada IDS ao longo do caminho do fluxo intruso pode ou não detectar como uma ameaça, dependendo apenas do seu próprio desempenho de detecção individual (sensibilidade e especificidade). Um determinado IDS_i que detecta o fluxo malicioso, junta as características básicas deste fluxo malicioso e divulga uma mensagem de atualização FlowSpec na rede BGP, incluindo-as no corpo da mensagem de divulgação. Um vez chegando na rede BGP do AS alvo, um total de N_D ($N_D \leq N_I$) mensagens são correlacionadas, de acordo com as características comuns do fluxo e combinadas para avaliar a credibilidade da hipótese final de intrusão contra o alvo.

A relação N_I/N_F depende basicamente do quão distante em número de ASs a origem do ataque está do seu AS de destino e tamanho da superfície do ataque. Os trabalhos apresentados em [113] e [114] discorrem sobre métodos para calcular o tamanho médio dos fluxos na Internet de acordo com o número de ASs. O tamanho do ataque está relacionado com o número de fontes que o atacante utiliza no seu ataque, quanto mais fontes utilizadas, mais amplo é o ataque. O trabalho apresentado em [115] analisa o tamanho dos ataques de DDoS como um parâmetro para determinar o seu grau de risco. Os autores apresentam uma tabela contendo o tamanho de vários tipos de ataques DDoS conhecidos, que podem variar desde algumas dezenas de *bots*, no caso do ataque *Ddoser*, até algumas centenas de milhares, no caso do *Dirtjumper*.

Assume-se a variável aleatória de Bernoulli I, para indicar a ocorrência ($I = 1$) ou não ($I = 0$) de uma intrusão real num dado instante de tempo. Assume-se também U_i ($i = 1, 2, \cdots, N_F$), como sendo um conjunto finito de variáveis aleatórias de Bernoulli, mutuamente independentes e identicamente distribuídas, indicando a detecção ($U_i = 1$) ou não ($U_i = 0$) de uma intrusão real ou falsa, por um determinado IDS_i ao longo do seu percurso na Internet.

A probabilidade de ocorrer uma intrusão neste sistema é dada por $\Pr(I = 1) = 1 - \Pr(I = 0)$. Da mesma forma, a probabilidade de um determinado IDS_i detectar uma intrusão (real ou falsa) é dada por $\Pr(U_i = 1) = 1 - \Pr(U_i = 0)$; para cada $i = 1, 2, \cdots, N_F$.

Assumindo que os valores médios da matriz de confusão de todos os N_F IDSs federados foram obtidos com o treinamento de um conjunto de dados suficientemente grande e diversificado, pode-se aproximar as métricas de desempenho de detecção da Seção 4.2.1 probabilisticamente (Pr), conforme Tabela 4.2.

Tabela 4.2: Representação probabilística das métricas de desempenho de detecção descritas na Seção 4.2.1, considerando que foram obtidas com o treinamento utilizando um conjunto de dados suficientemente grande e diverso.

Métrica	**Fórmula**	**Representação**
PPV_i	$\frac{TP_i}{TP_i+FP_i}$	$\sim \Pr(I = 1 \mid U_i = 1)$
TPR_i	$\frac{TP_i}{TP_i+FN_i}$	$\sim \Pr(U_i = 1 \mid I = 1)$
TNR_i	$\frac{TN_i}{TN_i+FP_i}$	$\sim \Pr(U_i = 0 \mid I = 0)$
FNR_i	$\frac{FN_i}{TP_i+FN_i}$	$\sim \Pr(U_i = 0 \mid I = 1)$
FPR_i	$\frac{FP_i}{TN_i+FP_i}$	$\sim \Pr(U_i = 1 \mid I = 0)$

Valor de Predição Positiva - PPV

Conforme descrito na Seção 4.2.1, a métrica PPV, também conhecida como precisão positiva, mede a capacidade de um IDS em alarmar o que realmente representa uma ameaça. Isto é, uma vez emitindo um alarme de intrusão para o AS de destino, o o PPV mede a capacidade do IDS em acertar nesta indicação de ameaça.

O alarme de intrusão do IDS é recebido no AS de destino como uma evidência de intrusão dentro de um quadro de discernimento. O quadro de discernimento Ω pode ser definido então como o conjunto exaustivo de evidências mutuamente exclusivas, que podem chegar ao AS de destino para serem combinadas dentro do processo de fusão de dados, conforme Figura 4.2.

Assim, considerando que chegam somente as $N_D \leq N_I \leq N_F$ mensagens indicando uma intrusão, que podem ser verdadeiras ou falsas, teremos $\Omega = \{I = 1|U = 1, I = 0|U = 1\}$. Formado a partir do quadro de discernimento, o conjunto potencia contém todos os subconjuntos possíveis do conjunto Ω, sendo então $2^{|\Omega|} = \{\{I = 1|U = 1\}, \{I = 0|U = 1\}, \{I = 1|U = 1\} \cup \{I = 0|U = 1\}, \{\emptyset\}\}$.

O problema da regra de combinação de Dempster-Shafer relacionado com o conflito das fontes de evidências não existe neste modelo, uma vez que só serão fundidas mensagens evidenciando e concordando correta ou incorretamente com a hipótese de intrusão ($U_i = 1$). No modelo proposto, o interesse maior é sobre o número de mensagens redundantes que suportam a hipótese de intrusão. Quanto maior for este número, maior será a massa de crença combinada da hipótese de intrusão.

A massa de crença individual $m_i(I = 1|U_i = 1)$ representa a parte da crença oferecida pelo IDS_i, que suporta a hipótese de intrusão no subconjunto $A_s = \{\{I = 1|U = 1\}\} \cap \{\{I = 1|U = 0\} \cup \{I = 0|U = 1\}\}$. Assim, a massa de crença fundida do sistema como um todo, combinando todas as N_D mensagens FlowSpec recebidas no alvo atacado, $m_C(I = 1|U = 1)$ pode ser obtida através da Equação 4.6.

$$m_C(I = 1|U = 1) = \frac{\sum_{\cap A_s = (I=1|U=1)} \prod_{1 \leq i \leq N_D} m_i(A_s)}{1 - K} \quad ;$$
$$\text{onde:} \quad K_X = \sum_{\cap A_s = \emptyset} \prod_{1 \leq i \leq N_D} m_i(A_s) \quad (4.6)$$

Levando em consideração que não há conflitos nas mensagens que são combinadas, $K_X = 0$ na Equação 4.6.

Da mesma forma apresentada em [110], a massa de crença individual de cada fonte de evidência $m_i(I = 1|U_i = 1)$ pode ser calculada por lógica subjetiva binomial proposta em [116].

$$m_i(I = 1|U_i = 1) = \frac{TP_i}{TP_i + FP_i + 2} \sim PPV_i \quad (4.7)$$

Para uma base de dados de treinamento suficientemente grande, pode-se assumir $TP_i + FP_i \gg 2$. Desta forma, a massa de crença individual de cada IDS_i $m_i(I = 1|U_i = 1)$, para cada mensagem U_i $(i = 1, 2, \cdots, N_D)$ que chega para ser combinada no AS alvo, pode ser representada pelo seu Valor de Predição Positiva PPV_i, conforme Equação 4.1.

A crença final na hipótese de intrusão, dado que a plataforma DIDS a detectou, é, na verdade, o valor de predição positiva do sistema distribuído de detecção como um todo (PPV_{DIDS}) e pode ser representada pela combinação de todas N_D as mensagens que chegaram no AS alvo. Como não há conflitos nas evidências dentro do quadro

de discernimento no AS de destino, $K = 0$ no denominador da Equação 4.6. O valor de PPV_{DIDS} pode ser aproximado através da Equação 4.8. Os detalhes da evolução da Equação 4.6, até se chegar na Equação 4.8 estão no Apêndice A.

$$PPV_{DIDS} = m_C(I = 1|U = 1) = 1 - \prod_{i=1}^{N_D}(1 - PPV_i) \tag{4.8}$$

O cálculo do valor de predição positiva da plataforma DIDS (PPV_{DIDS}) ainda pode aproximado na Equação 4.9, assumindo-se um valor médio de predição positiva PPV_{av} para todos os N_F IDSs federados, considerando todos os J *datasets* utilizados no treinamento.

$$PPV_{DIDS} = m_C(I = 1|U = 1) = 1 - (1 - PPV_{av})^{N_D} \tag{4.9}$$

PPV_{av} corresponde ao valor médio a precisão positiva de todos os N_F IDSs federados, que pode ser calculado a partir da Equação 4.1, considerando todas as J matrizes de confusão (Tabela 4.1) extraídas dos J *datasets* utilizados no treinamento.

$$PPV_{av} = \frac{1}{JN_F}\sum_{i=1}^{N_F}\sum_{j=1}^{J}\frac{TP_{ij}}{TP_{ij} + FP_{ij}} \tag{4.10}$$

Apesar de provir originalmente do cálculo da massa de crença combinada dos IDSs que detectaram um fluxo considerado malicioso $m_C(I = 1|U = 1)$, a Equação 4.8 representa realmente a probabilidade de pelo menor um dos N_D IDSs ter detectado corretamente um fluxo suspeito atravessando seu AS como uma intrusão real, podendo ser reescrita na Equação 4.13.

$$PPV_{DIDS} \sim \Pr(I = 1|U = 1) = 1 - P(X = 0) \tag{4.11}$$

Na Equação 4.11, $X \leq N_D$ é uma variável aleatória que indica o número de IDSs que detectaram e alarmaram uma intrusão real. Assim, a probabilidade de x IDSs detectarem e alarmarem uma intrusão real pode ser escrita como:

$$\Pr(X = x) = Bin(N_D, PPV_{av}) = \binom{N_D}{x} {PPV_{av}}^x (1 - PPV_{av})^{(N_D - x)} \tag{4.12}$$

Como $X \leq N_D$ representa o número de IDSs acertaram na detecção de uma intrusão real, $\Pr(X = 0)$ representa então a probabilidade de nenhum dos N_D IDSs que detectaram a falha tenha acertado em alarmar uma intrusão real. Assim, a Equação 4.11 pode ser reescrita como:

$$PPV_{DIDS} = 1 - (1 - PPV_{av})^{N_D} \tag{4.13}$$

Modelagem da Probabilidade à Posteriori

A Taxa de Verdadeiro-Positivos da plataforma DIDS (TPR) mede sua sensibilidade na detecção correta de intrusões. A partir de sua representação probabilística, mostrada na Tabela 4.2, recorre-se à regra de probabilidades condicionais de Bayes para desenvolver a base de cálculo.

$$TPR_{DIDS} \sim \Pr(U = 1|I = 1) = \frac{\Pr(I = 1|U = 1) \times \Pr(U = 1)}{\Pr(I = 1)} \tag{4.14}$$

Dado o caráter mutuamente exclusivo da variável aleatória I, o denominador $\Pr(I = 1)$ na Equação 4.15 pode ser decomposto na soma ponderada $\Pr(I = 1) = \Pr(I = 1|U = 1)\Pr(U = 1) + \Pr(I = 1|U = 0)\Pr(U = 0)$. Desta forma, $\Pr(I = 1)$ funciona apenas como um fator de normalização na Equação 4.14, podendo ser omitido se forem considerados limites superiores e inferiores.

$$TPR_{DIDS} \sim \Pr(U = 1|I = 1) \geq \Pr(I = 1|U = 1) \times \Pr(U = 1) \tag{4.15}$$

Substituindo-se os resultados da Equação 4.13 na Equação 4.15.

$$TPR_{DIDS} \geq \left[1 - (1 - PPV_{av})^{N_D}\right] \times \Pr(U = 1) \tag{4.16}$$

A taxa de alarmes falso-negativos (FNR) do DIDS (FNR_{DIDS}) pode ser obtida da identidade ($FNR_{DIDS} = 1 - TPR_{DIDS}$).

$$FNR_{DIDS} \leq 1 - \left[1 - (1 - PPV_{av})^{N_D} \times \Pr(U = 1)\right] \tag{4.17}$$

A taxa de alarmes falso-positivos (FPR) pode ser obtida a partir do nível de descrença em todas as N_D mensagens que foram recebidas no AS alvo. Isto é, considerando o cálculo da massa de crença combinada $m_C(I = 1|U = 1)$ mostrado na Equação 4.8, o FPR_{DIDS} pode ser calculado segundo a Equação 4.18 abaixo.

$$FPR_{DIDS} \leq (1 - PPV_{DIDS}) \times \Pr(U = 1) = (1 - PPV_{av})^{N_D} \times \Pr(U = 1) \tag{4.18}$$

Finalmente, podemos obter a taxa de alarmes verdadeiro-negativos (TNR) através da igualdade ($TNR_{DIDS} = 1 - FPR_{DIDS}$).

$$TNR_{DIDS} \geq 1 - \left[(1 - PPV_{av})^{N_D} \times \Pr(U = 1)\right] \tag{4.19}$$

Modelagem da Probabilidade à Priori

A probabilidade à priori geral da plataforma DIDS $\Pr(U = 1)$, parte das Equações das métricas de desempenho (4.16, 4.17, 4.18 e 4.19), descreve a tendência do sistema de detecção em alarmar positivamente uma intrusão, mesmo que incorretamente (falso alarme). Esta probabilidade pode ser obtida buscando-se informações prévias acerca do desempenho de detecção de cada um dos N_F IDSs da federação.

Conforme descrito na Seção 2.3.1, a probabilidade à priori na Equação de Bayes também representa o nível de crença no dado que está sendo analisado. Este conhecimento prévio, quando disponível, deve considerar também o contexto no qual está inserida a análise de Bayes [117]. Conforme já mencionado na Seção 2.3.1 A utilização de uma probabilidade à priori muito informativa pode levar a resultados incorretos ou sem sentido.

Uma forma de se obter este conhecimento prévio, seria através da matriz de confusão j de cada IDS_i entre todos os N_F IDSs federados. Ou seja, considerando o *dataset* j suficientemente grande e diverso, é possível assumir que:

$$\Pr(U_i = 1) \approx PR_i = \frac{TP + FP}{TP + FP + TN + FN} \tag{4.20}$$

Se forem considerados múltiplos *datasets*, a taxa média de positivos dos N_F IDSs federados pode ser calculada como:

$$PR_{av} = \frac{1}{JN_F} \sum_{i=1}^{N_F} \sum_{j=1}^{J} PR_{ij} \tag{4.21}$$

Onde **J** é o número total de *datasets* utilizados para obter a matriz de confusão de todos os N_F IDSs membros da federação e PR_{av}, é a taxa de detecção positiva média do sistema. Quando maior for o número de *datasets* diferentes, mais vale a similaridade da probabilidade à priori com a taxa de positivos.

$$p = PR_{av} \rightarrow Pr(U = 1) \tag{4.22}$$

Assim, no caso do sistema distribuído de detecção proposto, considerando que apenas $N_I \leq N_F$ IDSs foram atravessados pelo fluxo suspeito, o valor integral de $\Pr(U = 1)$ da plataforma DIDS poderia ser calculado como a probabilidade de ao menos um dos N_I IDS atravessados pelo fluxo suspeito alarmar positivamente uma intrusão, ou seja:

$$\Pr(U = 1) \approx 1 - (1 - PR_{av})^{N_I} \tag{4.23}$$

Como pode ser notado, o valor de $\Pr(U = 1)$ da Equação 4.23 depende apenas do número de IDSs atravessados pelo fluxo suspeito N_I, permanecendo constante no

cálculo das métricas de desempenho mostradas nas Equações 4.16, 4.17, 4.18 e 4.19, caso este número não se altere no cenário do ataque. Ou seja, como uma constante, a probabilidade à priori exerce um mesmo fator de ponderação, independente das hipóteses de detecção testadas.

A suposição de existir um *dataset* suficientemente grande e diverso que suporte a Equação 4.22, pode ser considerada frágil, em face da diversidade e da dinâmica no surgimento de novos ataques. Além disto, a opção de treinar todos os N_F IDSs em múltiplos *datasets* se configura como inviável, em função do cenário globalmente distribuído e independente dos agentes federados e das diferenças de projeto entre os próprios *datasets*, que são muitas vezes concebidos para fins específicos [118]. De fato, se for considerado o *dataset* [1], tem-se $\mathrm{PR}_i = 0,271$, enquanto que o *dataset* [2] produz um $\mathrm{PR}_i = 0,373$, para um mesmo tipo de IDS_i (Snort) treinado.

Uma outra forma de se modelar a probabilidade à priori é utilizar o próprio dado de saída do sistema de detecção para ajustar o valor do conhecimento prévio, à medida que as hipóteses de detecção são testadas. Neste caso, é como se o modelo fosse "aprendendo" com as evidências, dentro do seu próprio perfil de utilização. Esse tipo de probabilidade à priori, considerado mais genérico, é conhecido como modelo objetivo semi-informativo. A utilização do modelo objetivo semi-informativo para estimar a probabilidade à priori do sistema distribuído de detecção proposto se adéqua à arquitetura do sistema proposto na Seção 4.1, pois varia em função das hipóteses de detecção, valorizando-as no resultado à posteriori.

No modelo objetivo, a probabilidade de um determinado IDS_i da federação detectar correta ou incorretamente uma mesma intrusão, é tratada como uma variável aleatória $\Theta = \Pr(U_i = 1)$, com $\Pr(\Theta = \theta) \in [0, 1]$. Neste modelo, a probabilidade de N_D IDSs terem detectado positivamente uma mesma intrusão, dentre os N_I IDSs atravessados pelo fluxo malicioso pode ser escrita como uma equação binomial:

$$\Pr(N_D|N_I, \theta) = \binom{N_I}{N_D} \theta^{N_D} (1-\theta)^{N_I - N_D} \tag{4.24}$$

Embora sob uma perspectiva clássica frequentista pode se considerado inaceitável representar uma distribuição de probabilidades como um parâmetro, na perspectiva Bayesiana esta representação é perfeitamente consistente, desde que a probabilidade à priori possa se adequar ao modelo. Na abordagem Bayesiana proposta, a probabilidade à priori entra como parâmetro do modelo, enquanto que os dados observados das hipóteses de detecção entram no modelo como valores fixos, representando os dados da verossimilhança. Entretanto, diferentemente das abordagens de máximo à posteriori ou máxima verossimilhança [119], cujo principal objetivo é maximizar a verossimilhança ou a probabilidade à posteriori a partir do parâmetro, propõe-se uma probabilidade à priori balanceada e semi-informativa como uma variável alea-

tória, combinando dados históricos com dados correntes das hipóteses de detecção. Em outras palavras, ao invés de utilizar um valor fixo $p = PR_{av}$ da Equação 4.21, faz-se $p \sim \Theta$ com $F_\Theta(\theta) \in [0, 1]$. Assim, a taxa de alarmes verdadeiro-positivos de um determinado IDS_i, TPR_i, pode ser reescrita como:

$$TPR_i = \Pr(U_i = 1|I = 1) = \frac{\Pr(I = 1|U_i = 1) \times f_\Theta(\theta)}{\Pr(I = 1)} \tag{4.25}$$

O formato binomial da verossimilhança na

De acordo com a análise apresentada em [120] e considerando o formato binomial da verossimilhança, a probabilidade à priori na Equação 4.24 pode ser representada pela função Beta (conjugada), com parâmetros α e β. Neste caso a variável aleatória Θ é da forma:

$$\Theta \sim \frac{1}{B(\alpha, \beta)} \theta^{\alpha-1}(1 - \theta)^{\beta-1}$$

Onde:

$$B(\alpha, \beta) = \int_0^1 \theta^{(\alpha-1)}(1 - \theta)^{(\beta-1)} d\theta$$

Na equação de Bayes proposta em [121], a probabilidade a posteriori $\Pr(\theta|N_D, N_I)$ pode ser calculada a partir da Equação 4.26 abaixo.

$$\Pr(\theta|N_D, N_I) = \frac{\Pr(N_D, N_I|\theta) f_\Theta(\theta)}{\Pr(N_D, N_I)} \tag{4.26}$$

Onde:

$$f_\Theta(\theta) = \frac{dF_\Theta(\theta)}{d\theta} = \frac{1}{B(\alpha, \beta)} \theta^{\alpha-1}(1 - \theta)^{\beta-1} \tag{4.27}$$

Tendo a representação da própria função Beta em função da função Gama como

$$B(\alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)}$$

Juntando as Equações 4.24, 4.26 e 4.27, chega-se a

$$\Pr(\theta|N_D, N_I) = \frac{\binom{N_I}{N_D} \theta^{N_D}(1 - \theta)^{N_I - N_D} \theta^{\alpha-1}(1 - \theta)^{\beta-1}}{B(\alpha, \beta) \int_0^1 \Pr(N_D, N_I|\theta) f_\Theta(\theta) d\theta} \tag{4.28}$$

Desenvolvendo o cálculo da integral no denominador da Equação 4.28:

$$\int_0^1 \Pr(N_D, N_I|\theta) f_\Theta(\theta) d\theta = \int_0^1 \frac{1}{B(\alpha, \beta)} \binom{N_I}{N_D} \theta^{N_D+\alpha-1}(1 - \theta)^{N_I - N_D + \beta - 1} d\theta$$

$$= \frac{1}{B(\alpha,\beta)}\binom{N_I}{N_D}B(N_D+\alpha, N_I-N_D+\beta)$$

Voltando com este último resultado na Equação 4.28

$$\Pr(\theta|N_D,N_I) = \frac{\theta^{N_D+\alpha-1}(1-\theta)^{N_I-N_D+\beta-1}}{B(N_D+\alpha, N_I-N_D+\beta)} \tag{4.29}$$

Calculando-se a marginal em relação ao valor de N_D, tem-se

$$\begin{aligned}\Pr(\theta|N_I) =& \sum_{x=0}^{N_I}\Pr(\theta|N_D=x,N_I)\Pr(N_D=x|N_I)\\ =& \frac{\theta^{x+\alpha-1}(1-\theta)^{N_I-x+\beta-1}}{B(x+\alpha,N_I-x+\beta)}\Pr(N_D=x|N_I)\end{aligned} \tag{4.30}$$

Considerando um total de N_I IDSs atravessados pelo fluxo malicioso, é possível obter a probabilidade de detecção p como a probabilidade à priori na Equação 4.25 calculando-se o valor esperado na Equação 4.30.

$$\begin{aligned}&\mathbb{E}_{\Pr(\theta|N_I)}(\Theta) = \int_0^1 \theta\Pr(\theta|N_I)d\theta\\ &= \int\sum_{x=0}^{N_I}\Pr(N_D=x|N_I)\frac{\theta^{x+\alpha}(1-\theta)^{N_I-x+\beta-1}}{B(x+\alpha,N_I-x+\beta)}\\ &= \sum_{x=0}^{N_I}\Pr(N_D=x|N_I)\frac{\int_0^1\theta^{x+\alpha}(1-\theta)^{N_I-x+\beta-1}d\theta}{B(x+\alpha,N_I-x+\beta)}\\ &= \sum_{x=0}^{N_I}\Pr(N_D=x|N_I)\frac{B(x+\alpha+1,N_I-x+\beta)}{B(x+\alpha,N_I-x+\beta)}\\ &= \sum_{x=0}^{N_I}\Pr(N_D=x|N_I)\frac{x+\alpha}{N_I+\alpha+\beta}\\ &= \frac{1}{N_I+\alpha+\beta}\sum_{x=0}^{N_I}\Pr(N_D=x|N_I)(x+\alpha)\\ &= \frac{1}{N_I+\alpha+\beta}\left(\alpha+\sum_{x=0}^{N_I}\Pr(N_D=x|N_I)x\right)\\ &= \frac{\alpha+E(N_D)}{N_I+\alpha+\beta}\end{aligned} \tag{4.31}$$

O valor $\mathbb{E}(N_D)$ pode ser obtido como a fração média dos N_I IDSs atravessados por um fluxo considerado malicioso e que o detectaram como uma intrusão:

$$\mathbb{E}(N_D) = \sum_{i=1}^{N_I} PR_i = N_I \times PR_{av} \tag{4.32}$$

Lembrando que o valor de PR_{av} é a taxa de alarmes positivos detectados, calcu-

lado a partir das matrizes de confusão [1] e [2], é possível estimar o valor a probabilidade de detecção positiva de um determinado IDS$_i$ $\Pr(U_i = 1)$ como o valor médio probabilidade a posteriori, considerando agora o número de IDSs atravessados pelo fluxo malicioso e o valor médio do número de IDSs que detectaram uma intrusão $\mathbb{E}(N_D)$. Ou seja:

$$p = \Pr(U_i = 1) \sim \mathbb{E}_{\Pr(\Theta|N_I)}(\Theta) = \frac{\alpha + PR_{av}N_I}{N_I + \alpha + \beta} \tag{4.33}$$

Para calcular o valor da probabilidade à priori $\Pr(U = 1)$ da plataforma DIDS, considera-se novamente que basta um dos N_I IDSs detectar uma intrusão. Assim, a probabilidade à priori integral da plataforma DIDS $\Pr(U = 1)$ pode ser calculada como:

$$P(U = 1) = 1 - (1 - p)^{N_I} \tag{4.34}$$

Voltando à Equação 4.18 como ponto de partida, pode-se reescrevê-la da seguinte forma:

$$FPR_{DIDS} \sim (U = 1|I = 0) \leq (1 - PPV_{av})^{N_D} \times (1 - (1 - p)^{N_I}) \tag{4.35}$$

O termo $(1 - PPV_{av})^{N_D}$ da Equação 4.35 também decorre da distribuição binomial com $P(X = 0)$ na Equação 4.12, que representa a probabilidade de todos os N_D alarmes enviados pelos IDSs que detectaram intrusão serem falsos.

O restante das métricas de desempenho decorrem seguindo o mesmo raciocínio das Equações 4.16, 4.17 e 4.19, isto é:

$$TPR_{DIDS} \sim (U = 1|I = 0) \geq 1 - (1 - PPV_{av})^{N_D} \times (1 - (1 - p)^{N_I}) \tag{4.36}$$

$$FNR_{DIDS} \sim (U = 1|I = 0) \leq 1 - [1 - (1 - PPV_{av})^{N_D} \times (1 - (1 - p)^{N_I})] \tag{4.37}$$

$$TNR_{DIDS} \sim (U = 1|I = 0) \geq 1 - [(1 - PPV_{av})^{N_D} \times (1 - (1 - p)^{N_I})] \tag{4.38}$$

4.3 Resultados Obtidos

Esta seção apresenta os resultados dos modelos matemáticos propostos na Seção 4.2.2 para avaliar as métricas de detecção propostas na Seção 4.2.1. Os resultados numéricos dos modelos são confrontados com os resultados obtidos das matrizes de confusão do IDS Snort [1] e [2].

As métricas de detecção da plataforma DIDS são analisadas de acordo com a precisão média dos IDSs membros (PPV_{av}) e considerando diferentes hipóteses para ($N_D \times N_I$).

4.3.1 Matrizes de Confusão

A matriz de confusão é uma forma organizada de apresentar os resultados do treinamento de sistemas de detecção utilizando *datasets* rotulados. Com as informações da matriz de confusão é possível calcular métricas para avaliar o desempenho de um sistema de detecção.

Na Tabela 4.3 é apresentada a matriz de confusão gerada no trabalho de [1]. Esta matriz foi gerada treinando-se o Snort [122] com a base de dados UNSW-NB15 [123]. O treinamento contabilizou um total de 2.540.047 registros, sendo que aproximadamente 12% dos registros correspondiam a intrusões reais.

Tabela 4.3: Matriz de confusão binária do Snort obtida em [1].

2.540.047		Valor Previsto	
		Ataque	Normal
Banco de Dados	Ataque	7.808	313.475
	Normal	680.365	1.538.399

Na Tabela 4.4 é apresentada a matriz de confusão gerada no trabalho de [2]. Esta matriz foi gerada treinando-se o Snort [122] com a base de dados KDD-99. O treinamento contabilizou um total de 2.073.624 registros.

Tabela 4.4: Matriz de confusão binária do Snort obtida em [2].

2.073.624		Valor Previsto	
		Ataque	Normal
Banco de Dados	Ataque	702.111	70.910
	Normal	72.516	1.228.087

As métricas de desempenho calculadas a partir das Tabelas 4.3 e 4.4 estão dispostas na Tabela 4.5 abaixo:

Tabela 4.5: Métricas de desempenho de detecção calculadas a partir das matrizes de confusão geradas em [1] e [2].

Métrica	**Fórmula**	**Valor** [1]	**Valor** [2]	**Média**
PPV	$\frac{TP}{TP+FP}$	$0,011$	$0,906$	$0,459$
TPR	$\frac{TP}{TP+FN}$	$0,024$	$0,908$	$0,466$
TNR	$\frac{TN}{TN+FP}$	$0,693$	$0,944$	$0,819$
FNR	$\frac{FN}{TP+FN}$	$0,975$	$0,091$	$0,533$
FPR	$\frac{FP}{TN+FP}$	$0,306$	$0,056$	$0,181$
PR	$\frac{TP+FP}{TP+TN+FP+FN}$	$0,271$	$0,373$	$0,322$

Comparando-se os valores das métricas obtidas das duas matrizes [1] e [2], nota-se que, apesar de treinar no mesmo IDS (Snort), os resultados são bastante diferentes. Isto mostra que a análise de desempenho do IDS pela matriz de confusão depende muito do *dataset* utilizado para treiná-lo.

4.3.2 Métricas Modeladas

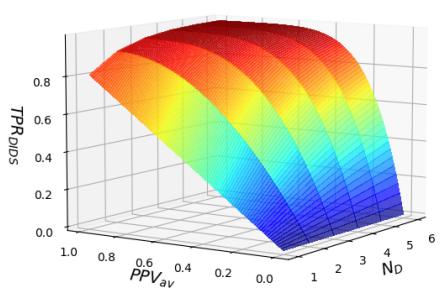
Esta Seção apresenta os resultados numéricos das métricas de desempenho de detecção modeladas de acordo com as Equações 4.35, 4.36, 4.37 e 4.38.

Os gráficos apresentados na Figura 4.3 mostram o comportamento 3D de cada uma das métricas de desempenho da plataforma DIDS, em função do valor de predição positivo médio (PPV_{av}) dos IDSs que compõem a federação e do número de IDSs que detectaram a intrusão e enviaram o alarme N_D.

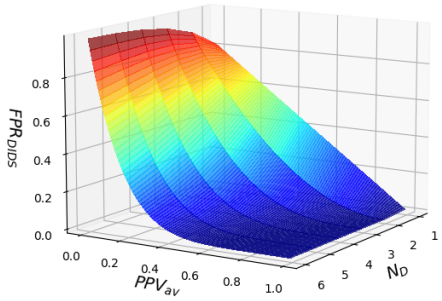
A Figura 4.4 mostra o comportamento de duas métricas de desempenho de detecção do sistema (TPR e FPR) em função do valor de predição positivo médio dos IDSs federados (PPV_{av}), de acordo com o número de IDSs que detectam a intrusão ($N_D = [1-6]$). Neste caso, o número de IDSs que foram atravessados pela intrusão foi fixado $N_I = 6$, conforme [113, 114].

A Figura 4.5 mostra o comportamento da diferença entre métrica taxa de verdadeiro-positivos e a taxa de falso-positivos, em função da probabilidade de detectção positiva média (PR_{av}). Este comportamento, que é uma aproximação do conhecido ROC (*Receiver Operating Characteristic*) mede na verdade o quanto que o TPR se afasta do FPR, em função de ajustes nos limiares de sensibilidade e especificidade dos detectores.

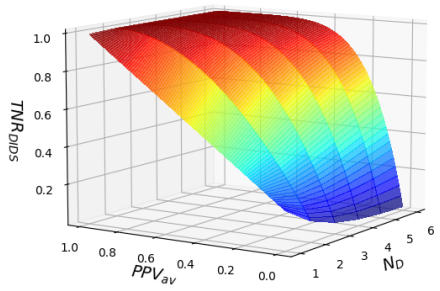
A Figura 4.6 mostra o comportamento 3D das métricas TPR_{DIDS} e FPR_{DIDS} em função do número de N_D e N_I, respeitando-se a restrição $N_D \leq N_I$. Neste caso,



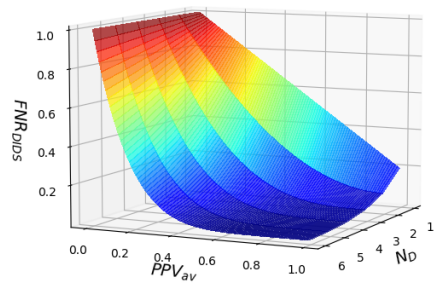
(a) TPR_{DIDS} da Equação 4.36



(b) TPR_{DIDS} da Equação 4.37



(c) TNR_{DIDS} da Equação 4.38



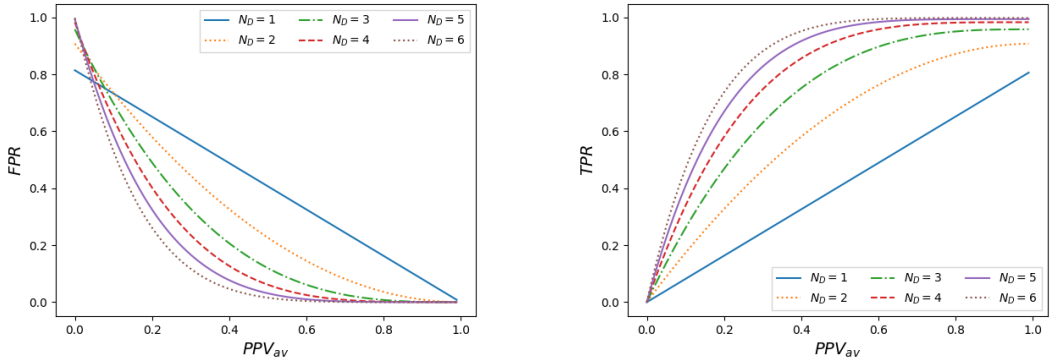
(d) TNR_{DIDS} da Equação 4.35

Figura 4.3: Avaliação 3D das Métricas de desempenho do sistema distribuídos de detecção em função do número de IDSs que detectam a intrusão (N_D) e do valor de predição positiva médio (PPV_{av}). Fixando-se o valor de $\text{PR}_{av} = 0,323$ (média aritmética dos *datasets*) e o valor do número de IDSs atravessados pelo fluxo intruso $N_I = 6$ conforme [113, 114].

os valores de PPV_{av} e PR_{av} foram fixados em $(0,459)$ e $(0,322)$, respectivamente.

4.4 Análise dos Resultados

Analisando os 4 gráficos da Figura 4.3, observa-se que os melhores resultados, altos para TPR/TNR e baixos para FNR/FPR, são obtidos quando os valores de PPV_{av} e N_D convergem juntos para seus valores máximos $\text{PPV}_{av} \to 1$ e $N_D \to 6$ respectivamente. Considerando o ambiente de detecção distribuído estudado, este comportamento indica que o desempenho do sistema tende a degradar quando o valor de predição positiva médio dos IDSs (PPV_{av}) federados é baixo. De fato, se for considerado um valor de $\text{PPV}_{av} \leq 0,33$ o sistema proposto apresenta desempenho de detecção equivalente com sistemas monolíticos típicos, como o sistema referenciado em [2], mesmo quando $N_D = N_I$. Esta conclusão indica que os IDSs membros



(a) FPR_{DIDS} da Equação 4.35; $N_D = [1 - 6]$. (b) TPR_{DIDS} da Equação 4.36; $N_D = [1 - 6]$.

Figura 4.4: Avaliação das métricas FPR_{DIDS} e TPR_{DIDS} em função do valor de predição positiva médio (PPV_{av}), para cada valor de N_D. Fixando-se o valor de $\text{PR}_{av} = 0,322$ (média aritmética dos *datasets*) e o valor do número de IDSs atravessados pelo fluxo intruso $N_I = 6$ conforme [113, 114].

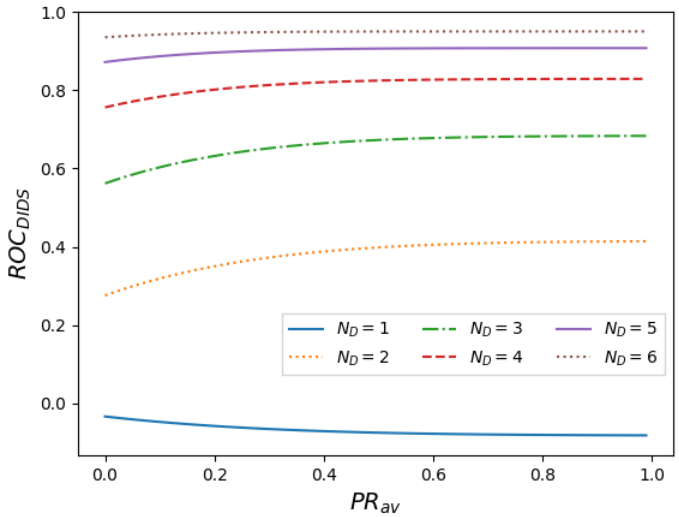
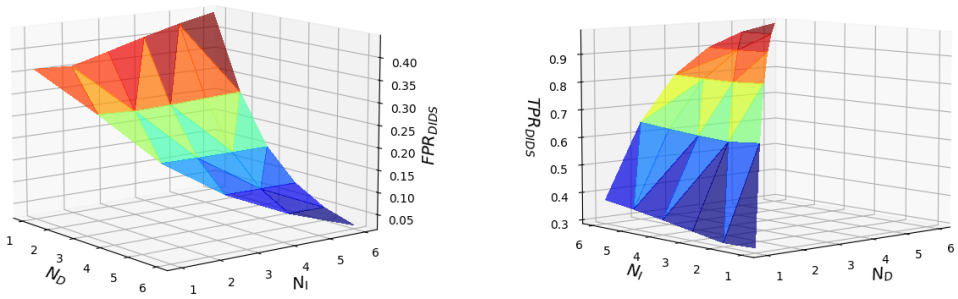


Figura 4.5: Capacidade de alarmar positivamente, minimizando o número de alarmes falsos. $ROC_{DIDS} = TPR_{DIDS} - FPR_{DIDS}$.

da federação precisam funcionar com um mínimo de precisão positiva para viabilizar os ganhos da proposta distribuída cooperativa.

Na Figura 4.4 é possível observar o comportamento da taxa de verdadeiro-positivos (TPR_{DIDS}) e falso-positivos (FPR_{DIDS}) da plataforma de detecção distribuída, em função do valor de predição positiva médio dos IDSs membros da federação (PPV_{av}), considerando diferentes números de IDSs que detectam a intrusão (N_D). A figura 2D permite observar que o desempenho do sistema distribuído de detecção melhora mais rapidamente em função de (PPV_{av}) quando $N_D \to N_I$. Em ambas as curvas 4.4(b) e 4.4(a) é possível notar a linha sólida azul, quando o número



(a) FPR_{DIDS} (4.35); $N_D, N_I = [1-6]$.

(b) TPR_{DIDS} (4.36); $N_D, N_I = [1-6]$.

Figura 4.6: Avaliação 3D das métricas TPR_{DIDS} e FPR_{DIDS}, em função do número de IDSs atravessados pela intrusão (N_I) e do número dentre estes que a detectaram (N_D). Os valores de predição positiva média e da probabilidade de detecção foram fixados em $\text{PPV}_{av} = 0,459$ e $\text{PR}_{av} = 0,322$, respectivamente (média aritmética dos *datasets*).

de $N_D = 1$. Neste caso, quando somente um dentre os N_I IDSs detecta a falha, a plataforma se comporta como um IDS típico, variando linearmente com o PPV_{av}.

Sabe-se que a relação entre a taxas de verdadeiro-positivos e falso-positivos é uma métrica importante para avaliar a capacidade do IDS em alarmar positivamente e verdadeiramente. Esta métrica é conhecida como ROC (*Receiver Operating Characteristic* . A Figura 4.5 mostra uma aproximação do ROC da plataforma DIDS como $ROC_{DIDS} = TPR_{DIDS} - FPR_{DIDS}$, em função do número de IDSs membros que detectam a intrusão N_D. Analisando os comportamentos das curvas, é possível notar que, mesmo com valores de PR_{av} altos, o ROC se mantém em níveis elevados quando $N_D = N_I$. Isto significa que, mesmo com uma alta sensibilidade na detecção de positivos, a informação proveniente da combinação das evidências é confiável, graças à combinação consensual da proposta.

Na Figura 4.6, são mostrados duas superfícies representando o comportamento das métricas TPR_{DIDS} e FPR_{DIDS}, em função do número de IDSs atravessados pelo fluxo intruso (N_I) e do número de IDSs, dentre estes, que detectam a intrusão (N_D), onde $N_D \leq N_I$. Como esperado, o melhor desempenho é obtido quando N_D e N_I atingem seus valores máximos, mostrando que a combinação de evidências de detecção enviadas pelo sistema atua positivamente no desempenho da plataforma. Entretanto, também é possível notar uma degradação significativa nas métricas quando a diferença entre N_I e N_D aumenta.

A Tabela 4.6 compara os valores das métricas de desempenho da platafoma DIDS com os valores médios das métricas das duas matrizes de confusão do Snort, treinados nos *datasets* [1, 2] e também com os valores extraídos do sistema distribuídos de

detecção proposto em [124], calculados a partir do modelo analítico proposto nesta tese.

Tabela 4.6: Comparação das métricas de desempenho de detecção da plataforma DIDS para $N_D = 6$ com o Snort [1] e [2].

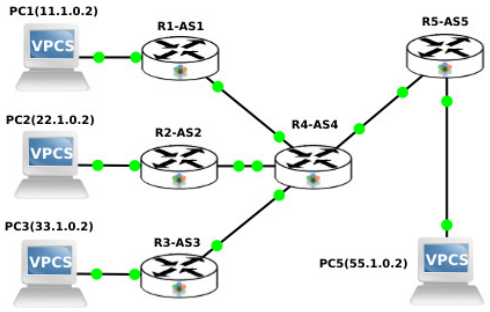
Métrica	**Valor** [1]	**Valor** [2]	**Média**	**Valor** [124]	**Valor DIDS**
PPV	$0,011$	$0,906$	$0,459$	$0,891$	$0,974$
TPR	$0,024$	$0,908$	$0,466$	$0,951$	$0,973$
TNR	$0,693$	$0,944$	$0,819$	$0,932$	$0,975$
FNR	$0,975$	$0,091$	$0,533$	$0,073$	$0,026$
FPR	$0,306$	$0,056$	$0,181$	$0,035$	$0,025$
PR	$0,271$	$0,373$	$0,322$	$0,897$	$0,998$

Os números mostrados na Tabela 4.6 mostram que o desempenho do sistema distribuído proposto nesta tese de doutorado supera largamente os sistemas típicos com base no Snort apresentados em [1] e [2]. A comparação com o sistema distribuído proposto em [124] é mais parelha, indicando também a coerência do modelo analítico proposto nesta tese.

Com relação à modelagem proposta, por tomar como base apenas as evidências positivas de intrusão enviadas pelos IDS membros, a mesma deixa a desejar na avaliação das métricas que só podem ser calculadas a partir das informações negativas de intrusão. É o caso da métrica NPV (*Negative Prediction Value*), que mede a capacidade do IDS em não emitir alarmes desnecessários para situações normais. Sem o NPV, não há como estabelecer uma relação direta entre a sensibilidade e a especificidade para levantar a curva do ROC. Outro problema decorrente desta deficiência da modelagem é o cálculo das métricas para $N_D = 0$, isto é, quando nenhum IDS detecta uma intrusão. Sem nenhuma evidência de intrusão através dos alarmes que chegam para serem combinados, não há como diferenciar a não ocorrência de uma intrusão de um alarme falso-negativo.

4.5 Modelo Experimental

O modelo experimental emula um cenário de rede com 5 roteadores e tem como principal objetivo avaliar o funcionamento do protocolo FlowSpec BGP, bem como sua viabilidade como infraestrutura de comunicação normalizada entre os membros da federação de IDSs. A topologia proposta para o cenário de teste considerado está mostrada na Figura 4.7(a).



(a) Visão da topologia na tela do GNS3

```
Network layer reachability information (19 bytes)
  FLOW_SPEC_NLRI (19 bytes)
    NRLI length: 18
    Filter: Destination prefix filter (55.1.0.2/32)
    Filter: Source prefix filter (11.1.0.2/32)
    Filter: IP protocol filter (=1)
    Filter: ICMP type filter (=8)
Network layer reachability information (19 bytes)
  FLOW_SPEC_NLRI (19 bytes)
    NRLI length: 18
    Filter: Destination prefix filter (55.1.0.2/32)
    Filter: Source prefix filter (22.1.0.2/32)
    Filter: IP protocol filter (=1)
    Filter: ICMP type filter (=8)
Network layer reachability information (19 bytes)
  FLOW_SPEC_NLRI (19 bytes)
    NRLI length: 18
    Filter: Destination prefix filter (55.1.0.2/32)
    Filter: Source prefix filter (33.1.0.2/32)
    Filter: IP protocol filter (=1)
    Filter: ICMP type filter (=8)
```

(b) Registros de mensagens coletados na WAN do roteador $R5$

Figura 4.7: Topologia de rede composta por 5 roteadores de borda, cada um no seu respectivo AS, simulando um ataque coordenado das origens *PC1*, *PC2* e *PC3* contra o *PC5*.

O modelo utiliza o software de emulação GNS3 [125], instalado numa máquina com sistema operacional Ubuntu 16.04 Server. O cenário de teste é composto por 5 roteadores virtuais, executando uma mesma versão de imagem Junos 12.1 da Juniper.

Os roteadores *R1*, *R2* e *R3* divulgam a atualização BGP-FlowSpec de um fluxo, tido como intruso, com destino 55.1.0.2/32, origem 11.1.0.2/32, protocolo IP/ICMP.

As mensagens BGP são capturadas na interface WAN do roteador *R5* e combinadas de acordo com os campos coincidentes. Os registros de Wireshark [126] da Figura 4.7(b) resume o conteúdo do campo NLRI das três mensagens coletadas. Uma possível combinação, que pode ser feita no $AS5$, é pelo campo IP de destino das mensagens recebidas.

Capítulo 5

Sistema de Auto-Defesa e Acurácia Baseada em Aprendizado de Máquina

A colaboração e o compartilhamento de informações entre agentes distribuídos tem contribuído bastante para melhorar a acurácia das decisões e reduzir o tempo de reação em muitas aplicações de defesa [127]. O sistema distribuído de detecção de intrusões é uma destas aplicações que surgiram a partir de ideias de cooperação para fazer frente ao crescente número de ataques e sofisticação das intrusões. Apesar dos comprovados benefícios no aumento do desempenho operacional na redução da taxa de alarmes falso-positivos e no aumento na taxa de alarmes verdadeiro-positivos, surgem também questões de segurança relacionadas com eventuais ataques contra o próprio sistema de detecções. Neste aspecto, a própria natureza da aplicação, o ambiente colaborativo entre membros heterogêneos e espelhados geograficamente fazem dos sistemas DIDS alvos cobiçados para ataques cibernéticos. Para lidar com este problema e reduzir os riscos de ataques, mecanismos de credenciamento de membros e avaliação da confiança tem sido propostos pela academia para compor estratégias de auto-defesa e acurácia dos sistemas distribuídos de detecção de intrusões [124, 128].

As abordagens de auto-defesa em sistemas de detecção de intrusões se referem ao conjunto de mecanismos com o objetivo de proteger o IDS de ataques contra sua própria infraestrutura ou para evitar sua utilização como vetor em ataques contra terceiros. O levantamento proposto em [129] relaciona e classifica os principais ataques contra sistemas de detecção de intrusões, propondo uma taxonomia baseada nas vulnerabilidades de cada um dos seus blocos funcionais (Figura 2.2).

Em se tratando de um sistema distribuído globalmente para detecção de intrusões, confiar nas informações que são compartilhadas cooperativamente entre os

membros da federação é um tema crítico, tanto no domínio da auto-defesa quanto na acurácia. Um IDS membro da federação DIDS controlado para fins maliciosos ou que não esteja funcionando adequadamente pode enviar informações de intrusão falsas ou desnecessárias, sobrecarregando o processo de fusão e comprometendo a detecção de intrusões verdadeiras de um AS (*overstimulation attack*). Pior ainda, se alguns dos IDSs membros forem controlados maliciosamente para criar uma coalizão de ataque com o objetivo de enviar muitos alarmes falsos coordenadamente, o funcionamento da plataforma como um todo pode ser seriamente comprometido.

O artigo publicado em [78] discorre justamente sobre o grave perigo que corre a Internet nos dias de hoje por causa da possibilidade de ataques baseados no protocolo BGP. Assim, considerando ainda o ambiente federativo geograficamente distribuído e heterogêneo proposto neste capítulo, avaliar a confiabilidade e a precisão de uma mensagem BGP que chega ao AS de destino para ser combinada e processada é de suma importância. O sistema de combinação das massas de crença das mensagens de atualização correlatas que chegam num determinado AS, proposto na Seção 4.2.2, já proporciona um nível básico de segurança para uma eventual decisão baseada na massa de crença combinada. Entretanto, é considerável a hipótese de um determinado AS_j de destino receber apenas um único anúncio de um determinado AS_i de origem, com massa de crença PPV_i pequena. Neste caso, o sistema de detecção pode negligenciar sua ocorrência, criando uma situação grave, análoga à ocorrência de um evento falso-negativo.

Neste capítulo será apresentado um sistema complementar de auto-defesa para aumentar a robustez da plataforma em caso de ataques internos e também para melhorar a acurácia das detecções. O sistema proposto se baseia na consolidação da massa de crença de cada anúncio BGP-FlowSpec M_{C_i}, representada até o momento pela precisão do agente federado PPV_i (vide Equação 4.7), usando um parâmetro adicional, que represente a reputação do AS_i $\mathrm{RPT}(i) \in [0, 1]$ que originou o anúncio a ser combinado num determinado AS de destino AS_j.

$$M_{C_i} = PPV_i \times RPT(i) \tag{5.1}$$

Neste ponto, é importante observar que a especificação BGP-FlowSpec [26] já prevê mecanismos para validar a divulgação do fluxo pelo AS originador. São eles:

- A informação do prefixo dentro do AS de destino deve vir incorporada na especificação do fluxo.
- O AS originador da mensagem BGP-FlowSpec deve coincidir como a melhor rota *unicast* para o prefixo de destino anunciado dentro do campo NLRI da mensagem.

- Não pode haver rota *unicast* mais específica para o destino incorporado no campo NLRI, recebida de outros ASs vizinhos, do que o anúncio feito pelo originador da mensagem FlowSpec.

Em resumo, as regras acima garantem que o AS originador do anúncio BGP-FlowSpec sempre faça parte do melhor caminho (*AS-Path*) para o prefixo de destino incorporado no campo NLRI do anúncio. Apesar desta associação entre o fluxo divulgado e a tabela de rotas do AS originador, ainda é bastante possível que este AS originador divulgue sua mensagem FlowSpec com objetivos maliciosos [130].

Em virtude da grande diversidade de membros prevista numa vislumbrada federação global de IDSs, haveria grande dificuldade em inferir sobre a credibilidade das mensagens BGP-FlowSpec que chegam para serem combinadas no AS de destino. Seja pelo desempenho operacional dos membros (PPV) da federação, seja pela possibilidade de envios maliciosos, visando comprometer o processo de detecção no destino, as informações que chegam não são suficientes para estabelecer uma relação direta com a massa de crença de cada anúncio. Deste modo, uma boa opção seria a criação de um *dataset* rotulado, contendo alguns atributos estratégicos extraídos do campo "atributos de caminho" (*path attributes*) da própria mensagem de atualização BGP-FlowSpec e um rótulo relacionando com a ocorrência ou não de uma intrusão contra o AS de destino. Este *dataset* poderia ser utilizado para treinar um modelo de aprendizado de máquina baseado em regressão, com o objetivo de inferir sobre a reputação do AS_i (REP_i) e consolidar a massa de crença de novos anúncios BGP-FlowSpec (Equação 5.1), antes do processo de correlação e fusão para obtenção da massa de crença combinada M_C, de acordo com a Figura 5.1.

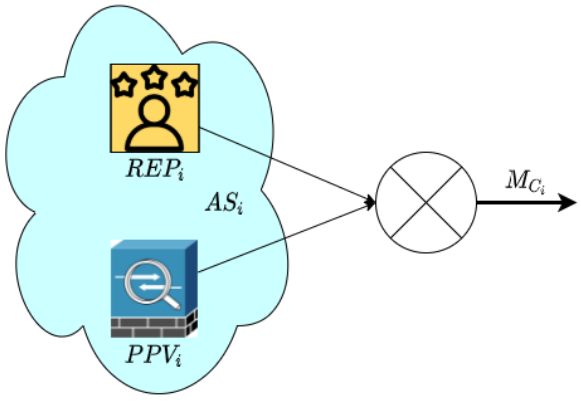


Figura 5.1: Massa de crença consolidada M_{C_i} do AS_i a ser depois combinada no AS_j, dando origem à evidência de intrusão com massa de crença combinada geral M_C.

Duas hipóteses então devem ser analisadas.

1. Se os atributos extraídos do anúncio BGP são suficientes para inferir sobre a ocorrência ou não de uma intrusão.

2. Se o modelo de aprendizado de máquina utilizado conseguiria generalizar esta inferência para novos anúncios.

Ao invés de se apegar aos detalhes técnicos do modelo de aprendizado, considerando os vários critérios de escolha do modelo e as muitas opções de ajustes para garantir o melhor resultado, o principal objetivo neste Capítulo 5 é mostrar que é possível inferir sobre a massa de crença de um anúncio BGP ordinário, através de um modelo de aprendizado de máquina simples, treinado a partir de um *dataset* constituído apenas por alguns atributos extraídos do próprio anúncio, individualmente.

5.1 Base de Dados do Modelo

O processo de modelagem de qualquer sistema de aprendizado de máquina deve se iniciar sempre pelo perfeito entendimento do problema a ser endereçado e do objetivo do modelo. Neste caso, deseja-se obter um algoritmo para inferir sobre a massa de crença de um anúncio de atualização BGP ($M_C \in [0, 1]$), tendo como base um conjunto de atributos extraídos desta mensagem, especialmente escolhidos de acordo com o objetivo do modelo.

O segundo passo na modelagem do sistema de aprendizado é a aquisição dos dados para a montagem do *dataset*. A aquisição dos dados pode ser feita de duas formas:

1. Buscar uma base de dados aberta da Internet que faça sentido no escopo do modelo de aprendizado pensado.

2. Formar uma base de dados específica levando em conta o cenário do problema atacado.

Como ainda não existe nenhum sistema distribuído de detecção de intrusões que emita anúncios BGP-Flowspec para alarmar situações potenciais de ataques, também não existe nenhum *dataset* público derivado deste ambiente que possa ser utilizado neste modelo.

O trabalho proposto em [87] compara vários sistemas de detecção de anomalias baseados em algoritmos de aprendizado de máquina. Os modelos propostos em [87] utilizam um *dataset* com 66 atributos extraídos de anúncios BGP, coletados durante eventos de escala global, tais com o ataque Code Red II e o terremoto

de Fukushima. Os resultados obtidos a partir das matrizes de confusão obtidas mostram uma acurácia superior a 90%.

O Code Red II foi um ataque do tipo “verme" (*worm*), cuja primeira ocorrência documentada se deu em 19 de julho de 2001, entre 10:00 e 20:00 horas GMT. Ainda é considerado um dos maiores ataques de todos os tempos, seja em termos de alcance, com 359.000 computadores infectados ao redor do globo em menos de 14 horas, seja pelo do prejuízo aproximado de 2,6 bilhões de dólares americanos. O código malicioso se aproveitava de uma vulnerabilidade de estouro de *buffer* (*buffer-overflow*) do sistema *Microsoft Internet Information Services* para configurar uma porta de acesso desautorizada para atacante ao sistema (*backdoor*).

O modelo de aprendizado de máquina proposto neste trabalho para inferir sobre a massa de crença dos anúncios BGP se inspira em [87] para propor um *dataset* de apenas 15 atributos diretos e indiretos. Entretanto, diferentemente do contexto volumétrico de detecção de anomalias apresentado em [87], o *dataset* proposto neste trabalho possui apenas atributos pontuais, extraídos individualmente de cada anúncio que chega ao AS de destino para ser combinado. Em outras palavras, ao invés de consolidar um conjunto de anúncios coletados durante um determinado período, como feito em [87], o *dataset* utilizado neste trabalho cria uma linha atributos para cada anúncio que chega para ser combinado. A razão que motiva esta análise individual é a própria abordagem alarmística do sistema de detecção proposto neste Capítulo 4, onde cada anúncio BGP-FlowSpec que chega num AS de destino é um alarme de intrusão vindo da Internet.

Para montar o *dataset*, foram utilizados os registros de anúncios BGP do coletor RRC04 na base pública RIPE NCC [84]. Segundo [87], o coletor RRC04 recebeu vários anúncios BGP pelos seus ASs vizinhos 513, 559 e 6893 durante o período de ataque do Code Red II. Foram escolhidos registros de três períodos distintos: antes do ataque (dia 12/07/2001), durante o ataque (dia 19/07/2001) e após o ataque (dia 26/07/2001) todos das 12 às 15 horas GMT. A Figura 5.2 abaixo mostra as conexões de vizinhança BGP do RIPE AS 12654 no RRC04.

5.2 Atributos Diretos

Dentro de um sistema de aprendizado de máquina, os atributos se referem às representações das características estratégicas dos dados de entrada que permitem ao modelo “aprender" com os mesmos. Considerando a função de transferência derivada do aprendizado, cada atributo tem o sua importância ou peso no resultado esperado. Por exemplo, se for considerado um sistema de aprendizado de máquina para inferir sobre a prevalência de uma pessoa para o câncer, o atributo com a informação se esta pessoa é fumante ou não terá um peso maior que outro atributo informando se

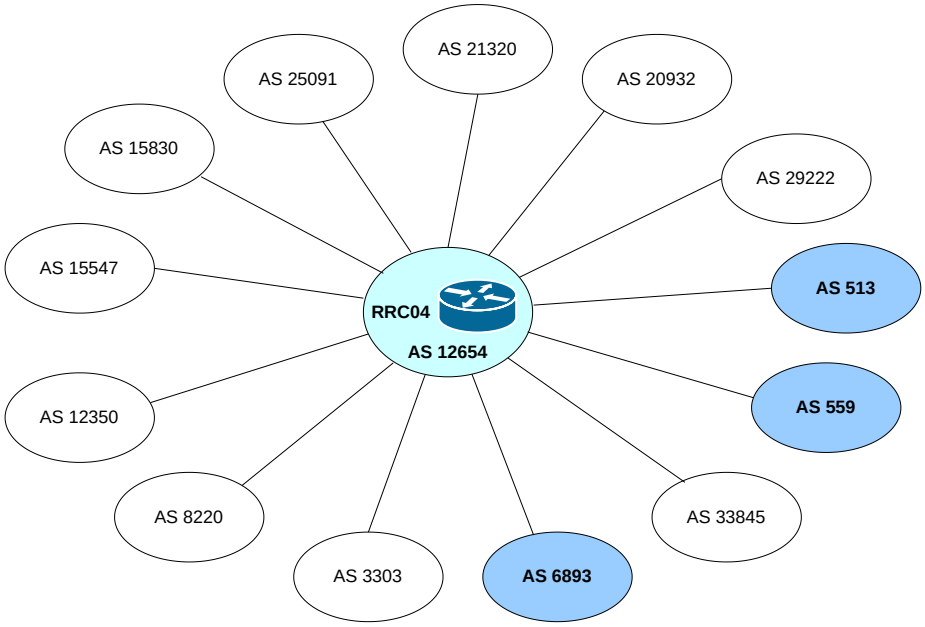


Figura 5.2: Esquema com os sistemas autônomos vizinhos do AS do RIPE (12654) no coletor RRC04 (Genebra).

o indivíduo mora numa casa ou num apartamento.

Origem do Anúncio

O primeiro atributo direto que pode ser extraído do campo "atributo de caminho" da mensagem de atualização BGP-FlowSpec (*path attribute*) é o campo *ORIGIN*. Este campo especifica a origem da informação de roteamento do campo NLRI que está presente na mensagem. Conforme descrito na Seção 2.6, este campo pode assumir três valores:

- IGP(0)- Quando a informação de roteamento provém de dentro do próprio AS que anuncia a mensagem de atualização.
- EGP(1) - Quando a informação de roteamento foi recebida de um outro AS.
- INCOMPLETE(2) - Quando a origem da informação do campo NLRI é outra diferente das duas primeiras.

Dentro do contexto do sistema distribuído de detecção de intrusões, o atributo de origem tem relação direta com a topologia. Por exemplo: uma mensagem BGP-FlowSpec com o campo *ORIGIN* igual a 0 leva a três hipóteses:

- Que um fluxo supostamente malicioso na direção do AS de destino atravessou um AS vizinho adjacente e foi detectado pelo seu IDS federado.
- Que um fluxo supostamente malicioso contra um determinado prefixo de destino foi gerado no próprio AS vizinho adjacente que gerou o anúncio.

- Que se trata de uma tentativa de comprometer o AS de destino com uma mensagem falsa para o AS de destino.

Em contraste, um anúncio de um fluxo supostamente malicioso que chega ao AS de destino com origem EGP tende a ser menos confiável, uma vez que não foi detectado por nenhum outro AS no percurso entre a origem e o destino (*AS_PATH*).

Número de Repetições

O mecanismo de *PRE_PENDING* é um artifício comumente utilizado para fazer engenharia de tráfego nos ASs. O mecanismo consiste basicamente em se inflar o campo *AS_PATH*, repetindo o ASN divulgador do anúncio múltiplas vezes. Como o tamanho do *AS_PATH* é o segundo critério de desempate na escolha da rota, o *PRE_PENDING* visa sobretudo reduzir a probabilidade daquele caminho ser escolhido para alcançar o prefixo anunciado. Apesar de sua simplicidade e de ser muito utilizado na Internet, existem questões de segurança a serem observadas. Ataques de DDoS podem se utilizar deste mecanismo para sobrecarregar interconexões entre ASs [131].

Tamanho do Caminho do Anúncio

O atributo de tamanho do caminho do anúncio se refere à quantidade de ASs percorridos pelo anúncio, deste a sua origem até o seu destino, onde será correlacionado e combinado. Este atributo poder ser extraído do campo *AS_PATH_LENGTH* dentro do *AS_PATH*. Durante a extração deste dado de dentro do anúncio, é importante observar se existem repetições de ASNs. A ocorrência de um mesmo ASN repetidas vezes dentro do campo *AS_PATH* indica a utilização do *PRE_PENDING*, comummente utilizado para fazer engenharia de tráfego. A escolha deste atributo se justifica pelo comportamento do tráfego de Internet no que se refere ao número médio de ASs que são atravessados desde a origem até o destino. Trabalhos recentes como em [132] mostram que a maior parte do tráfego na Internet atravessa até cinco ASs até chegar ao seu destino. Assim, anúncios originados em ASs distantes tendem a ser menos prováveis, e portanto menos confiáveis, do que anúncios provenientes de ASs mais próximos.

5.3 Atributos Indiretos

Os atributos indiretos são obtidos a partir das informações presentes nos dados de entrada, necessitando de algum processamento para se transformarem em atributos.

Representatividade do AS Originador

A representatividade de um determinado AS_i REP(i) é medida em função do número de prefixos que podem ser alcançados através deste AS (cone). O cálculo da representatividade de um AS leva em consideração o número de outros ASs com os quais se interliga, o seu número de prefixos e o seu número de endereços. Atualmente a organização CAIDA [133] oferece uma lista ordenada de todos os ASs da Internet, de acordo com a sua representatividade.

Reputação Comunitária do AS Originador

A reputação de um determinado AS_i em relação à comunidade de ASs PRE(i) da Internet pode ser aferida através de ferramentas públicas disponíveis na Internet [134]. O cálculo do fator de impacto de um determinado AS é calculado com base na quantidade dos seus prefixos presentes em listas negras espalhadas pela rede. A utilização deste atributo se justifica como um indicativo da vulnerabilidade do AS gerador de anúncio, sugerindo a hipótese de ter seu ambiente IGP ou o próprio IDS federado comprometido.

Representatividade Média dos ASs

A representatividade média dos ASs é obtida a partir do cálculo do valor médio das representatividades de todos os n ASs que compõem o caminho do anúncio (AS_PATH), desde a origem até o AS de destino. Um anúncio de passa por ASs de baixa representatividade pode representar um possível ataque de sequestro de prefixo (*prefix hijacking*), explicado em [130].

$$\overline{REP}(AS\_PATH) = \frac{1}{n}\sum_{i=1}^{n} REP(AS_i) \tag{5.2}$$

Reputação Comunitária Média dos ASs

Da mesma forma que o atributo anterior, a reputação média comunitária é obtida a partir do cálculo do valor médio da reputação de todos os n que compõem o caminho do anúncio (AS_PATH), desde a origem até o AS de destino. O anúncio que percorre um caminho com baixa reputação médio pode ser interceptado e modificado maliciosamente por um AS.

$$\overline{PRE}(AS\_PATH) = \frac{1}{n}\sum_{i=1}^{n} PRE(AS_i) \tag{5.3}$$

Representatividade Média dos AS Vizinho

O AS vizinho (*AS_PEER*) é último AS do caminho por onde chega o anúncio, fazendo fronteira direta com o AS considerado alvo ou destino. Geralmente uma vizinhança BGP entre dois ASs distintos são celebrados dentro de um acordo de negócio e confiança. Apesar de estar diretamente ligada à parte do negócio no acordo de vizinhança, é esperado que não se receba mensagens maliciosas dos ASs vizinhos.

Reputação Comunitária do AS Vizinho

A reputação de um candidato a AS vizinho (*AS_PEER*) está relacionada com a parte de confiança do acordo de vizinhança e é verificado antes do estabelecimento da mesma. Entretanto, esta pode variar ao longo do tempo em função do comportamento do vizinho, trazendo riscos para a outra parte [135].

Máxima Representatividade Dentro do Caminho

A máxima representatividade do caminho é o maior valor de representatividade aferido de cada AS que compõe o caminho, desde sua origem até o seu AS de destino.

Mínima Representatividade Dentro do Caminho

A mínima representatividade do caminho é o menor valor de representatividade aferido de cada AS que compõe o caminho, desde sua origem até o seu AS de destino.

Representatividade Mediana dentro do Caminho

A representatividade mediana do caminho é o valor central de representatividade, considerando todos os ASs do caminho. O valor da mediana é interessante como atributo num *dataset* em situações de dados distorcidos, como em distribuições de cauda pesada.

Máxima Reputação Dentro do Caminho

A máxima reputação do caminho é o maior valor de reputação aferido de cada AS que compõe o caminho, desde sua origem até o seu AS de destino.

Mínima Reputação Dentro do Caminho

A mínima reputação do caminho é o menor valor de reputação aferido de cada AS que compõe o caminho, desde sua origem até o seu AS de destino.

Reputação Mediana dentro do Caminho

A reputação mediana do caminho é o valor central de reputação, aferida para cada um dos ASs do caminho. O valor da mediana é interessante como atributo num *dataset* em situações de dados distorcidos, como em distribuições de cauda pesada.

5.4 Testes Não-Supervisionados

Com já foi explicado no início da Seção 2.7, o modelos de aprendizado não-supervisionado tenta criar superfícies de separação a partir de padrões de concentração de dados. Em outras palavras, ao invés de fazer uma predição, os métodos não-supervisionados são utilizados para descobrir a estrutura do conjunto de dados. Os algoritmos de aprendizado não-supervisionados não utilizam rótulos para chegar nos resultados esperados, sendo portanto comumente utilizados como passo inicial, permitindo evoluir mais rapidamente para modelos supervisionados.

No modelo não-supervisionado, tem-se um conjunto de N exemplos $(x_1, x_2, ..., x_N)$ de um vetor d-dimensional X, definido por $\Pr(X)$. O objetivo é inferir sobre a PDF $\Pr(X)$ dispensando a ajuda dos rótulos do conjunto de dados, considerando uma determinada margem de erro. Com foco neste objetivo e com base no conjunto de dados montado conforme descrito na Seção 5.1, serão apresentados a seguir testes e resultados de modelos de aprendizado de máquina não-supervisionados.

Um dos principais algoritmos de aprendizado não-supervisionado é o de aglomeração ou concentração (*clustering*). As técnicas de aglomeração são muito utilizadas para particionar ou segmentar um conjunto de dados em grupos distintos, dentro dos quais as observações são muito similares umas com as outras e fora dos quais são muito diferentes. Fundamental para todas as técnicas de agrupamento é a escolha da medida de distância ou dissimilaridade entre dois objetos.

A medida de distância pode ser obtida a partir do valor médio de todas as medidas de distância dos dados. As medidas podem ser organizadas numa matriz de distâncias $\mathbb{D}$ simétrica $N \times N$, onde N corresponde ao número de observações e $d_{ii'}$ representa a distância entre o i-ésimo e o j-ésimo objetos, onde $d_{ii'} = 0 \; \forall \; i = i'$; $i, j = \{1, 2, ..., N\}$. Considerando o valor x_{ij} ; $i = \{1, 2, ..., N\}$ para um determinado atributo j ; $j = \{1, 2, ..., p\}$, a matriz de dissimilaridade $\mathbb{D}$ pode ser obtida como:

$$\mathbb{D} = \sum_{j=1}^{p} d_j(x_{ij}, x_{i'j}). \tag{5.4}$$

De longe, a forma mais comum para calcular a dissimilaridade entre o mesmo atributo j da observação i e i' é o erro quadrático.

$$d_j(x_{ij}, x_{i'j}) = (x_{ij} - x_{i'j})^2. \tag{5.5}$$

O critério de agrupamento, além de representar o principal aspecto de um algoritmo de agrupamento, também está ligado à maioria das alternativas de avaliação dos resultados do algoritmo. Por exemplo, enquanto o algoritmo K-médias identifica mais facilmente agrupamentos esféricos, o agrupamento hierárquico funciona melhor para captar a densidade dos agrupamentos [136].

5.4.1 Aglomeração K-médias

O método K-médias consegue particionar um conjunto de dados (*dataset*) em K diferentes grupos não-sobrepostos, garantindo que a variação total entre os pertencentes de um mesmo grupo seja a menor possível, em todos os K grupos. Assim, antes de iniciar o algoritmo, é necessário especificar um valor para K. Neste ponto, é importante ressaltar que a escolha do valor de K interfere diretamente no desempenho do algoritmo e nos resultados do modelo. Considere um total de n observações e $C_1, ..., C_K$ conjuntos com a identificação das observações em cada grupo. Estes conjuntos precisam satisfazer a duas propriedades básicas:

1. $C_1 \cup C_2 \cup C_K = 1, ..., n$. Isto é, cada observação deve pertencer a no mínimo a um dos K grupos.

2. $C_k \cap C_{k'} = \emptyset \ \forall \ k \neq k'$. Em outras palavras, os grupos nunca se sobrepõem: uma observação nunca poderia pertencer a mais de um grupo.

Conforme explicado no primeiro parágrafo, a principal desvantagem do método K-médias é a necessidade de se especificar o valor de K antes da execução. Uma boa prática neste caso é analisar primeiramente o dendograma e utilizar este resultado para testar no K-médias. Neste caso, considerando a análise apresentada na Seção 5.4.2, o algoritmos foi executado com $K = 2$ (Grupo_A e Grupo_B) e o resultado pode ser conferido na Figura 5.3.

A métrica utilizada para medir a dissimilaridade entre os grupos (aglomerações) $W(C_k)$ é a distância euclidiana quadrática, de acordo com a Equação 5.6 abaixo.

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 \tag{5.6}$$

Como pode ser observado no código Python B.1, o critério utilizado para dividir os grupos foi o da variância mínima (Ward [137]). O método de Ward define a distância entre dois grupos A e B em quanto a soma dos quadrados aumenta à medida que os grupos são juntados. Em outras palavras, num mesmo grupo (*cluster*) estão as

Figura 5.3: Gráfico 2-médias - Grupos A e B - considerando três diferentes *datasets*: 0 → fora do período de ataque, 1 → dentro do período de ataque e 2 → ambos, obtidos dos 3 ASs vizinhos (513, 559 e 6893).

observações de menor variância em relação à distância euclidiana. A variância entre duas observações quaisquer de grupos diferentes sempre será maior que a variância interna de um grupo qualquer.

$$\triangle(A,B) = \frac{n_A n_B}{n_A + n_B} ||\overrightarrow{m}_A - \overrightarrow{m}_B||^2 \tag{5.7}$$

Na Equação 5.7, o termo $\triangle(A,B)$ se refere à função de custo de fusão dos grupos A e B. Os termos n_A e n_B se referem ao número de observações nos grupos A e B respectivamente. O termo $\overrightarrow{m}_A$ e $\overrightarrow{m}_B$ representam respectivamente as centroides dos conjuntos A e B, respectivamente.

A Figura 5.3 mostra uma fotografia das observações dos três *datasets* (dia normal-0, dia do ataque-1 e ambos os dias juntos-2) divididas em dois grupos (Grupo_A e Grupo_B), de acordo com o AS vizinho (AS_peer) recebido (513, 559 e 6893). Os resultados mostram uma alteração bastante significativa no comportamento do número de observações aglomeradas nos grupos A e B. Por exemplo, enquanto durante o período fora de ataque (*dataset* 0) a relação A/B é favorável ao Grupo A, considerando os períodos que contém registros de ataque (*datasets* 1 e 2) esta relação se inverte favoravelmente ao Grupo B nos três ASs vizinhos.

5.4.2 Aglomeração Hierárquica

A técnica de aglomeração hierárquica é uma das mais atrativas por não necessitar de nenhuma pré-definição para produzir o resultado, além de apresentá-lo num formato gráfico bastante informativo e fácil de interpretar, o **dendograma**.

O dendograma consiste basicamente num gráfico em formato de árvore invertida, cuja construção se inicia pelas folhas que se combinam nos troncos, de acordo com a similaridade entre os dados observados. O eixo horizontal mostra todas as n observações a respeito dos dados que são analisados e separados. O eixo vertical mostra o nível de separação entre os grupos. Fazendo uma análise visual na direção de baixo para cima, quanto antes as folhas forem conectadas nas ramificações, maior será a similaridade mútua dos dados.

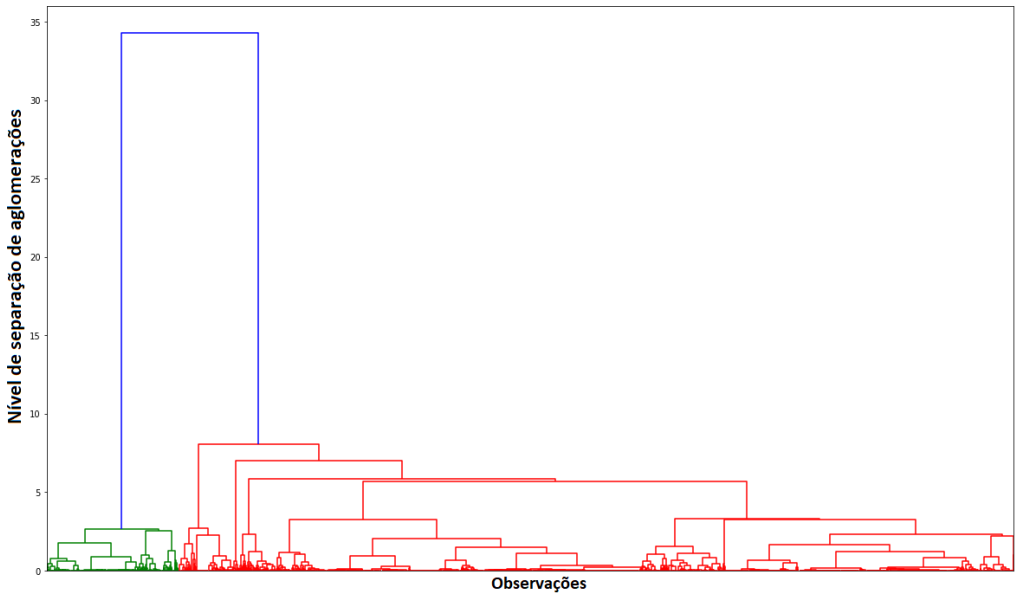


Figura 5.4: Dendograma.

A Figura 5.4 mostra que existem basicamente duas aglomerações bem definidas em verde e vermelho, que são separadas por uma distância de aproximadamente 16 unidades quadráticas euclidianas.

Assim como no código da Seção anterior 5.4.1, o código utilizado para gerar o dendograma B.2, a medida de dissimilaridade (por padrão) e o critério de divisão dos grupos são os mesmos: distância euclidiana quadrática e mínima variância.

5.5 Testes Supervisionados

Tendo em mente os resultados obtidos nos testes não-supervisionados da Seção 5.4, é possível concluir que o conjunto de dados com os 15 atributos extraídos dos anúncios BGP pode ser dividido em dois grupos significativamente diferentes (grupos A e B). Entretanto, apesar desta importante conclusão, ainda não é possível extrair o nível de confiança destes anúncios, nem inferir sobre a massa de crença dos novos anúncios. Para tanto, faz-se necessária utilização de modelos supervisionados, que requerem uma coluna adicional no *dataset*, cujo valor infere sobre massa de crença

$(0 \leq M_C \leq 1)$ do registro com atributos, como fruto de um ataque real ou de uma operação normal do sistema BGP.

5.5.1 Conjunto de Dados Rotulado para Treinamento

O conjunto de dados rotulado para treinamento do modelo supervisionado foi construído a partir do *dataset* extraído do dia 19/07/2001 (dia do ataque Code Red II), contendo rótulos de "ataque" e "não-ataque". A este novo conjunto de dados - contendo os 15 atributos especificados na Seção 5.1 extraídos dos anúncios BGP recebidos das vizinhanças 513, 559 e 6893 do coletor RIPE RRC04 - foi adicionada uma nova coluna (ataque) com o seguinte rótulo:

- Rótulo 1 - indica que aquele registro é fruto de um ataque.
- Rótulo 0 - indica que aquele registro não é fruto de nenhum ataque.

A atribuição do rótulo foi feita a partir de uma combinação dos registros deste novo *dataset* mencionado acima, com o *dataset* disponibilizado no trabalho em [138], cuja acurácia na detecção do ataque Code Red II ocorrido no dia 19/07/2001 através das mensagens BGP chegou a 98%. O *dataset* rotulado resultante desta combinação foi dividido em três partes, de acordo com a estratégia de treinamento, teste e validação do modelo, proposta em [139].

- Treinamento - Partição rotulada do conjunto de dados, utilizado exclusivamente para treinar o modelo supervisionado.
- Validação - Partição rotulada do conjunto de dados utilizada para validar as alterações e refinar o modelo supervisionado através de testes comparativos dos rótulos originais com os rótulos inferidos pelo modelo.
- Teste - Partição rotulada do conjunto de dados utilizada para testar o modelo supervisionado, consolidando seu desempenho funcional e a sua capacidade de generalizar na predição sobre dados novos.

A Figura 5.5 abaixo mostra graficamente como foi feita a divisão do *dataset* para cumprir os objetivos de treinamento, testes e validação do modelo supervisionado.

5.5.2 Redes Neurais

O problema de aprendizado supervisionado pode ser definido como um problema de procurar uma solução dentro de um determinado espaço [136]. Algumas técnicas de aprendizado de máquina realizam esta busca pela hipótese que descreve os dados recorrendo à otimização de algumas funções, conhecidas como funções de custo.

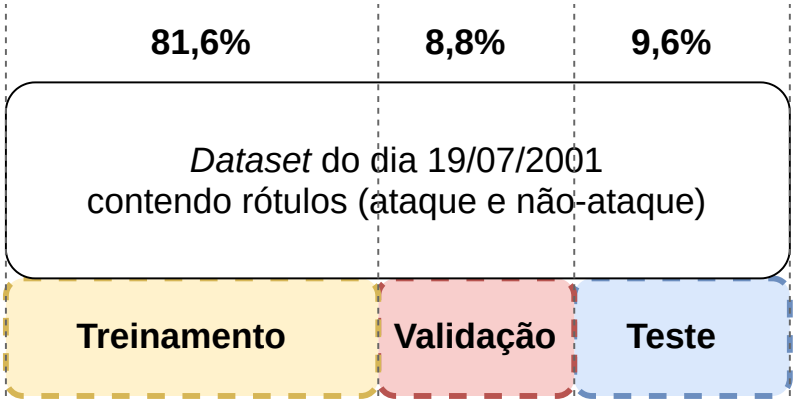


Figura 5.5: Estratégia de sub-divisão do *dataset* de treinamento para validação das alterações e teste para aferição do desempenho do modelo de aprendizado.

Este é o caso da rede neural. AS redes neurais artificiais são sistemas computacionais inspirados, mas não idênticos, às redes neurais biológicas, que conseguem "aprender" através de exemplos. Sua arquitetura peculiar, formada por neurônios interconectados através de setas com diferentes pesos W, recebe um vetor com p variáveis de entrada $X = (X_1, X_2, ..., X_p)$ e constrói uma função não-linear $f(X)$, capaz de predizer a resposta Y, de acordo com o algoritmo de treinamento utilizado. A Figura 5.6 abaixo mostra uma rede neural simples de apenas uma camada com $K = 5$ unidades de predição.

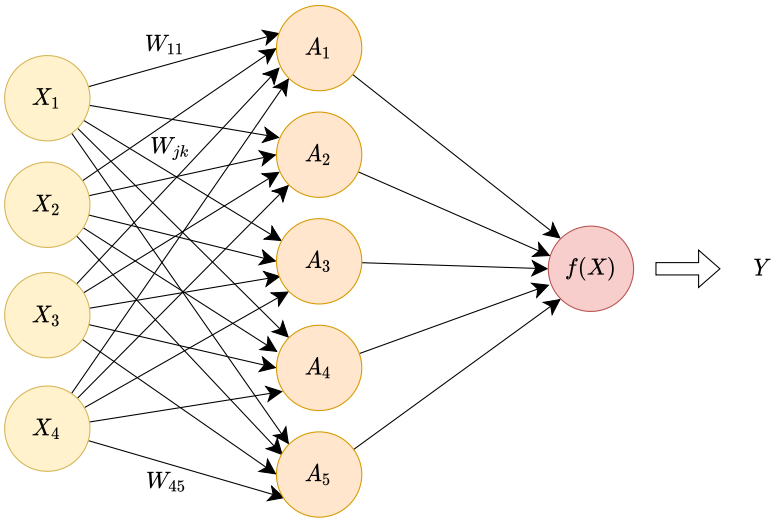


Figura 5.6: Exemplo de rede neural comum sem retroalimentação (*feedforward*) com apenas uma camada entre as camadas de entrada e saída.

A função não-linear $f(X)$ pode ser calculada de acordo com a Equação 5.8 abaixo.

$$f(X) = \beta_0 + \sum_{k=1}^{K} \beta_k A_k \tag{5.8}$$

Onde:

$$A_k = g(W_{0k} + \sum_{j=1}^{p} W_{jk} Xj) \tag{5.9}$$

Na Equação 5.8, $g()$ é conhecida como função de ativação e será detalhada posteriormente nos testes e os termos $W_{0k}, ..., W_{pK}$ serão estimados no processo de treinamento do modelo.

5.5.3 Modelo de Aprendizado

A escolha do modelo de aprendizado para a obtenção do valor da massa de crença dos anúncios BGP foi feita através do ambiente *tensorflow* do Python. Neste ambiente é possível testar sequencialmente diferentes combinações de parâmetros (hiperparâmetros) do modelo de aprendizado e escolher a melhor combinação, de acordo com critérios pré-selecionados.

Na lista de comandos B.3, o algoritmo testa modelos entre 16 e 128 neurônios de entrada e saída. O algoritmo também testa diferentes funções de ativação:

- relu - *rectified linear activation function* - espelha na saída o exato valor da entrada, desde que este seja positivo. Caso contrário retorna zero.
- tanh - fórmula exponencial da tangente hiperbólica retorna valores entre -1 e 1 na saída, independentemente do valor de entrada - $\theta(s) = \frac{e^s - e^{-s}}{e^s + e^{-s}}$.
- sigmoid - a função sigmoide retorna valores entre 0 e 1, independentemente do valore de entrada - $\theta(s) = \frac{1}{1+e^{-s}}$.

Os valores de *dropout* indicam a frequência em que o algoritmo testa a desabilitação randômica de algumas entradas numa determinada frequência.

A taxa de aprendizado ou *learning rate* indica o tamanho do passo do algoritmo gradiente descendente para se chegar no valor mínimo da função de custo.

A função de perda (ou custo), associada às perdas quando as predições de classificação forem incorretas é a entropia binária cruzada $H_p(q)$, que pode ser calculada como:

$$H_p(q) = \frac{1}{N} \sum_{i=1}^{N} y_i \log[p(y_i)] + (1 - y_i) \log[1 - p(y_i)] \tag{5.10}$$

A proposta da função de perda é avaliar as predições do modelo, retornando valores altos para predições ruins ou valores baixos para predições mais próximas do rótulo. Assim, pode-se dizer que o processo de aprendizado neste caso “tenta" minimizar a função de perda, aproximando o valor das predições dos seus rótulos. Na Equação 5.10, os termos p e q se referem às funções distribuição do processo de

predição e dos rótulos respectivamente. O termo N se refere ao número de registros no conjunto de dados de treinamento ou *dataset* de treinamento.

O modelo automaticamente escolhido pelo algoritmo de acordo com os critérios de melhor resultado possui duas camadas: uma camada de entrada com 112 neurônios e uma camada de saída com 128 neurônios. A função de ativação escolhida foi a de espelhamento dos positivos (relu). A taxa de desabilitação (*dropout*) escolhida foi de $0,1\%$ para as duas camadas. A taxa de aprendizado (*learning rate*) foi selecionada em $0,01$.

5.5.4 Resultados

Na estratégia de treinamento, validação e testes mostrada na Seção 5.1 e considerando uma classificação simples para o valor da massa de crença do anúncio como fruto de um ataque real ($M_C > 0.5$) ou não ($M_C \leq 0.5$), é possível extrair acompanhar o comportamento do modelo durante os processos de validação e testes.

Como parte do processo de aprendizado, a validação consiste em utilizar o *dataset* de validação para validar algumas alterações no algoritmo de aprendizado com relação à redução da perda. A Figura 5.7 mostra algumas métricas aferidas ao longo do processamento do algoritmo, comparando os resultados obtidos durante treinamento com a validação.

- Perda - Mostra o resultado direto do cálculo da função de perda (entropia cruzada) durante o processo de validação.
- PRC - Mostra a relação entre o valor do PPV (*precision*) e o valor do TPR (*recall*).
- PPV - Mostra a precisão do modelo em sinalizar positivo legítimo, dentro do conjunto de positivos apontados pelo modelo.
- TPR - Mostra a sensibilidade do modelo em apontar um positivo, dentro do conjunto total de resultados positivos legítimos.

O comportamento esperado, que pode ser observado em todos os gráficos da Figura 5.7 é que não ocorre o "descolamento" das curvas de treinamento e validação, mostrando uma tendência de convergência para o modelo. A convergência é um indício de que o modelo irá generalizar para dados novos, que será comprovado no processo de teste a seguir.

Entrando no processo de teste, cujo objetivo é avaliar o desempenho do modelo na predição de dados novos, é possível construir a matriz de confusão geral (Tabela 5.1), considerando agora o *dataset* de teste, ao invés do *dataset* de validação.

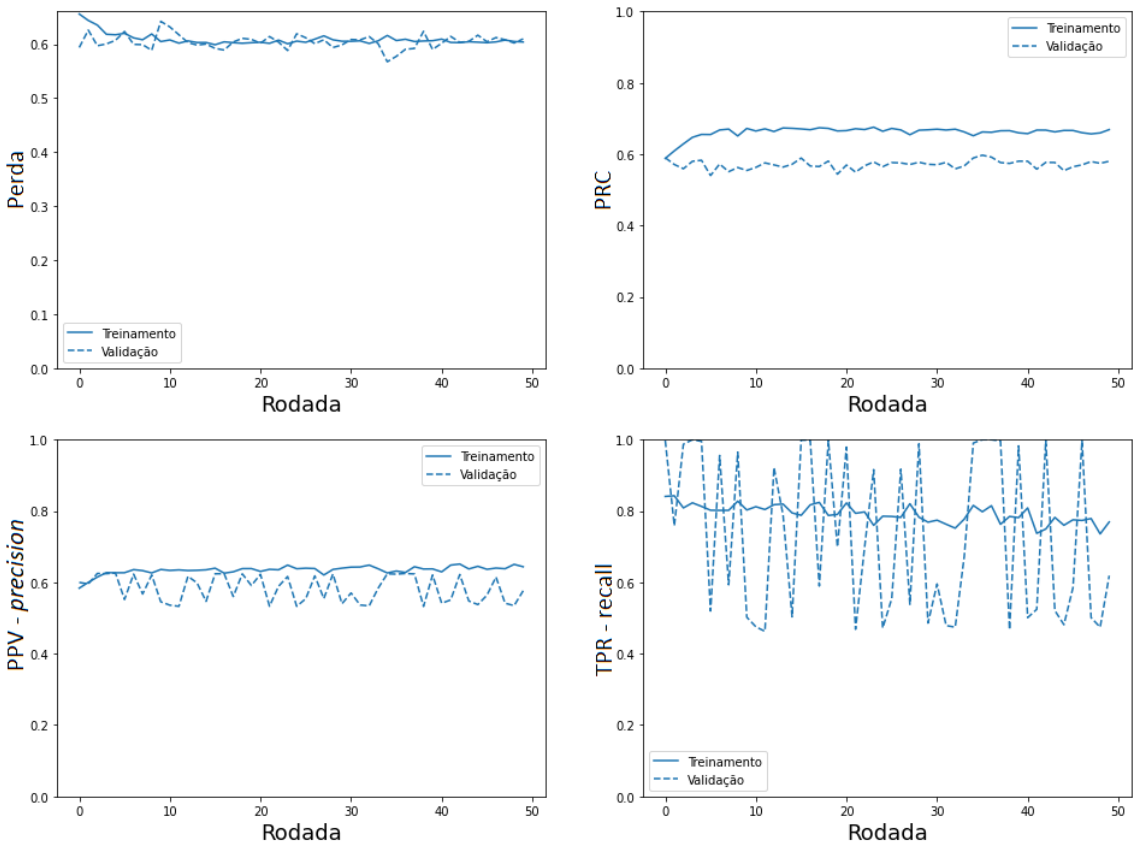


Figura 5.7: Métricas de desempenho utilizadas para aferir os ajustes no modelo de aprendizado durante o processo de validação, classificando os anúncios com $M_C > 0,5$ como verdadeiros e os anúncios com $M_C \leq 0,5$ como falsos.

Tabela 5.1: Matriz de confusão binária com o desempenho de classificação do modelo de aprendizado de máquina escolhido, mostrando sua capacidade de predição de dados novos.

1907		Predição	
		Ataque	Normal
Rótulo	Ataque	1023	30
	Normal	492	362

A partir da matriz de confusão mostrada do modelo mostrada na Tabela 5.1 é possível extrair métricas adicionais para avaliação do desempenho geral do modelo, que estão dispostas na Tabela 5.2 logo abaixo.

Ainda considerando o processo de teste, a Figura 5.8 mostra a relação entre o TRP% x FPR%. Esta relação serve para avaliar o desempenho da predição em que anúncios BGP com $M_C > 0,5$ são classificados como confiáveis (verdadeiros) e por outro lado, anúncios com $M_C \leq 0,5$ são classificados como não-confiáveis (falsos). Quanto mais afastada na parte superior do eixo diagonal (linha vermelha), melhor

Tabela 5.2: Métricas de desempenho de detecção calculadas à partir das matrizes de confusão.

Métrica	**Fórmula**	**Valor**
PPV (*precision*)	$\frac{TP}{TP+FP}$	$0,68$
TPR (*recall*)	$\frac{TP}{TP+FN}$	$0,97$
Acurácia	$\frac{TP+TN}{TN+TP+FN+FP}$	$0,73$
AUC	Área abaixo do ROC	$0,76$
PRC	Relação entre PPV e TPR	$0,79$

é a escolha do limite.

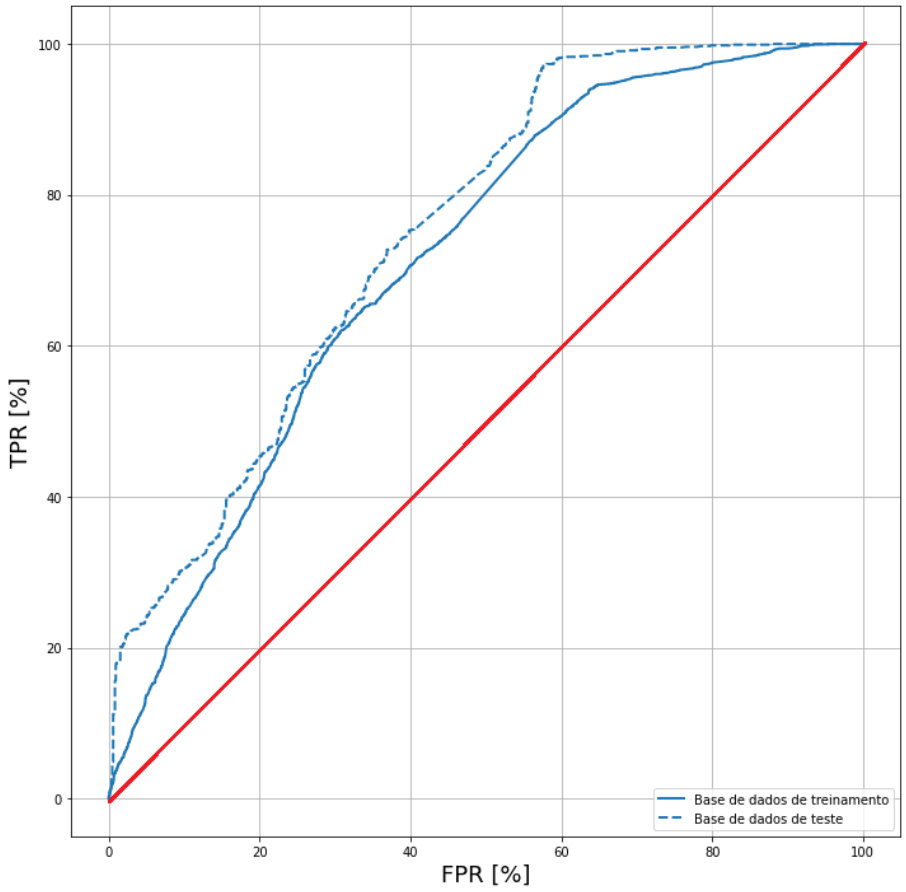


Figura 5.8: Relação gráfica entre o TPR e o FPR – ROC (*Receiver Operating Characteristics*) para mostrar o desempenho do gatilho ($M_C = 0,5$) da função de ativação.

A Figura 5.9 mostra a distribuição das massas de crença $M_C \in [0,1]$ preditas pelo modelo de aprendizado para os dados do *dataset* de teste. Como pode ser

verificado pelas cores azul e laranja no gráfico, a predominância da cruz laranja no alto no gráfico indica que há realmente uma boa sensibilidade na predição (TPR ou *recall*). Também é possível observar que a taxa de anúncios de ataques reais, incorretamente preditas como falsas pela plataforma é baixa. Entretanto, nota-se claramente que há um desempenho não tão bom quando na predição da massa de crença de anúncios falsos, visualizado nas bolas azuis na parte superior do gráfico (FPR). A grande concentração de formas nos extremos inferior e superior do gráfico também mostra que o limite de classificação $M_C = 0,5$ é apropriado.

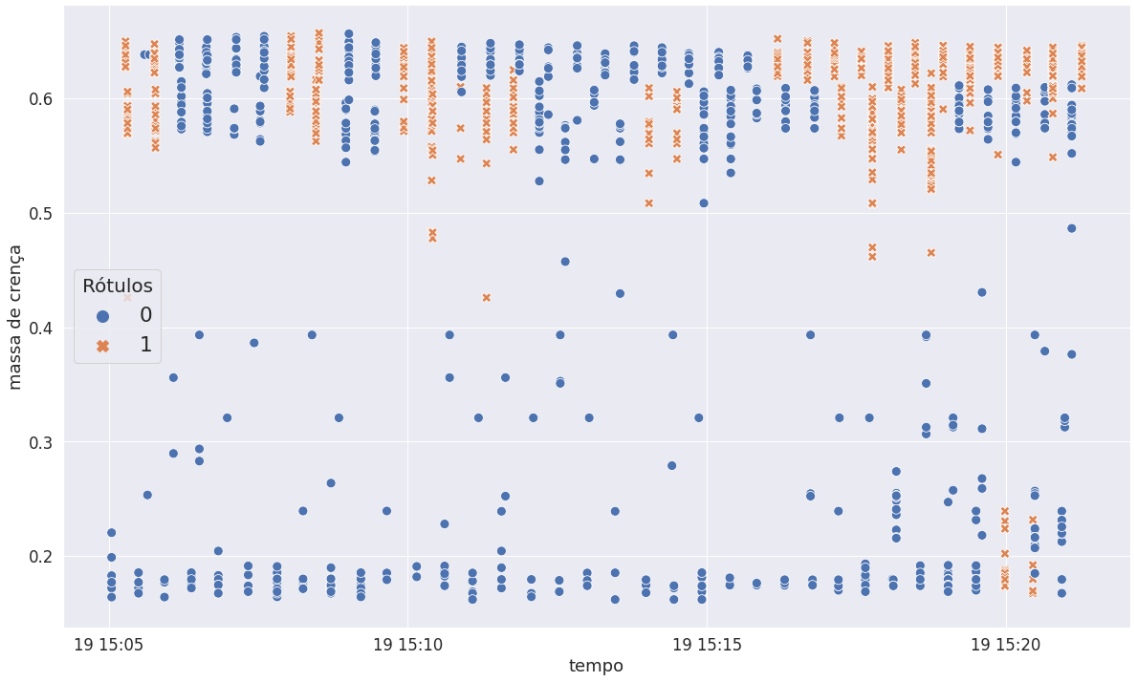


Figura 5.9: Resultado da predição da massa de crença dos anúncios novos (*dataset* de teste) no tempo do conjunto de dados.

Capítulo 6

Sistema de Mitigação de Ataques de Negação de Serviços Contra o Plano de Controle do 5G

O crescente uso dos telefones celulares inteligentes *smartphones* e a grande diversidade de novas aplicações em rede deram um grande impulso para o crescimento da Internet. Atualmente, conforme estatísticas disponibilizadas em [140], cerca de três quartos dos usuários utilizarão somente seus *smartphones* para acessar a Internet em 2025. Graças ao desenvolvimento tecnológico dos dispositivos, as aplicações móveis projetam a geração de 164 EB (ExaBytes) de tráfego por mês em 2025 [141]. De fato, as aplicações móveis têm se tornado cruciais na rotina das pessoas, e a sua larga utilização alavanca o desenvolvimento de novas aplicações inovadoras todos os dias [142]. Entretanto, o crescimento contínuo do número de dispositivos móveis e a diversidade de novas aplicações permanentemente conectadas requerem cada vez mais capacidade da rede para escoar todo este tráfego.

As redes 5G já nascem com a necessidade de superar grandes desafios relacionados principalmente com o aumento da banda de dados, reduzindo a latência fim a fim e a redução do custo operacional. Para superar estes desafios e se viabilizar como a próxima geração das redes de comunicação, o desenvolvimento do 5G conta com algumas tecnologias inovadoras, tais como: redes auto-organizáveis (*Self-organising Networks* - SON), antenas inteligentes MIMO (*Multiple-Input/Multiple-Output*) e hiper-densificação de estações rádio-base (*Small-cells*) [143, 144]. Desta forma, para lidar com toda esta diversidade de tecnologias e fazê-las operar como uma "orquestra", o plano de controle do 5G também precisou evoluir para uma estrutura distribuída e virtualizada, capaz de comportar múltiplos inquilinos (*multi-tenants*).

Todo este ambiente inovativo e de alto consumo de dados também traz consigo consequências e preocupações. A evolução da Internet e o desenvolvimento de novas

tecnologias também desencadeou um aumento bastante significativo no número de ataques cibernéticos. Estatísticas recentes levantadas em [145] mostram que o número de ataques cresceu aproximadamente em 50% desde 2017. Independentemente das suas motivações ou seus objetivos, os atacantes cibernéticos enxergam o novo ecossistema tecnológico do 5G, incluindo poderosos *smartphones*, alta banda de dados e infraestrutura baseada em *software*, como uma excelente oportunidade para ampliar a diversidade de ataques. Apesar de contar com mecanismos de proteção melhorados em relação ao seu antecessor LTE, a arquitetura de segurança do 5G ainda apresenta algumas vulnerabilidades, principalmente relacionadas aos ataques de sinalização [146].

Os pilares da estrutura de segurança do 5G estão baseados na arquitetura do LTE, onde o controle emana principalmente do núcleo para a rede de acesso (RAN - *Radio Access Network*) [147]. Ou seja, a arquitetura de segurança 5G segue a mesma premissa de controle do LTE, em que o móvel na maioria das vezes apenas responde às requisições de sinalização do núcleo, conhecido como *Evolved Packet Core* (EPC). Embora isto soe como algo antigo, a estratégia de controle centralizado apresenta uma série de vantagens em relação ao controle distribuído. Uma das principais vantagens é justamente de preservar a integridade do plano de controle através da padronização da sinalização, habilitando decisões de alto nível relacionadas com a mobilidade, escalonamento e proteção.

Embora a abordagem centralizada da arquitetura 5G ajuda na proteção do plano de controle prevenindo ataques internos, o elevado número de mensagens trocadas no núcleo para processar as transações de sinalização também propicia a exploração de ataques de amplificação [148]. Desempenhando a função de *gateway* de sinalização entre a rede de acesso (RAN) e o plano de controle (EPC), o MME (*Mobility Management Entity*) filtra as mensagens de sinalização que provêm da rede de acesso. Entretanto, apesar da existência de muitos trabalhos acadêmicos endereçando a função de proteção do MME [93, 95, 96, 149–154], ainda existem muitas questões relacionadas a como detectar e mitigar os ataques de negação de serviços de sinalização sem comprometer o tráfego legítimo.

Este trabalho de doutorado se baseia o cenário ameaçador descrito em [49, 50], onde atacante externo assume o controle de um certo número de *smartphones* e inicia um ataque com o objetivo congestionar o plano de controle, para propor um sistema de mitigação de ataques de DDoS de sinalização. Entretanto, ao invés de implementar medidas duras de bloqueio do tráfego atacante, que acabam por também afetar o tráfego legítimo, propõe-se uma abordagem baseada na estratégia *defense-in-depth*. Neste tipo de estratégia, já mencionada no Capítulo 1, as medidas de defesa funcionam em camadas complementares, com o objetivo de “enfraquecer" ou “frear" o ataque. Utilizando o mesmo conceito das companhias de seguros, propõe-se

a utilização de recursos extra para absorver imediatamente todo o tráfego entrante, como uma primeira linha de defesa para mitigar os efeitos do ataque. O objetivo desta estratégia é de permitir ao sistema de defesa um tempo crucial para estudar o ataque e implementar medidas mais efetivas, sem contudo comprometer os serviços do plano de controle.

Para avaliar a efetividade e a viabilidade da abordagem proposta, é apresentado um ambiente de testes composto por um protótipo de EPC virtualizado (vEPC) e uma rede de acesso (RAN), contendo três estações rádio-base LTE (*enodeB*) com seus *smarthphones* associados (UE). Os resultados dos testes mostram que o gerenciamento do fator de peso (*weight factor*) do vMME funciona de acordo com a especificação [51] para balancear o tráfego da rede de acesso entre os múltiplos vMMEs. Mais especificamente, quando o número de vMMEs é duplicado para balancear o tráfego de sinalização, observa-se uma redução proporcional na ocupação de memória dos equipamentos, sem comprometer o tráfego legítimo da rede. Um modelo analítico baseado em Teoria de Filas é proposto para extrapolar o número de *bots*[1] dos testes no protótipo, a fim de projetar os efeitos de negação de serviços do ataque de sinalização.

6.1 Arquitetura Proposta

A diversidade dos serviços de rede, alguns deles com pré-requisitos rigorosos de altíssima disponibilidade e latência ultra-baixa (*ultra-reliable low-latency communication* - URLLC), motivaram o desenvolvimento do 5G como um sistema totalmente convergente, com um núcleo virtualizado, capaz de agregar diferentes tecnologias de acesso. Para controlar um ambiente tão complexo e heterogêneo, o plano de controle 5G precisa ser capaz de processa mais rapidamente uma verdadeira avalanche de mensagens, ainda maior que seu predecessor LTE [155]. Entretanto, ao mesmo tempo que uma eventual inundação de mensagens representa uma vulnerabilidade de segurança, que coloca em risco principalmente a disponibilidade dos serviços do plano de controle 5G, um dos seus principais pilares de desenvolvimento, a virtualização, também permite o surgimento de novas ideias de mitigação, sem comprometer o tráfego legítimo.

A estratégia de defesa proposta neste trabalho de doutorado, referenciada no trabalho apresentado em [45], tem por objetivo construir uma frente de defesa, capaz de garantir a disponibilidade dos serviços do plano de controle durante algum tempo, durante o ataque de DDoS de sinalização. A arquitetura do sistema se baseia em 4 colunas principais para (*i*) criar um mecanismo inteligente de balanceamento de

[1] O *bots* podem ser entendidos como agentes distribuídos que atacam, sob comando de um mestre.

tráfego de sinalização, capaz de garantir o funcionamento pleno do plano de controle durante a inundação e (*ii*) potencialmente mitigar o ataque, aumentando a relação custo/benefício do mesmo na visão do agressor. São eles:

- Um sistema de detecção de intrusões capaz de detectar anomalias de tráfego diversas e distribuídas.
- Um plano de controle totalmente virtualizado, que permite a rápida instanciação de funções de rede encadeadas.
- O mecanismo de balanceamento de carga do 5G para controlar o tráfego da rede de acesso ao plano de controle.
- A frustração do atacante, que monitorando o ataque, não vê os resultados esperados.

A Figura 6.1 mostra a arquitetura que suporta o sistema de mitigação proposto.

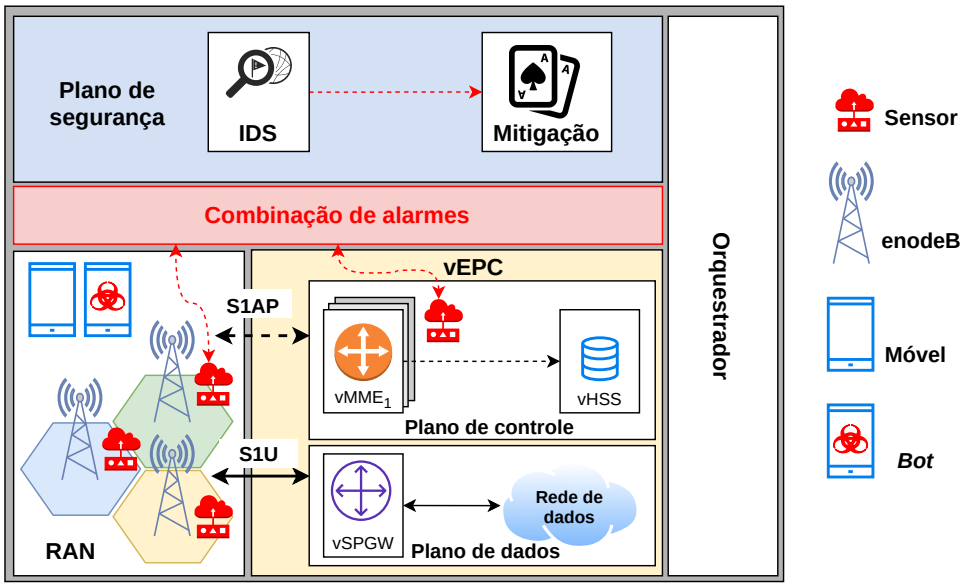


Figura 6.1: Arquitetura de mitigação baseada nas 4 colunas: IDS, virtualização do plano de controle , balanceamento de carga e frustração do atacante.

Conforme especificação 3GPP [156], todo o controle de tráfego de sinalização oriundo da rede de acesso é recebido no vMME. Dentro do plano de controle, o vHSS (*virtual Home Subscriber Server*) é responsável em responder às consultas dos vMMEs com as informações dos clientes. O vMME também comanda o SPGW (*virtual Serving Packet Gateway*) para configurar um canal lógico com o equipamento do usuário. Após o estabelecimento do canal lógico, o equipamento do usuário se conecta ao plano de dados através da interface S1U para trafegar normalmente.

Geralmente, um equipamento de usuário não comprometido (legítimo) tem um comportamento de sinalização normal, requisitando serviços do plano de controle

de tempos em tempos. Por exemplo, um equipamento íntegro se conecta ou se desconecta da rede poucas vezes por dia. Entretanto, no caso de um ataque de sinalização, um grupo de equipamentos de usuário corrompidos (*bots*) podem ser remotamente controlados remotamente para requisitarem sintética e repetidamente serviços do plano de controle como conexão/desconexão (*attach/dettach requests*) ou *handover* [157].

No cenário de ataque assumido, após ser comandado pelo atacante, cada *bot* inicia um ataque individual de negação de serviços contra o plano de controle. O baixo volume de tráfego de sinalização e a sua essência furtiva e dissimilada dificultam a detecção do ataque por um sistema de detecção qualquer, localizado na própria enodeB. Entretanto, um sistema IDS central localizado dentro do plano de controle pode sinalizar a ocorrência de um ataque coordenado de negação de serviços a partir da combinação das evidências enviadas pelas camadas inferiores [76]. Outros sistemas voltados para a detecção de intrusões contra o plano de controle podem ser verificadas em [158–160].

Uma vez detectado a anomalia, o IDS mede a intensidade do tráfego e dispara um comando para o sistema de mitigação para iniciar o jogo[2]. O mecanismo de defesa começa o jogo avaliando o número de vMMEs que estão em funcionamento dentro do EPC e a intensidade de tráfego de cada um deles. Tendo apenas um vMME naquele momento, o mecanismo envia uma ordem ao orquestrador para imediatamente instanciar um noco vMME, independentemente da intensidade do tráfego. Esta primeira jogada tem dois principais objetivos: criar um vMME reserva para prevenir a interrupção dos serviços e causar frustração ao atacante, reduzindo a relação custo/benefício do ataque. No caso de já ter mais de um vMME em funcionamento no momento do ataque, o mecanismo de defesa avalia a intensidade do tráfego de sinalização e aguarda a próxima jogada do atacante, antes de incrementar o número de vMMEs. Se a intensidade de tráfego não aumentar ou diminuir, o mecanismo de defesa dispara novamente o orquestrador para desligar um dos vMMEs em funcionamento. Caso contrário, se o tráfego suspeito continuar aumentando, o mecanismo de defesa segue adicionando vMMEs e balanceamento o tráfego entre eles. O diagrama de mostrado na Figura 6.2 ilustra o processo descrito.

O orquestrador opera verticalmente, fazendo interface com todas as camadas da arquitetura, inclusive com o plano de controle. Sua responsabilidade no jogo de mitigação é adicionar ou remover vMMEs do plano de controle, de acordo com as instruções recebidas do plano de segurança. Os vMMEs recém instanciados recebem um fator de peso maior para atrair para si as novas requisições de conexão vindas da rede de acesso, tanto legítimas quanto provenientes dos *bots*, prevenindo uma

[2]O termo "jogo" aqui se refere ao processo de análise do comportamento do atacante para definição da estratégia de escalonamento

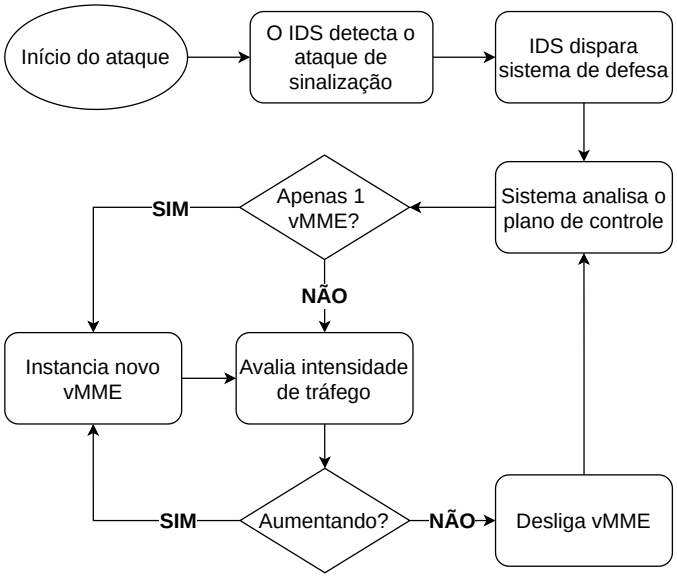


Figura 6.2: Lógica do sistema de defesa, baseado no escalonamento de vMMEs dentro do EPC.

eminente perda de disponibilidade do plano de controle. Também é possível mover tráfego de um vMME para outro recém instanciado, através de um comando de especial (*disable-implicit-detach*) enviado pelo administrador de rede, desde que ambos os vMMEs esteja no mesmo grupo [161].

A interconexão de uma determinada enodeB com múltiplos vMMEs do mesmo grupo é suportada na arquitetura LTE, de acordo com [51]. Entretanto, para viabilizar esta funcionalidade, a rede de comunicação da enobeB deve ser configurada previamente para alcançar a faixa de endereços IP onde os novos vMMEs serão instanciados. O fator de peso de um vMME é representado pelo parâmetro de capacidade relativa (*relative-MME-capacity*), que por sua vez está relacionado com a sua capacidade de processamento, em relação aos outros vMMEs do mesmo grupo. Isto significa que a instanciação de um novo vMME pelo orquestrador deve observar não só a faixa de endereços IP alcançáveis para as enodeBs cujo tráfego pretende atrair, como também a configuração necessária de recursos para suportar o tráfego a ser processado.

6.2 Modelagem Matemática

A modelagem de um ataque de DDoS é uma tarefa desafiadora devido à sua natureza furtiva e a escala necessária para reproduzir os seus efeitos disruptivos, entre outras coisas. Entretanto, alguns cenários de ataque podem ser simulados utilizando modelos analíticos, alimentados com resultados obtidos em testes práticos. Nesta seção serão apresentados dois modelos analíticos baseados em teoria de filas e jogos, com

o objetivo de estudar o comportamento de carga do vEPC nos cenários de ataque e defesa e dos agentes do ataque (atacante e defensor), respectivamente.

A Tabela 6.1 abaixo lista a descrição dos símbolos utilizados nos modelos analíticos propostos.

Tabela 6.1: Lista dos símbolos utilizados nos modelos analíticos da Seção 6.2.

Símb.	**Descrição**	**Símb.**	**Descrição**
n	# total de *smartphones*	N	# de *smartphones* legítimos
M	Número total de *bots*	λ_l	Tx. transações legítimas
λ_a	Tx. transações atacantes	μ	Capacidade de proc. servidor
λ	Tx. total sinalização	q	Tamanho *buffer* servidor
K	Cap. armazenamento EPC	Q	Tamanho da fila
m	# vMMEs (servidores)	S	# max. vMMEs (servidores)
U_d	Função ganho defensor	U_a	Função ganho atacante
C_d	Função custo defensor	C_a	Função custo atacante
S_d	Função estratégia defensor	S_a	Função estratégia atacante

A seguir, segue uma lista de suposições adotadas para tornar o modelo tratável matematicamente, sem comprometer a sua generalidade.

- Um atacante único controla todos os *bots*, que por sua vez enviam a mesma taxa de sinalização λ_a para o vEPC.
- Embora o defensor possa perceber um aumento anormal na taxa de entrada de transações da rede de acesso, ele não sabe a priori se está sob ataque.
- A capacidade de escoamento da rede interna que interconecta todas as entidades do vEPC é infinita.
- A taxa de transações de um único *bot* é pequena o suficiente para não ser detectada individualmente por um IDS, nem pode ser alterada pelo atacante.
- Tanto o atacante quanto o defensor (jogadores) têm a sua disposição recursos finitos. Portanto, como tomadores de decisões, devem considerar constantemente a relação custo/benefício de suas respectivas estratégias de jogo.
- O custo do atacante C_a, relacionado com o seu tempo de trabalho no recrutamento e no lançamento de novos *bots*, aumenta proporcionalmente com o número de *bots* presentes no ataque.
- O custo do defensor C_d, relacionado com a alocação e o gerenciamento de recursos extra para mitigar os efeitos do ataque, aumenta proporcionalmente com o número de vMMEs em operação.
- De alguma forma, o atacante monitora o resultado do seu ataque para estimar a relação custo/benefício do mesmo [162].

6.2.1 Modelagem dos Efeitos do Ataque DDoS na Carga do vEPC

Como mencionado na seção anterior, sua natureza furtiva e o volume fazem da reprodução prática dos efeitos de um ataque de negação de serviços uma difícil tarefa. Em se tratando de sinalização, esta tarefa é ainda mais árdua em função da complexidade e especificidade da estrutura que precisa ser construída. Entretanto, uma vez conhecidos os parâmetros de entrada, é possível projetar os efeitos exaustivos de um ataque de negação de serviços usando modelos analíticos.

Embora a essência disruptiva de qualquer ataque de negação de serviços confronte a premissa ergódica do sistema de filas, o modelo proposto para aproximar os cenários em questão supõe uma eventual operação em regime estacionário, em que o sistema de mitigação atua dinamicamente adicionando recursos, para garantir que o sistema permaneça estável. Utilizando sistemas de filas, como proposto em [97], é possível chegar a um modelo analítico capaz de produzir análises assintóticas aproximadas ao cenário prático, durante um ataque de negação de serviços. Este mesmo modelo pode inclusive fornecer informações importantes sobre o desempenho dos sistemas de proteção empregados para mitigar os efeitos do ataque.

No modelo proposto, as transações de sinalização que entram no vEPC vindas da rede de acesso (RAN) são representadas por usuários, que entram no sistema de filas para serem processados. Uma vez no sistema, os usuários (transações) são processados por m servidores idênticos, cada um com capacidade de processamento igual a μ usuários/segundo, representando os vMMEs em operação dentro do vEPC. Assim, a capacidade total de processamento de usuários (transações) do sistema de filas é $m\mu$ usuários/segundo.

Para cada servidor (vMME) dentro do sistema (vEPC), é alocado um pequeno *buffer* de tamanho q, que armazena os usuários, enquanto os servidores estão ocupados, processando um usuário. A área total de armazenamento do sistema proposto $K = m(q + 1)$, acomoda os usuários que estão sendo correntemente processados pelos servidores (vMMEs) e aqueles que esperam na fila $Q = mq$ durante um tempo T até serem processados.

A carga de usuários oferecida a cada servidor é controlada através de um balanceador, que representa o fator de peso (*weight factor*) de cada vMME, atraindo o tráfego da rede de acesso de acordo com a sua capacidade relativa no grupo (*relative capacity*), vide Seção 6.1. A Figura 6.3 logo abaixo oferece uma visão gráfica do modelo proposto na aproximação.

Os usuários, representando as transações de sinalização provenientes da rede de acesso, chegam ao sistema de filas (o grande retângulo à direita na Figura 6.3) com taxa $\lambda = \lambda_a + \lambda_l$ e intervalo exponenciais entre as chegadas. Os intervalos de

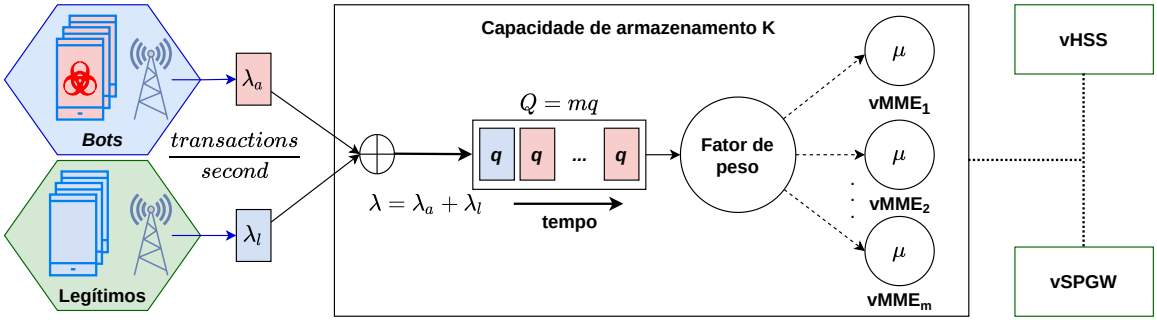


Figura 6.3: Modelo de filas representando o grupo de vMMEs processando as transações vindas da RAN.

tempo que o servidor leva para processar uma transação também são distribuídos exponencialmente, com média μ^{-1}, são independentes mutuamente e independentes do processo de chegada $(M/M/)$. Apesar de não aderir completamente ao cenário do ataque, uma vez que os usuários são gerados deliberadamente a partir de uma fonte comum (o atacante), a suposição de independência entre os eventos e entre os processos viabiliza a utilização de modelos Markovianos de nascimento/morte, como uma aproximação para o cenário de exaustão do vEPC e os efeitos do sistema de mitigação proposto.

Considerando as características operacionais e a arquitetura já descritas anteriormente na Seção 6.1, especialmente dos múltiplos vMMEs (m), do espaço finito no sistema (K) e da população limitada de usuários (M), adotou-se o sistema de filas $M/M/m/K/M$, apresentado em [52] - páginas 108–110, como o mais adequado para modelar o sistema em questão, onde $m \leq K \leq M$. Ou seja, o número de servidores em operação m é sempre menor ou igual ao espaço de armazenamento do sistema K, que por sua vez é sempre menor ou igual à população de usuários M. Assim, se há um usuário sendo processado no sistema, outros $(M-1)$ chegam ao mesmo com uma taxa de $(M-1)\lambda$ usuários/segundo. Além disto, um usuário que chega e já encontra outros K usuários dentro no sistema ou, que aguarda na fila por um tempo superior a T, é descartado, deixando o sistema sem ser processado. A Figura 6.4 abaixo mostra o diagrama de transição de estados do modelo adotado.

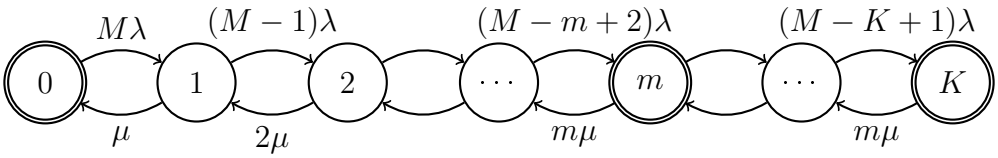


Figura 6.4: Diagrama de transição de estados do modelo $M/M/m/K/M$.

O diagrama de transição de estados da cadeia de Markov finita apresentado na Figura 6.4 mostra os valores possíveis da variável aleatória I, representando o

número de usuários (transações) dentro do sistema num determinado instante. Neste diagrama, os estados inicial ($I = 0$), ($I = m$) e ($I = K$) demarcam duas regiões distintas, de acordo com o comportamento das transições: $0 \leq I \leq (m-1)$ e $m \leq I \leq K$. A partir do diagrama de transição de estados, chega-se nas equações que definem a probabilidade do sistema abrigar i usuários $p_i = \Pr(I = i)$, onde i é a realização da variável aleatória I, considerando as duas regiões mencionadas.

$$p_i = \begin{cases} p_0 \left(\frac{\lambda}{\mu}\right)^i \binom{M}{i} & ; \text{para } 0 \leq i \leq (m-1) \\ p_0 \left(\frac{\lambda}{\mu}\right)^i \binom{M}{i} \frac{i!}{m!} m^{m-i} & ; \text{para } m \leq i \leq K \end{cases} \tag{6.1}$$

Onde a probabilidade do sistema estar vazio $p_0 = \Pr(I = 0)$ é calculada a partir da relação $\sum_{i=0}^{K} p_i = 1$.

$$p_0 = \left[\sum_{i=0}^{m-1} \left(\frac{\lambda}{\mu}\right)^i \binom{M}{i} + \sum_{i=m}^{K} \left(\frac{\lambda}{\mu}\right)^i \binom{M}{i} \frac{i!}{m!} m^{m-i} \right]^{-1} \tag{6.2}$$

Considerando o caso especial, quando o número de usuários no sistema i já preenche todo o espaço do mesmo K, é possível calcular a probabilidade de bloqueio $p_B = \Pr(I = K)$.

6.2.2 Modelagem Comportamental Durante o Ataque

A Teoria de Jogos pode ser entendida como uma estrutura matemática usada para modelar interações estratégicas entre agentes racionais (ou grupo de agentes) tomadores de decisões (jogadores), onde as ações de um jogador têm relevância e efetivamente interferem nas decisões dos outros jogadores. O jogo em si, é a representação completa do conjunto de regras e/ou condições de contorno, às quais os jogadores estão submetidos, e das interações entre os jogadores, sejam estas cooperativas ou não-cooperativas [53]. Particularmente, o conceito de Teoria dos Jogos tem sido largamente empregado na área de segurança de redes para projetar as estratégias dos jogadores (atacante e defensor), durante o evento de um ataque. Esta abordagem permite projetar as ações futuras dos jogadores, com base nos resultados previstos analiticamente [163–169]. Graças à sua capacidade de modelar situações de escolha condicionada (*what-if*), o uso da Teoria de Jogos para modelar ataques de DDoS de sinalização requer alguns ajustes especiais, pois, ao invés de atentar diretamente contra o seu alvo, o atacante se beneficia de conhecimentos previamente adquiridos sobre o alvo para fazer com que ele próprio efetive os objetivos do ataque, neste caso a exaustão.

O modelo proposto nesta seção assume um jogo não-cooperativo de dois jogadores, onde o atacante e o defensor competem entre si para conseguirem seu melhor

resultado, relacionado com o impacto disruptivo do ataque de sinalização sobre os usuários do plano de controle da plataforma 5G. Neste contexto, define-se U_a como a função de recompensa do atacante, diretamente proporcional ao número de usuários impactados pelo ataque e inversamente proporcional ao esforço do atacante para lançar o ataque. Da mesma forma, define-se U_d como a função de recompensa do defensor, diretamente proporcional ao número de usuários utilizando o sistema normalmente e inversamente proporcional aos recursos utilizados pelo defensor na instanciação de novos vMMEs. Tanto o atacante quanto o defensor tem apenas duas estratégias: atacar ou não atacar (S_a), no caso do atacante e defender ou não defender (S_d), no caso do defensor. O objetivo do atacante no jogo é lançar um ataque efetivo, que cause o maior dano possível, consumindo o mínimo de recursos (menor custo). Neste objetivo, o atacante pode aumentar ou reduzir a intensidade do ataque, de acordo com o número de *bots* recrutados pelo mesmo. Por outro lado, o desafio do defensor é manter a máxima disponibilidade dos serviços do plano de controle, gastando o mínimo de recursos possível. Para cumprir este desafio, o defensor pode aumentar ou reduzir o número de vMMEs em operação no vEPC, considerando os recursos disponíveis no seu *datacenter*.

$$Jogo{=}G(S_a{,}S_d{,}U_a{,}U_d)$$

Baseado na abordagem matemática descrita em [167], o modelo proposto considera uma partida com uma única jogada por vez para ambos os jogadores. Isto é, após escolherem suas respectivas estratégias, tanto o atacante quanto o defensor decide seu movimento e jogam ao mesmo tempo.

Modelando as Funções de Ganho

Assumindo $\mathcal{N}(\gamma{,}\sigma^2)$ uma variável aleatória normal com média γ e variância σ^2. Pode-se supor também um certo número de usuários normais $N{\leq}n$ gerando uma mesma taxa de sinalização (transações/segundo) com taxa λ_l para o vEPC.

É possível modelar a distribuição de probabilidade da taxa de sinalização dos usuários normais escolhendo $N{\leq}n$ amostras de variável aleatória com distribuição normal $X_i{=}\mathcal{N}(\lambda_l{,}\sigma_l^2)$, $i{=}(1{,}2{,}\cdots{,}N)$. Também é válida a igualdade $X_l{=}\sum_{i=1}^{N}X_i{\sim}$ $\mathcal{N}(N\lambda_l{,}N\sigma_l^2)$, representando a distribuição de probabilidade da taxa total de transações de sinalização de usuários íntegros que chegam aos vMMEs.

No caso de um ataque coordenado envolvendo $M{\leq}n$ usuários comprometidos, também é possível estabelecer $X_a{\sim}\mathcal{N}(M\lambda_a{,}M\sigma_a^2)$ como a distribuição de probabilidade da taxa total de transações de sinalização, com média $M\lambda_a$ e variância $M\sigma_a^2$. Desta forma, a distribuição de probabilidade da taxa tola de sinalização, considerando o tráfego legítimo e comprometido é $X_t{=}X_l{+}X_a$.

Continuando com a modelagem Gaussiana, considere $\alpha{=}\mu/X_t$ como a fração de transações de sinalização que será descartada por um vMME sobrecarregado, quando $X_t{>}\mu$ e β como a taxa de sinalização mínima de um usuário legítimo.

$$\alpha{=}\frac{\mu}{(N\lambda_l{+}M\lambda_a)} \tag{6.3}$$

O número de médio de usuários legítimos registrados ao plano de controle 5G, cujas transações de sinalização serão processadas normalmente sem perdas pelo vEPC pode ser escrito como:

$$n_p{=}N{\times}\Pr\left[X_i{>}\frac{\beta}{\alpha}\right] \tag{6.4}$$

A partir da Equação 6.4, a fração de usuários legítimos que serão prejudicados, tendo suas transações de sinalização descartadas pelo plano de controle sobrecarregado, também pode ser calculado como:

$$D{=}\frac{N{-}n_p}{N} \tag{6.5}$$

Combinando as Equações 6.3 e 6.4 com a Equação 6.5, tem-se:

$$D{=}\Pr\left[X_i{<}\frac{\beta(N\lambda_l{+}M\lambda_a)}{m\mu}\right] \tag{6.6}$$

Na Equação 6.6, $m{\leq}S$ é o número de vMMEs em operação dentro do vEPC e S é o número total de vMMEs que podem ser instanciados dentro do vEPC, considerando a capacidade total disponível no *datacenter*.

A recompensa final do atacante U_a está relacionada com uma recompensa parcial u_a, referente às perdas de sinalização impostas ao maior número possível de usuários legítimos u_a, e ao custo do seu esforço c_a em recrutar os *bots* e lançar o ataque. Lembrando que N é o número de usuários legítimos, isto é, que não fazem parte do grupo de *bots*, a recompensa parcial pode ser estimada a partir da Equação 6.6, da seguinte forma:

$$u_a{=}N{\times}D \tag{6.7}$$

A função de custo do atacante deve considerar o custo em recrutar o número suficiente de *bots* M e ao esforço em lançar o seu ataque, coordenando os M *bots* recrutados dentro do grupo total de usuários n. Neste caso, é assumido um crescimento exponencial em relação ao número de *bots* em virtude do custo para recrutar e controlar os mesmos [170].

$$c_a{=}\frac{M^2}{n} \tag{6.8}$$

Juntando as Equações 6.7 e 6.8, a função de recompensa final do atacante é representada pela seguinte expressão:

$$U_a \text{ (usuários/sec.)} = u_a - c_a = ND - \left(\frac{M^2}{n}\right) \tag{6.9}$$

Da mesma forma, é possível estimar a recompensa do defensor U_d do ganho parcial do defensor em manter o funcionamento normal dos usuários u_d, descontando-se o custo do mesmo em instanciar recursos disponíveis dentro do *datacenter* para escalonar o número de vMMEs c_d, que podem ser calculados como:

$$u_d = N(1-D) \tag{6.10}$$

$$c_d = m\frac{\mu}{S} + (S-m)\frac{\mu}{n} \tag{6.11}$$

Na Equação 6.11, o primeiro fator da soma se refere ao custo de se instanciar m vMMEs, cada um com capacidade idêntica de processamento μ, dentro de um total de S possíveis. O segundo fator representa o custo em manter estes recursos ociosos dentro do *datacenter* para serem utilizados como proteção em caso de ataque.

Assim, juntando-se as Equações 6.10 e 6.11, chega-se à recompensa final do defensor.

$$U_d \text{ (usuários/sec.)} = N(1-D) - m\frac{\mu}{S} - (S-m)\frac{\mu}{n} \tag{6.12}$$

As recompensas do atacante e do defensor definidas nas Equações 6.9 e 6.12 são respectivamente parametrizadas na tabela estática de jogo não-cooperativo (G), como mostrado na Tabela 6.2, logo abaixo.

Tabela 6.2: Tabela de estratégia de jogo, demarcando estratégia e recompensas do atacante e do defensor.

		Defensor	
		não defender	defender
Atacante	não atacar	(0,0)	$(c_d, -c_d)$
	atacar	$(U_a, -U_a)$	$(c_d - c_a, U_d + c_a)$

A primeira célula da Tabela 6.2 (*não atacar/não defender*) define o estado ocioso de ambos os jogadores, com recompensa nula. O estado *não atacar/defender* é simétrico e mostra o custo do defensor (c_d) em implementar esforço de defesa sem necessidade aparente. Este mesmo custo do defensor também é creditado como recompensa do atacante, uma vez que este recebe um benefício eventual em conhecer a estratégia de defesa, antes mesmo de qualquer tentativa de ataque. O estado *não atacar/defender* simula uma possível deficiência no desempenho do sistema de

detecção de intrusões do lado da defesa, no quesito taxa de alarmes falso-positivos. O estado *atacar/não defender* é possivelmente o estado mais crítico, onde há um ataque em curso, mas não existem medidas de defesa implementadas. Neste estado simétrico, a recompensa do atacante, em relação ao número de usuários com perdas de sinalização, é o próprio custo do defensor. O último estado *atacar/defender* marca o início do jogo de mitigação, onde ambos os jogadores desprendem esforços para desempenhar as suas respectivas estratégias, buscando a máxima recompensa. No caso do atacante, sua recompensa é proporcional à diferença entre o preço pago pelo defensor em manter a disponibilidade do sistema e o seu próprio custo com os *bots*. Do lado do defensor, assumindo que é bem-sucedido no jogo de mitigação, sua recompensa é a normalidade dos serviços dos usuários, descontando-se o preço que é pago para isto.

O estado de equilíbrio deste jogo, conhecido como "Equilíbrio de Nash" [100], define o ponto onde os jogadores se encontram ambos num estado estacionário, conformados com os resultados obtidos. Em outras palavras, uma vez atingido o equilíbrio, nem o atacante nem o defensor possui motivação para desprenderem esforços ou alterarem suas respectivas estratégias.

A próxima seção 6.3 descreve um modelo de experimental, cujos resultados obtidos serão utilizados como parâmetros de entrada nos modelos propostos nesta seção.

6.3 Modelo Experimental

O experimento apresentado nesta seção reproduz um cenário hipotético, em que um atacante explora a transação de solicitação de registro dos usuários (*attach request*) como vetor para um ataque de sinalização. Neste tipo de ataque, o agressor controla um grupo de 6 *bots*, previamente recrutado pelo mesmo, para emitirem de forma coordenada e síncrona as solicitações de registro na rede de acesso (*attach request*). O grupo de *bots* é estrategicamente distribuído em 3 diferentes estações rádio-base (*small-cells*), de forma que a tentativa de ataque não possa ser detectada por qualquer sistema de detecção individual. Importante observar que, embora tenha sido escolhida a transação de solicitação de registro como vetor, o atacante poderia escolher qualquer outra transação que dispara um grande número de mensagens dentro do plano de controle. Neste caso, além de ser relativamente fácil de se reproduzir, o cenário escolhido está bem endereçado em [152, 171].

Reproduzir o cenário prático de inundação de mensagens devido a um ataque de sinalização baseado na transação de solicitação de registro (*attach request*) está além dos objetivos propostos neste experimento. O experimento proposto visa sobretudo avaliar a carga dos vMMEs ao processar estas transações e demostrar o funcionamento do sistema de controle de sobrecarga do LTE para distribuir a carga entre os

vMMEs sob um suposto ataque distribuído de negação de serviços de sinalização. Os resultados do experimento serão extrapolados através dos modelos analíticos propostos nas Seções 6.2.1 e 6.2.2 e assim chegar na aproximação desejada do cenário de ataque.

O mecanismo de balanceamento de carga do LTE é baseado no controle do fator de peso do vMME através de alterações no parâmetro de capacidade relativa do mesmo. o parâmetro de capacidade relativa, por sua vez, está relacionada com capacidade de processamento de um vMME em relação aos outros do mesmo grupo. O vMME com maior capacidade relativa tem o maior fator de peso e, portanto, prevalece em relação aos outros, sempre que uma nova solicitação de registro for emitida pela RAN. Assim, dentro do experimento, o novo vMME será aprovisionado com um valor maior de capacidade relativa, para atrair o tráfego de sinalização e dividir a carga total.

Para demonstrar o mecanismo de balanceamento de carga, foi criado um ambiente de testes composto por um EPC virtualizado (vEPC) e uma plataforma de rede de acesso E-UTRAN (*universal terrestrial radio access network platform*). A configuração do vEPC usa a versão 0.5.0 do software Openairinterface [172], que está instalado em 4 diferentes máquinas virtuais (VMs), hospedadas dentro de uma infraestrutura Openstack [173]. As máquinas virtuais 1 e 2 (VM_1 e VM_2) emulam os $vMME_1$ e $vMME_2$ respectivamente, enquando que as VM_3 e VM_4 emulam o vHSS e o vSPGW. A configuração da plataforma E-UTRAN se distribui em 3 diferentes VMs, hospedadas na mesma infraestrutura Openstack. Em cada uma destas VMs funciona um simulador Openairinterface OAISIM, utilizado para simular juntos o par enodeB + equipamento(s) do(s) usuário(s) (EU). A VM_5 é composta de 1 enodeB e apenas EU, a VM_6 opera com 1 enodeB e 2 EUs e a VM_7 comporta 1 enodeB e 3 EUs. A Figura 6.5 a seguir ilustra a composição lógica do experimento dentro da nuvem Openstack.

As máquinas virtuais que hospedam os simuladores OAISIM ($OAISIM_1$, $OAISIM_2$ e $OAISIM_3$), equipadas com 2 CPUs virtuais e 30GB de armazenamento em disco, operam com o sistema operacional Ubuntu 14.4 com núcleo (*kernel*) de baixa latência. As máquinas virtuais que hospedam as entidades do vEPC ($vMME_1$, $vMME_2$, vHSS e vSPGW) utilizam a mesma plataforma de recursos abstraídos, mas operam com sistema operacional Ubuntu 16.5, com núcleo (*kernel*) 4.7. As enobeBs que funcional dentro das máquinas $AISIM_1$, $OAISIM_2$ e $OAISIM_3$ (10.68.34.100, 10.68.34.102 e 10.68.34.208) estão conectadas simultaneamente com ambos os vMMEs IPs 10.68.34.14 e 10.68.34.114, permitindo que possam se conectar ao vMME com maior fator de peso, de acordo com sua capacidade relativa. Embora os $vMME_1$ e $vMME_2$ possuam identidades diferentes, ambos estão no mesmo grupo.

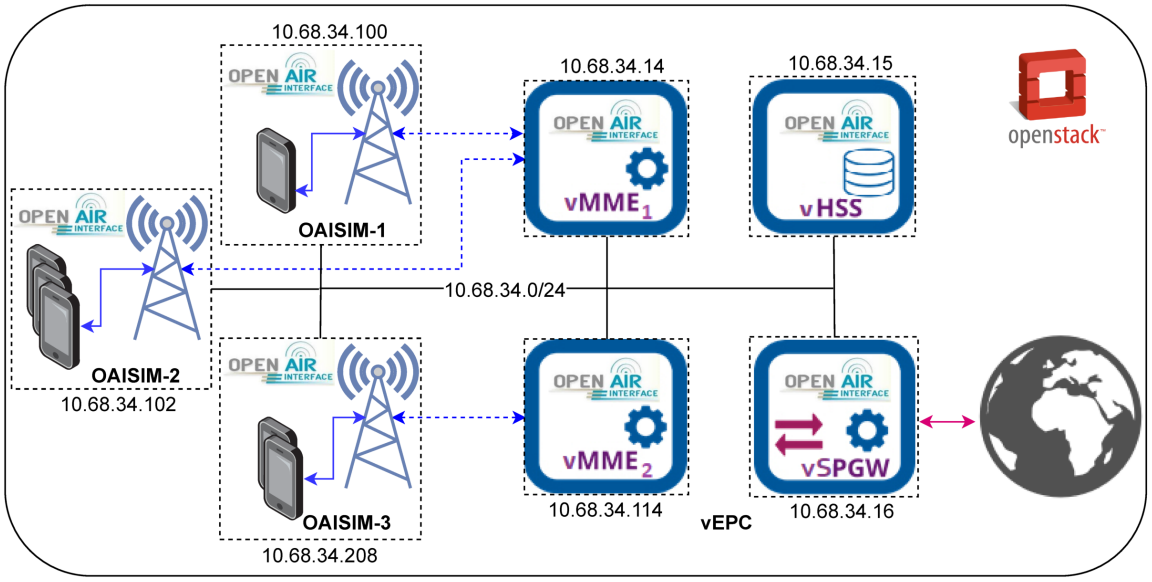


Figura 6.5: Configuração do experimento utilizado para testar as conexões entre a RAN e os vMMEs, de acordo com o fator de peso de cada um deles. A mesma nuvem Openstack também hospeda as demais VMs (vHSS, vSPGW e OAISIM) que compõem o vEPC e a plataforma E-UTRAN.

6.3.1 Roteiro do Experimento

Na primeira fase do experimento, dois agentes monitoram o consumo de memória da $vMME_1$ que serve de *gateway* de sinalização para todos os simuladores (OAISIM), incluindo os 6 EUs atrelados aos mesmos, a cada 0,5 segundo. A partir deste ambiente, são estabelecidas conexões de sinalização entre as enodeBs dos simuladores e o vEPC, através do $vMME_1$, que está configurado com capacidade relativa igual a 10.

Durante os testes iniciais, verificou-se que o parâmetro de utilização de memória das VMs era o que mais apresentava sensibilidade em relação à carga trazida pelas solicitações de registro dos EUs. Este comportamento era esperado, considerando o processamento baseado em estados (*state-full*) das funções MME que funcionam dentro das VMs.

A próxima fase do roteiro do experimento teve por objetivo demonstrar o nível de balanceamento de carga em função da operação do segundo vMME ($vMME_2$), atraindo para si as novas requisições de registro das enodeBs. Uma vez estabelecida a conexão de sinalização entre os $OAISIM_1$ e $OAISIM_2$ com o $vMME_1$, o $vMME_2$ foi iniciado com capacidade relativa configurada para 20. Depois disto, o $OAISIM_3$ foi então iniciado, junto com os seus 3 EUs. Como esperado, observou-se que todos os 3 EUs do $OAISIM_3$ se conectaram ao $vMME_2$, ao invés de se conectarem ao $vMME_1$. Durante todo este processo, o agente de monitoração continuou gravando os valores de utilização de memória dos 2 vMMEs.

A Figura 6.6 mostra graficamente a cronologia do experimento, considerando as

suas duas fases.

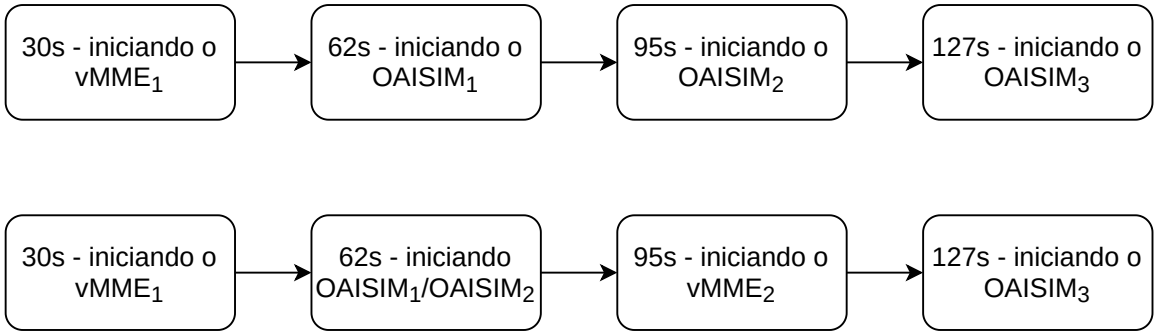


Figura 6.6: Cronologia da execução do experimento, dividido em duas fases.

6.3.2 Resultados do Experimento

Os resultados obtidos no experimento mostram que o mecanismo de balanceamento baseado no fator de peso dos vMMEs funcionam como esperado. De acordo com o roteiro do experimento, no lançamento da VM OAISIM_3, os 3 EUs no simulador foram todos registrados no vMME_2, que possui o maior fator de peso, ao invés do vMME_1. Depois de registrados, todo o tráfego de sinalização decorrente será trocado com o vMME_2, evitando sobrecarregar o vMME_1 com o tráfego da nova enodeB.

Outro ponto que deve ser enfatizado é sobre a possibilidade adicional de remanejar carga diretamente do vMME_1 para o vMME_2 através do comando de *overload start* para as enodeBs funcionando nos simuladores OAISIM_1 e OAISIM_2. Assumindo que ambas as enodeBs têm conectividade plena com o vMME_2, todos os EUs serão desconectados no vMME_1 e se reconectarão novamente no vMME_2.

A Figura 6.7 mostra o perfil de utilização de memória da primeira fase do roteiro, em que o vMME_1 está sozinho e processa toda a carga de sinalização da RAN.

A Figura 6.8 corresponde à segunda fase do roteiro e mostra o perfil de utilização de memória de ambos os vMMEs. Após 96 segundos e já com o vMME_2 em funcionamento é disparado o lançamento do simulador OAISIM_3.

Comparando os perfis nas Figuras 6.7 e 6.8, percebe-se que o vMME_2 consegue absorver algo em torno de 20% da carga (utilização de memória), que de outra forma seria totalmente assumida pelo vMME_1. Isto demonstra a efetividade da abordagem, balanceando a carga entre os vMMEs e garantindo a disponibilidade do plano de controle 5G.

Os resultados numéricos mostrados nas Figuras 6.7 e 6.8 darão origens a parâmetros específicos que serão extrapolados nos modelos analíticos propostos nas Seções 6.2.1 e 6.2.2 para aproximar o cenário disruptivos causado por um ataque de DDoS contra o plano de controle.

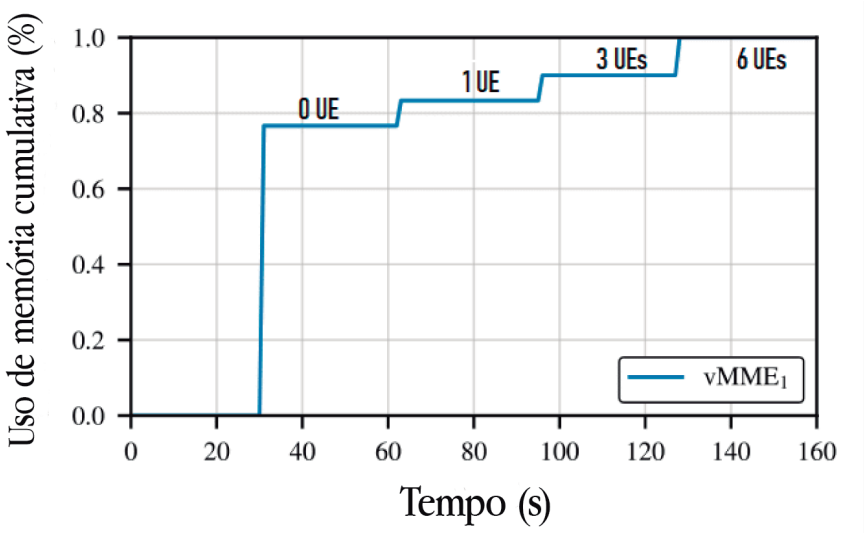


Figura 6.7: Perfil de uso de memória cumulativo do $vMME_1$ quando este processa todo o tráfego de sinalização da RAN (6 EUs). A utilização plena cumulativa (100%) corresponde a 3% em número absoluto.

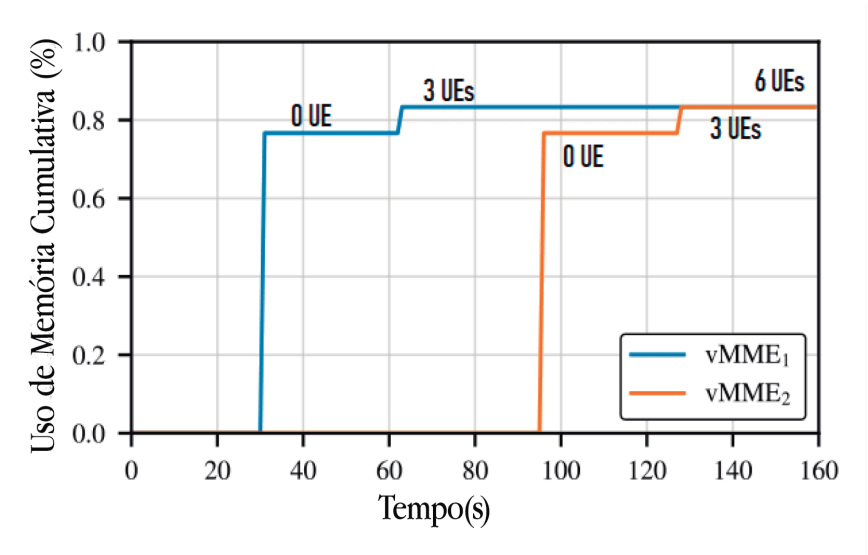


Figura 6.8: Perfil de uso de memória cumulativo dos $vMME_1$ e $vMME_2$ quando este último processa a carga de sinalização do simulador $OAISIM_3$.

6.4 Avaliação de Desempenho

Nesta seção, os modelos analíticos propostos nas Seções 6.2.1 e 6.2.2 são alimentados com os resultados numéricos da Seção 6.3. O objetivo é extrapolar os valores obtidos no experimento para reproduzir sinteticamente o cenário de exaustão no plano de controle em decorrência do ataque e demonstrar o funcionamento do escalonamento de recursos para manter a disponibilidade dos serviços de sinalização. Além disto, será apresentada uma análise das tendências comportamentais do atacante e do defensor, baseada nos estados de equilíbrio de Nash do jogo de mitigação, onde teoricamente os jogadores se encontram conformados com as suas respectivas

estratégias e não tem motivos para modificá-las.

6.4.1 Escalonamento de Recursos

O modelo analítico apresentado na Seção 6.2.1 leva em consideração os rigorosos requerimentos de latência do tráfego de sinalização, particularmente a transação de solicitação de registro (*attach request*) para propor o sistema de filas ilustrado na Figura 6.3, como uma aproximação assintótica do cenário de exaustão no plano de controle em decorrência de um suposto ataque de DDoS.

No cenário de ataque proposto em 6.1, os EUs comprometidos (*bots*) são remotamente controlados por um agressor para atentar contra a disponibilidade do vEPC através de um ataque de DDoS de sinalização. O atacante tira vantagem do grande número de mensagens desencadeadas pela transação de solicitação de registro na rede para inundar o vEPC e exaurir os recursos do plano de controle. Durante o ataque, cada EU que perde uma transação de registro gera imediatamente após uma nova solicitação ao vEPC, de acordo com o roteiro do ataque. Este processo se repete indefinidamente, enquanto o ataque está ativo. Por outro lado, ao detectar uma anormalidade no tráfego de sinalização, o defensor adota as medidas de proteção necessárias para assimilar o tráfego (legítimo e anômalo), antessipando o procedimento de balanceamento de carga.

Baseado na Figura 6.7, é possível estimar a taxa média de chegada de usuários (transações) no sistema $\lambda_a \approx 6/120 = 0{,}05$ transações por segundo. Considerando que todos os 6 EUs configurados nos simuladores OAISIM fazem parte do mesmo grupo de *bots* (*botnet*), o gráfico mostra que o vMME_1 recebeu 6 solicitações de registro (usuários) em aproximadamente 120 segundos. O valor absoluto da ocupação da memória do vMME_1 para processar as 6 transações de solicitação de registro (*attach request*) chegou a 0,03. Isto significa que a taxa de serviço do servidor ou a capacidade do vMME_1 (servidor) para processar estas 6 transações (usuários) pode ser calculada como $\mu = 0{,}05/0{,}03 \approx 2$ transações por segundo (ou usuários por segundo, considerando o modelo proposto). Note que a taxa de serviço μ é um parâmetro fixo, que depende apenas do tamanho médio das mensagens de sinalização que serão processadas.

O processo de sinalização da comunicação móvel é muito dinâmico e rigorosamente dependente do tempo. Assim, para cada transação de sinalização existe uma infinidade de contadores que limitam quanto tempo uma determinada transação pode ficar aguardando por uma resposta, antes de ser abandonada por retardo excessivo (*timeout*). No caso da solicitação de registro, considerada neste experimento, o tempo máximo que o EU aguarda a resposta do vMME antes de descartar a seção é de 10 segundos (contador T3410) [174]. Assim, considerando o cenário de

ataque endereçado, mesmo se uma transação solicitação de registro for processada pelo vMME dentro deste intervalo de tempo T=10 segundos, o roteiro do ataque se repete indefinida e deliberadamente, depois de um certo tempo. Portanto, voltando ao modelo proposto, se um usuário não for servido em até 10 segundos, ele simplesmente deixa o sistema e retorna ao estado inicial para novo disparo, como se tivesse sido efetivamente servido pelo sistema. Dentro do exposto, é possível assumir então um sistema sem perdas para lançar mão da Lei de Little no cálculo do tamanho máximo da fila de cada servidor q=$T\lambda_a$=10×0,05≈1, como o menor inteiro maior que 0,5. O tamanho total da fila Q e o espaço total de armazenamento do sistema K, considerando então os m servidores operando no sistema, podem ser obtidos a partir das igualdades Q=m e K=2m, respectivamente.

Utilizando os parâmetros obtidos nos parágrafos anteriores na Equação 6.1, é possível calcular a probabilidade de bloqueio p_B=Pr(I=K), sendo a probabilidade de um usuário chegar ao sistema e ser descartado pelo mesmo, ao encontrá-lo cheio. Voltando ao cenário modelado, o estado de bloqueio indica que o vEPC se encontra perdendo transações de sinalização dos EUs. Esta perda tende a aumentar à medida que aumenta o número de *bots* atacando o plano de controle. A Figura 6.9 mostra o comportamento da probabilidade de bloqueio p_B em função do número de *bots* atuando no ataque, considerando diferentes configurações do número de vMMEs em operação dentro do vEPC.

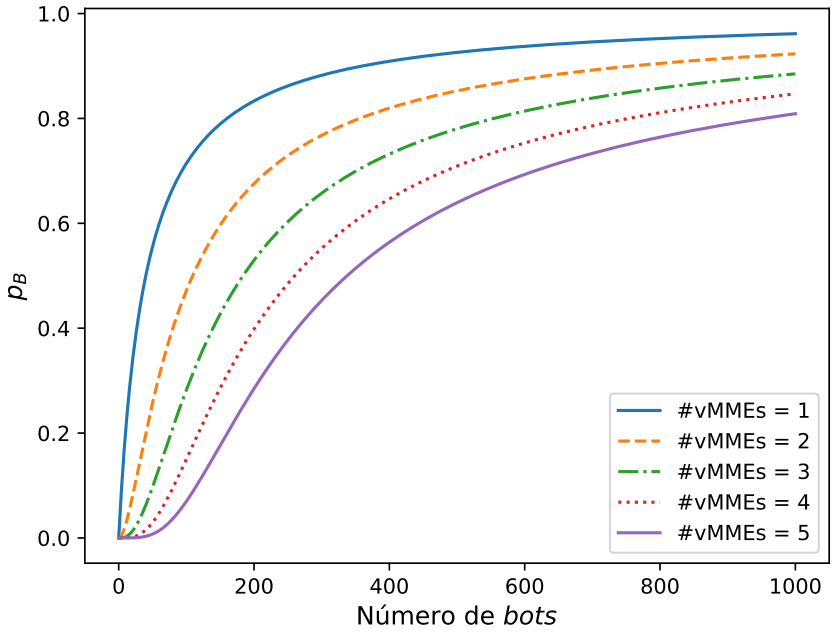


Figura 6.9: Probabilidade de bloqueio p_B da Equação 6.1 para k=K. Isto é, quando o número de usuários dentro do sistema k (sendo servidos ou aguardando em fila) preenche todo o espaço de armazenamento do mesmo K.

Como pode ser visto na Figura 6.9, a probabilidade de bloqueio p_B aumenta muito rapidamente até aproximadamente 200 *bots* envolvidos no ataque. Contando apenas com um único vMME em operação no vEPC, é possível perceber que o plano de controle perde cerca de 80% das transações de sinalização com 100 *bots* atacando o sistema de forma coordenada. Um outro extremo pode ser observado quando o sistema opera com 5 vMMEs. Neste caso, os mesmos 100 *bots* atacando o sistema, causam uma perda inferior a 1%, o que é um valor bastante razoável, em se tratando do sistema de comunicação celular 5G [47].

De acordo com a abordagem proposta, considerando o mesmo cenário de ataque composto por 100 *bots*, a adição de um segundo vMME em operação no vEPV é suficiente para reduzir a probabilidade de perdas em 20% (80% - 60%). Além de manter a disponibilidade do plano de controle, esta redução oferece ao defensor um tempo crucial para caracterizar o ataque e aliar contra-medidas mais efetivas e duradouras ao sistema de defesa, uma vez que não é viável para o defensor continuar escalando o número de vMMEs por muito tempo.

Como será mostrado na próxima seção, o escalonamento inteligente dos recursos do plano de controle pode elevar a relação custo benefício do ataque, frustrando o agressor e fazendo com que o mesmo perca a motivação e o *momentum* do ataque, a ponto de desistir do mesmo eventualmente.

6.4.2 Tendências de Comportamento

A abordagem de mitigação proposta nesta Capítulo 6 se baseia na tendência de comportamento dos jogadores (atacante e defensor) num jogo não-cooperativo para maximizarem suas respectivas recompensas. Durante o jogo, várias configurações diferentes da dupla número de *bots* x número de vMMEs ($M \times m$) serão testadas, capturando as probabilidades de alteração de estratégias de jogo, em relação aos pontos de equilíbrio de Nash.

A análise de comportamento dos jogadores se inicia com o levantamento dos pontos de equilíbrio de Nash para cada combinação $M \times m$. Isto é, enquanto o número de *bots* varia de 0 a 100, o número de vMMEs supostamente em operação dentro do vEPC varia de 1 a 5. Os intervalos usados para simular as recompensas dos jogadores em cada cenário $M \times m$ provém do modelo proposto em [175], considerando o tempo de inatividade dos EUs igual a 10 segundos.

A Figura 6.10 mostra o comportamento da recompensa final do atacante (U_a) em função do número de *bots*, considerando diferentes números de vMMEs em operação no plano de controle. O ponto A na Figura 6.10 marca o valor máximo da recompensa final do atacante, quando o mesmo lança o ataque contando com 24 *bots* contra o vEPC com apenas 1 vMME em operação. Sabendo que a recompensa

do atacante é função da taxa de sinalização anômala λ_a=10×λ_l e do número de *bots* envolvidos no ataque, o valor U_a=60,9 significa que este cenário de ataque causa perdas de sinalização em aproximadamente 60 EUs por segundo, valor compatível com os resultados apresentados em [176, 177].

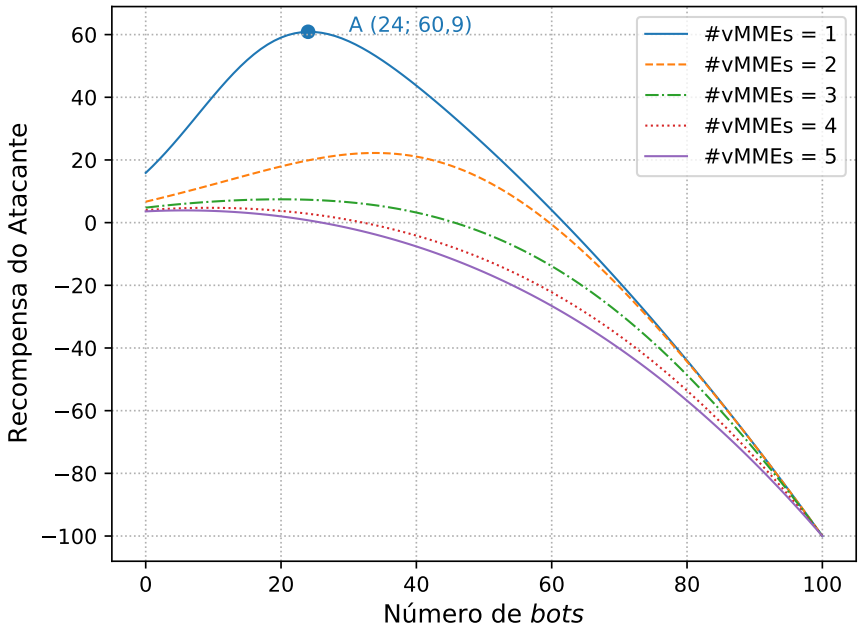


Figura 6.10: Perfil da recompensa do atacante U_a em função do número de *bots* M, considerando diferentes composições de vMMEs em operação no vEPC.

A Figura 6.11, por outro lado, mostra o perfil de recompensa final do defensor (U_d) em função do número de *bots* acionados pelo atacante, considerando diferentes configurações de vMMEs no plano de controle. O ponto A da Figura 6.11 marca o valor máximo da recompensa final do defensor, que ocorre quando apenas um vMME está em operação no vEPC e não existe nenhum ataque em andamento(M=0). Na verdade, trata-se de um comportamento esperado, uma vez que o defensor não tem qualquer perda no ataque, nem portanto tem que gastar recursos para se defender. Observando particularmente a curva de 1 vMME da Figura 6.11, nota-se que a recompensa (prejuízo) do defensor permanece constante a partir de 40 *bots* atacando. Este comportamento decorre do completo esgotamento do único vMME em operação a partir deste número de *bots*.

A Figura 6.12 compara as recompensas finais do atacante e do defensor (U_a e U_d) em função do número de *bots*, considerando apenas um vMME em operação no vEPC. O ponto A marca a primeira alteração de comportamento do atacante, quando ele tem recrutado 12 *bots* e tende a iniciar um ataque contra o plano de controle. O ponto B marca a segunda alteração de comportamento, quando ele atinge um total de 50 *bots* envolvidos no ataque e tende a interromper um eventual

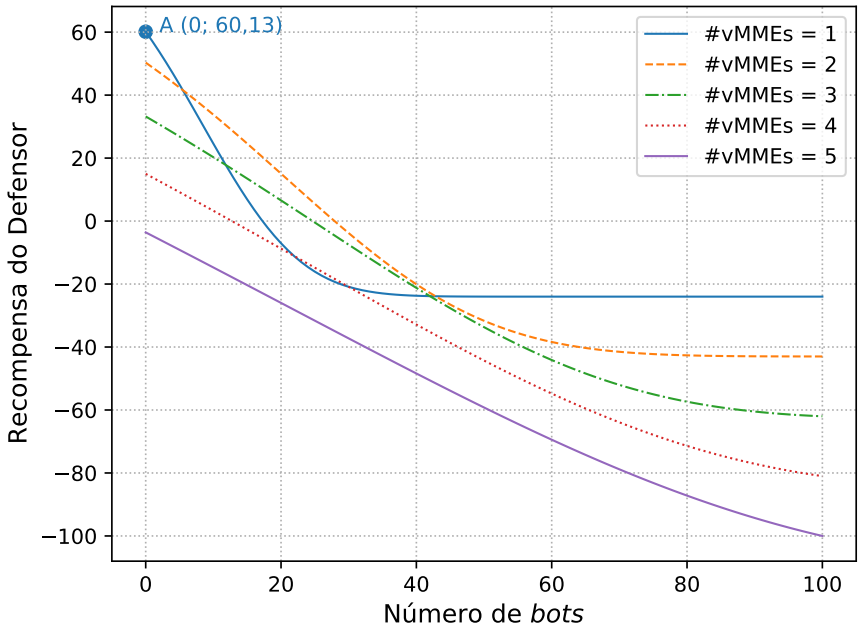


Figura 6.11: Perfil de recompensa do defensor U_a em função do número de *bots* M, considerando diferentes composições de vMMEs em operação no vEPC.

ataque. Do lado do defensor, o ponto C marca uma alteração no comportamento do defensor, quando ele tende a parar de defender o plano de controle.

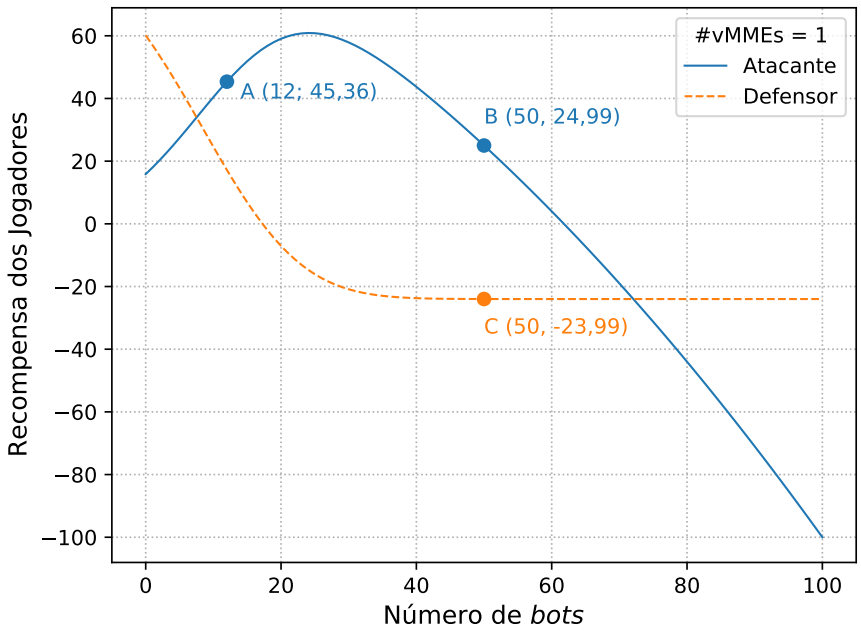


Figura 6.12: Perfis das recompensas do atacante e defensor (U_a e U_d) em função do número de *bots*, considerando somente 1 vMME em operação no vEPC.

A Figura 6.13 ainda compara os perfis de recompensa final do atacante e do defensor, considerando agora 2 vMMEs em operação no plano de controle. Neste cenário o atacante tende a não mais iniciar um ataque contra o plano de controle. O ponto A, contando com 39 *bots*, é o ponto onde há uma maior probabilidade do atacante iniciar o ataque com 35% de chances. O ponto B marca o momento no qual o comportamento do defensor tende a se alterar, passando de "defender" para "parar de defender".

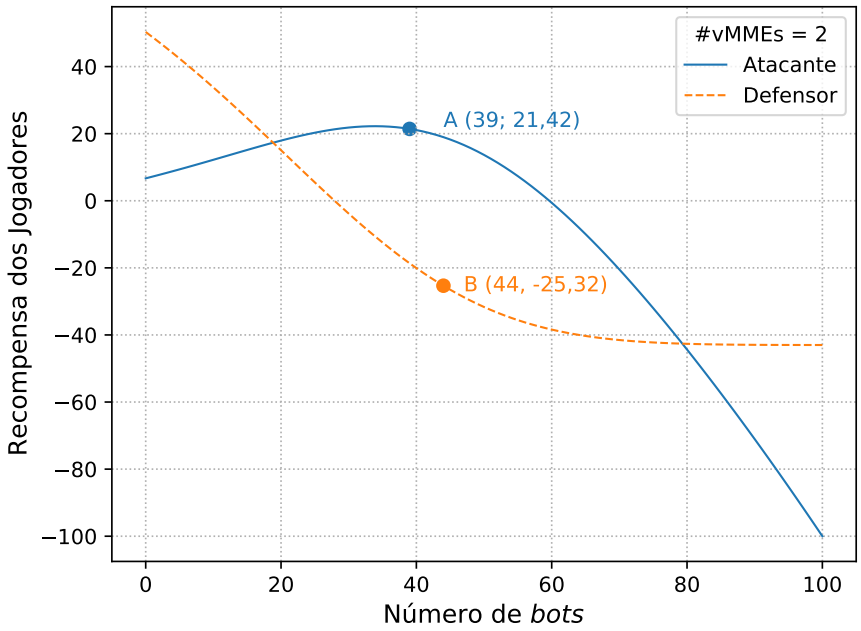


Figura 6.13: Perfis de recompensa do atacante e defensor (U_a e U_d) em função do número de *bots*, considerando 2 vMMEs em operação no vEPC.

Comparando as Figuras 6.12 e 6.13, percebe-se uma significativa diferença no comportamento do atacante, quando o plano de controle opera com 2 vMMEs. Enquanto na Figura 6.12 o intervalo onde o atacante tende a iniciar um ataque ocorre em $12 \leq M \leq 50$, na Figura 6.13 este intervalo não mais existe. Embora o defensor dificilmente saberia antecipadamente o número de *bots* inicial de um ataque, esta comparação mostra a efetividade do escalonamento de vMMEs no plano de controle no sentido de reduzir as chances de um eventual ataque de DDoS de sinalização, imediatamente duplicando o número de vMMEs em operação.

Capítulo 7

Conclusão

Este trabalho de Doutorado endereça problemas típicos, mas que apesar das inúmeras abordagens propostas na academia, permanecem desafiadores em vários aspectos. Sem exaustão, pode-se citar, por exemplo, (*i*) a definição de uma plataforma de comunicação segura e de fácil implementação para possibilitar a colaboração entre os agentes federados, no caso do sistema distribuído de detecção de intrusões proposto no Capítulo 4.

Ainda no contexto da federação de agentes, (*ii*) a suposição de que todos os agentes distribuídos na Internet possuem uma mesma reputação, independentemente do sistema autônomo ao qual pertence, pode ser entendida como frágil, considerando a essência autônoma e diversa da Internet. Esta foi, portanto, a principal motivação para a proposta apresentada no Capítulo 5, que apresenta um modelo de aprendizado de máquina capaz de inferir sobre o nível de crença dos anúncios BGP, a partir de atributos extraídos direta ou indiretamente do cabeçalho de cada mensagem, individualmente.

Um outro aspecto de segurança bastante conhecido se refere (*iii*) à relação de compromisso (*tradeoff*) entre "segurança e funcionamento". Neste aspecto, considera-se que qualquer sistema de mitigação que cause prejuízo aos serviços legítimos, eventualmente acaba por colaborar para o objetivo do ataque. Este *tradeoff*, foi a principal motivação para o sistema de mitigação proposto no Capítulo 6, onde recursos normalmente ociosos, são escalonados inteligentemente para absorver o tráfego excedente, garantindo um tempo precioso para aprimorar as defesas contra os ataques de negação de serviços.

Por fim, é importante enfatizar que a mesma evolução tecnológica que oferece novas oportunidades aos atacantes cibernéticos, também joga luz sobre novas estratégias de defesa. Novas abordagens segundo a estratégia de defesa em profundidade, como a ampliação da superfície de detecção e técnicas de resiliência baseadas em absorção de ataques com imposição de frustração, abrem caminhos para soluções inovadoras de segurança.

7.1 Sistema Distribuído de Detecção de Intrusões

O problema enfrentado pelos provedores de serviços, que além de terem que se protegerem a si próprios contra os ataques cibernéticos vindos da Internet, ainda podem ser responsabilizados por transportarem fluxos maliciosos até os seus clientes, é endereçado neste trabalho de uma forma holística e prática, dando origem à proposta de um sistema distribuído de detecção de intrusões. O sistema funciona de forma cooperativa, resgatando com responsabilidade a essência colaborativa da Internet, aproveitando-se de recursos já existentes e dispensando grandes alterações estruturais. Recursos como os já existentes IDSs típicos, que já fazem parte da grande maioria das topologias dos sistemas autônomos, do protocolo BGP, que cumpre muito bem seu papel crítico de interconectar os ASs considerando aspectos técnicos e de negócio e o FlowSpec, que apesar de não ser largamente utilizado no domínio externo, faz parte de muitos sistemas internos de mitigação de ataques de DDoS.

Considerando o quesito simplicidade de implementação, o sistema proposto se diferencia das demais propostas de sistemas distribuídos de detecção. Além de utilizar recursos de segurança já existentes na maioria dos sistemas autônomos, o sistema distribuído proposto neste trabalho ainda tira proveito da própria essência de conectividade da Internet. No que tange ao desempenho, os resultados dos modelos analíticos e de emulação propostos para avaliar o sistema, mostram que, dependendo da qualidade dos agentes federados distribuídos e do tamanho da federação, é possível chegar a valores de TPR e FPR bastante significativos, comparáveis às melhores tecnologias atuais. O gráfico que relaciona o TPR com o FPR mostra também que o processo de combinação dos anúncios correlacionados mantém uma diferença estável, mesmo com uma taxa de alarmes positivos elevada.

As métricas de desempenho, calculadas a partir do modelo descrito na Seção 4.2.2, mostram uma melhoria considerável no desempenho de detecção da plataforma, aumentando as taxas de verdadeiro-positivos e verdadeiro-negativos (TPR e TNR) e reduzindo das taxas de falso-positivos e falso-negativos (FPR e FNR). Este comportamento foi mostrado em gráficos tridimensionais e bidimensionais, variando-se simultaneamente o número de IDSs membros que são atravessados (N_I) e que detectam a intrusões (N_D); e o valor de predição positivo médio (PPV_{av}) dos IDSs federados.

Também foi possível observar que o desempenho da plataforma DIDS nem sempre é melhor que o desempenho de um único IDS isolado. Por exemplo: se o valor de predição positivo médio for um valor muito baixo (PPV_{av}<0,3) ou o número de IDSs que detectam a intrusão é bem menor que os IDSs atravessados pela mesma (N_D=1 e N_I=6), o desempenho de detecção da plataforma é inferior ao de um IDS típico monolítico. Comparando os números de desempenho obtidos pela plata-

forma DIDS proposta nesta tese com outros trabalhos relacionados [124, 178, 179], é possível perceber a coerência do modelo analítico proposto, que pode inclusive ser utilizado para avaliar outros sistemas distribuídos de detecção

Neste modelo, o conhecimento prévio traduzido pela probabilidade à priori é condicionado aos dados positivos e negativos das hipóteses de detecção. Esta condição tem como principal objetivo reduzir a dependência do modelo aos resultados das métricas, obtidas através da frequência relativa das matrizes de confusão dos *datasets* utilizados. No caso de ataques cibernéticos, a utilização da frequência relativa pode não aderir completamente à realidade, principalmente por não ser exaustiva para todos os tipos de ataques. Os resultados realmente mostram que o desempenho de detecção da plataforma DIDS aumenta de forma geral à medida que $N_D \rightarrow N_I$. O método de fusão de dados de Dempster-Shafer, utilizado no modelo para medir o nível de confiança da mensagem fundida, se enquadra no contexto proposto pelas características de incerteza e diversidade das fontes e pelo caráter concordante das evidências. Outros métodos de fusão podem ser testados e comparados entre si e em relação a resultados práticos. Entretanto, como mencionado na Seção 4.3, a modelagem não funciona nos cenários onde não existem evidências de detecção para serem combinadas, deixando assim de algumas métricas importantes como o NPV de fora.

O modelo experimental proposto na Seção 4.5 mostra o funcionamento do protocolo FlowSpec BGP num cenário de composto por 5 roteadores distribuídos em 5 diferentes ASs. Os registros do Wireshark coletados na interface WAN do roteador $R5$ mostram a possibilidade de combinar as mensagens, considerando o endereço IP de destino. A normalização das evidências, através das mensagens de atualização BGP disseminadas pelos IDSs federados, facilita o processo de fusão, homogeneizando a combinação, sem perder o aspecto principal como uma primeira linha de defesa avançada contra as intrusões. A autonomia dos membros federados cria uma base heterogênea nos métodos de detecção e aumenta as chances de detecção de um novo ataque (*zero-day attack*). Um novo ataque no Brasil, pode não ser novo na China.

7.2 Modelo de Aprendizado de Máquina para Inferir sobre a Reputação dos Anúncios BGP

Apesar dos importantes benefícios trazidos pela sua arquitetura distribuída e heterogeneidade dos membros, o sistema distribuído de detecção de intrusões também está sujeito à riscos de segurança inerentes a este próprio ambiente. A mesma rede BGP que é utilizada para constituir sua rede sobreposta de comunicação, também

tem sido utilizada em muitos ataques cibernéticos como alvo e como transporte de códigos maliciosos. Na proposta descrita no Capítulo 5, os anúncios BGP-FlowSpec que chegam para serem combinados no AS de destino têm sua massa de crença avaliada, de acordo com informações (atributos) extraídas individualmente do cabeçalho das mensagens. Assim como um anúncio único com uma massa de crença alta pode ser suficiente para suportar uma contra-medida dura de proteção, vários anúncios com massa de crença baixa podem ser desprezados, considerando que são falso-positivos. Os resultados obtidos do modelo de aprendizado de máquina proposto no Capítulo 5 mostra que (*i*) o cabeçalho das mensagens BGP podem fornecer atributos importantes, que podem ser utilizadas para inferir sobre a reputação do seu sistema autônomo de origem e (*ii*) que é possível treinar um algoritmo de aprendizado de máquina com estes atributos, capaz de generalizar para predizer sobre a massa de crença de novos anúncios.

O *dataset* utilizado para testar as hipóteses levantadas no parágrafo anterior foi construído com 15 atributos, extraídos individualmente de cada anúncio BGP coletado durante o ataque Code Red II, que aconteceu em julho de 2001. Este conjunto de dados, ainda sem o rótulo de ataque, foi submetido a dois algoritmos não-supervisionados de aglomeração, cujos resultados mostraram níveis bastante claros de agrupamento e separação entre os grupos.

O mesmo *dataset* utilizado nos testes não-supervisionados foi posteriormente rotulado, combinando-se os seus registros com o *dataset* proposto em [138]. O conjunto de dados rotulado foi utilizado então para treinar, validar e testar um modelo de aprendizado baseado em rede neural. Os resultados obtidos deste modelo mostram um comportamento convergente entre a base de treinamento e de testes, assegurando que o modelo foi capaz de aprender o suficiente para generalizar as predições de regressão para dados novos.

Considerando a parte de classificação, o teste de dados novos sobre o modelo treinado gerou uma matriz de confusão, donde foram extraídas métricas de desempenho específicas, como: taxa de verdadeiro-positivos, precisão, taxa de falso-positivos, acurácia, etc. Analisando estas métricas, é possível concluir que além de possível, o aprendizado a partir dos 15 atributos tem uma alta sensibilidade (TPR) e é suficiente para inferir acuradamente sobre as massas de crenças dos anúncios novos. Já quanto à precisão das predições (PPV), o modelo deixa um pouco a desejar, indicando que ainda há espaços para melhorias e desenvolvimentos adicionais.

7.3 Sistema de Mitigação de Ataques de DDoS de Sinalização

O enorme problema trazido pelos ataques de DDoS não é novo. Pelo contrário, o ataque de negação de serviços já representa atualmente um dos maiores riscos para a evolução da Internet e das redes de próxima geração, como o 5G e o 6G. Pela sua simplicidade e até mesmo pela sua robustez, este tipo de ataque consegue atingir seus objetivos muito facilmente e são frequentemente devastadores, principalmente em se tratando de sistemas de sinalização. A maioria das abordagens de defesa contra os ataques de DDoS passa por uma fase de bloqueio, que muitas vezes acaba comprometendo o tráfego legítimo e colaborando para a efetividade do ataque.

Entretanto, a mesma evolução tecnológica que abre novas oportunidades de ataques ainda mais complexos, também abre novas abordagens de defesa, até então desconhecidas ou desconsideradas. O novo ambiente virtualizado do plano de controle 5G facilita muito o escalonamento de recursos. O que antes significava a compra e a instalação física, atualmente se resume num comando que pode inclusive ser disparado automaticamente sob demanda. Esta enorme facilidade é, na verdade, a principal viabilizadora da segunda proposta para assimilar o tráfego atacante, utilizando recursos que ficariam ociosos em regime normal. Apesar de parecer ingênuo à primeira vista, a ideia de escalar recursos para se proteger de um ataque cibernético é muito similar à contratação de uma apólice de seguros. Ou seja, ninguém que contrata uma apólice de seguros espera utilizar este recurso em algum momento, mas numa situação crítica, na qual a perda é catastrófica em muitos sentidos, ter a segurança de que poderá contar com este recurso vale muito o custo da ociosidade.

A imediata ação para assimilar o tráfego excedente ao invés de simplesmente bloqueá-lo garante dois benefícios importantes:

1. Um intervalo de tempo sem perda de disponibilidade, que poder ser utilizado para preparar contra-medidas de proteção mais efetivas e abrangentes, sem comprometer o tráfego legítimo.

2. A possibilidade de interromper o ataque, reduzindo a relação custo/benefício para o atacante e impondo frustração ao atacante.

O Capítulo 6 endereça os efeitos dos ataques de negação de serviços contra o plano de controle das redes 5G, considerando a garantia da disponibilidade dos serviços como o principal objetivo. Baseado no núcleo virtualizado do 5G, propõe-se uma estratégia de segurança para evitar interrupções no plano de controle através do escalonamento inteligente no número de vMMEs. O tráfego de sinalização, inclusive o tráfego malicioso, é balanceado entre os vMMEs para assimilar os efeitos do ataque,

sem comprometer o tráfego legítimo. Os resultados do modelo de fila asseguram que os mecanismos de balanceamento de carga do próprio 5G, combinados com o escalonamento de recursos, são eficientes no sentido de evitar interrupções imediatas por exaustão.

Além de garantir um tempo adicional importante para se melhorar os sistemas de defesa ao longo do ataque, há que se considerar ainda a possibilidade de frustrar o atacante, induzindo-o a desistir do ataque. De fato, os resultados do modelo analítico baseado em Teoria de Filas sugerem uma tendência comportamental do atacante de desistir do ataque em função da relação custo/benefício.

Os novos vMMEs adicionados podem ser equipados com mecanismos de segurança mais evoluídos, desenvolvidos com base nas informações do ataque levantadas anteriormente. Esta possibilidade sugere novos trabalhos futuros na direção de se criar algo similar a uma vacina para os novos elementos, protegendo-o do ataque e aumentando a imunidade do sistema para outros ataques similares.

7.4 Lista de Contribuições

Nesta seção, apresenta-se de forma compacta as principais contribuições desta tese de doutorado, após a compilação das seções de análise dos Capítulos 4, 5 e 6.

- O sistema distribuído de detecção de intrusões proposto no Capítulo 4 se apresenta como uma plataforma viável, com alto desempenho funcional e capacidade para escalar, considerando que utiliza recursos já existentes, como a própria rede BGP mundial e os IDSs supostamente já existentes nos sistemas autônomos.

- A estrutura de modelagem proposta para combinar as massas de crença das evidências de intrusão e também para avaliar o desempenho funcional do sistema de detecção podem ser utilizado em outros sistemas cooperativos similares.

- A plataforma de detecção pode receber dados de entrada provenientes de IDSs de tecnologias diferentes, consolidando a hipótese de intrusão e suportando decisões de segurança.

- Por lidar apenas com evidências positivas de intrusão, o cálculo de algumas métricas baseadas em evidências negativas ficam comprometidas, como no cálculo do NPV - *Negative Prediction Value*.

- Esta primeira proposta contempla apenas a parte de detecção, deixando de fora a parte de classificação dos ataques. Entretanto, a estrutura de fusão de dados

baseada na Teoria da Evidência dá suporte a este tipo de complementação em trabalhos futuros.

- O modelo de aprendizado de máquina proposto no Capítulo 5 para inferir sobre a reputação dos sistemas autônomos e consolidar o nível de crença dos anúncios BGP individualmente, apesar de não atingir uma precisão muito alta (68%), atinge uma acurácia superior a 90%.

- Apesar de ser suficiente para treinar o modelo de aprendizado e generalizar para dados novos, o *dataset* rotulado com 15 atributos diretos e indiretos, extraídos individualmente do cabeçalho dos anúncios, pode ser melhorado, com a inclusão novos atributos.

- O *dataset* disponibilizado no trabalho pode ser utilizado para treinar outros modelos de aprendizado com objetivos similares.

- O sistema de mitigação de ataques de DDoS de sinalização proposto no Capítulo 6 é viável como uma primeira linha de defesa, reduzindo a carga dos servidores em 20% a cada rodada e atrasando os efeitos exaustivos do ataque no sentido de garantir o tempo necessário para a adoção de medidas mais efetivas de defesa, sem prejudicar o tráfego legítimo.

- O modelo de escalonamento de recursos baseado em Teoria de Jogos aumenta inteligentemente a relação custo/benefício do ataque a ponto de provocar sua interrupção por dissuasão e pode ser utilizado em outros trabalhos similares, considerando ataques racionais.

- O modelo de filas para avaliar o desempenho do processo de balanceamento de carga para assimilar o tráfego excedente e evitar a imediata exaustão do plano de controle pode ser utilizado na análise de sistemas similares.

7.5 Aplicações Práticas

Os sistemas propostos nos Capítulos 4, 5 e 6 são avaliados nesta seção em relação às perspectivas de utilização prática para melhorar a eficiência dos sistemas de segurança já existentes em operação.

No caso do sistema distribuído de detecção de intrusões proposto no Capítulo 4, acredita-se na possibilidade da formação de uma federação cooperativa composta principalmente por provedores de serviços de Internet. Importante lembrar que mesmo sendo um sub-conjunto de sistemas autônomos, quanto maior for o número de membros e quanto mais espalhados estiverem na federação, maior a eficiência do modelo. Os maiores desafios desta suposta federação de provedores de serviços

ainda estaria na padronização de configurações BGP para interconectar os sistemas autônomos usando o BGP-FlowSpec e a definição de parâmetros mínimos para os IDSs agentes em cada AS.

O sistema de aprendizado de máquina proposto no Capítulo 5, para inferir sobre a reputação dos sistemas autônomos a partir dos anúncios BGP originados por estes, pode ser melhorado para aumentar a precisão das inferências. Uma forma para se conseguir esta melhoria seria no aumento do número de atributos do *dataset*, considerando outros aspectos ou outras fontes de informação, que possam ser obtidas individualmente de cada anúncio. Uma outra aplicação prática viabilizada por este modelo seria num possível sistema de pré-avaliação do anúncio, antes do do processamento da rota no AS de destino. Apesar de não fazer nenhuma relação direta com o sistema de detecção proposto, esta aplicação teria como principal objetivo proteger os usuários de um determinado sistema autônomo, isolando conectividades suspeitas na INternet.

O sistema de mitigação de ataques de DDoS proposto no Capítulo 6 depende de uma arquitetura de orquestração de recursos de nuvem que pode ser construída com sistemas de código aberto como o o OSM [180]. O OSM é um orquestrador de código aberto bastante conhecido e estável que inclusive pode ser utilizado para integrar diferentes infraestruturas de nuvem, não somente o Openstack.

7.6 Trabalhos Futuros

Ao longo do desenvolvimento deste trabalho, que compreende basicamente os sistemas propostos nos Capítulos 4 e 6, muitas ideias promissoras de continuidade foram surgindo, mas também foram desaparecendo com a evolução no conhecimento. Ao chegar finalmente na conclusão deste trabalho de doutorado, poucas, mas sólidas ideias de trabalhos futuros permaneceram e são relacionadas logo abaixo:

- Levando-se em conta a estrutura que já existe em operação no Laboratório Ravel, composta basicamente pelo sistema de coleta de registros de fluxos IPTraf [181] e dos sistemas de detecção de anomalias que o utilizam como fonte de dados, é possível agregar no IPTraf a coleta de mensagens BGP. Esta nova fonte de dados, dando uma visão de alteração de topologia, analisada com relação à detecção de eventuais anomalias e combinada com os outros sistema, consolidando um sistema de detecção mais amplo e com mais acurácia.

- O modelo de aprendizado de máquina proposto no Capítulo 5 para inferir sobre a massa de crença dos anúncios BGP a partir de atributos individuais pode ser bastante útil para se tomar decisões de roteamento mais seguras, considerando os perigos e os riscos inerentes à rede BGP [78].

- O valor da massa de crença dos anúncios BGP que chegam para serem fundidos podem ser incorporados à equação de fusão de dados, melhorando a confiabilidade e a precisão da informação resultante.
- O desenvolvimento de experimentos práticos para testar e medir os limites da estratégia de escalonamento de recursos como fator indutivo na interrupção do ataque.
- O mesmo experimento prático proposto no item anterior também poderá ser utilizado para testar o sistema de escalonamento de recursos imunizados, a partir das informações coletadas dos fluxos maliciosos assimilados pela plataforma.

Referências Bibliográficas

[1] VALERO LEÓN, A. *INsIDES: A New Machine Learning-based Intrusion Detection System*. Ph.D. thesys, Universitat Pompeo Fabra, Barcelona, Spain, June 2017.

[2] NAJAFIAN, Z., AGHAZARIAN, V., HEDAYATI, A. "Signature-Based Method and Stream Data Mining Technique Performance Evaluation for Security and Intrusion Detection in Advanced Metering Infrastructures (AMI)". v. 7, pp. 128–139, 2015.

[3] MURPHEY, D. "A history of information security". Jun 2019. Disponível em: <`https://www.ifsecglobal.com/cyber-security/a-history-of-information-security/`>. (Acessado em: 10 nov. 2020, 11:35:10.).

[4] RAO, U. H., NAYAK, U. "History of Computer Security". In: *The InfoSec Handbook: An Introduction to Information Security*, pp. 13–25, Berkeley, CA, Apress, 2014.

[5] STIAWAN, D., YASEEN, A., IDRIS, Y., et al. "Intrusion prevention system: A survey", *Journal of Theoretical and Applied Information Technology*, v. 7, pp. 44–54, 06 2012.

[6] DAYA, B. "Network security: History, importance, and future", *University of Florida Department of Electrical and Computer Engineering*, v. 4, 2013.

[7] CHIERICI, L., FIORINI, G. L., LA ROVERE, S., et al. "The evolution of defense in depth approach: A cross sectorial analysis", *Open Journal of Safety Science and Technology*, v. 6, n. 2, pp. 35–54, 2016.

[8] (IAAG), I. A. A. G. *Defence in depth*. In: Report, ©TISN, jun. 2008.

[9] VACCA, J. R. *Network and System Security, Second Edition*. Syngress Publishing, 2013.

[10] OF HOMELAND SECURITY, U. D. *Recommended Practice: Improving Industrial Control Systems Cybersecurity with Defense-In-Depth Strategies*. North Charleston, SC, USA, CreateSpace Independent Publishing Platform, 2014.

[11] LOUKAS, G., DIANE GAN, TUAN VUONG. "A taxonomy of cyber attack and defence mechanisms for emergency management networks". In: *2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, pp. 534–539, 2013.

[12] ODOGWU, C. "Segurança reativa vs. proativa: o que é mais eficaz?" Jul 2021. Disponível em: <`https://vixblog.com/seguranca-reativa-vs-proativa-o-que-e-mais-eficaz/`>. (Acessado em: 17 set. 2021, 08:37:15.).

[13] PANDEY, K., KUMAR, V., PUNIA, D. "Facing the Reality Of Cyber Threats In The Power Sector", *Wipro Ltd.*, 11 2013.

[14] CORONA, I., GIACINTO, G., ROLI, F. "Adversarial Attacks against Intrusion Detection Systems: Taxonomy, Solutions and Open Issues", v. 239, pp. 201–225, 2013.

[15] BROWN, D. J., SUCKOW, B., WANG, T. "A Survey of Intrusion Detection Systems", *Department of Computer Science, University of California, San Diego*, 2002.

[16] AXELSSON, S. *Intrusion Detection Systems: A Survey and Taxonomy*. Relatório técnico, 2000.

[17] SABAHI, F., MOVAGHAR, A. "Intrusion Detection: A Survey". In: *2008 Third International Conference on Systems and Networks Communications*, pp. 23–26, out. 2008.

[18] DEBAR, H., DACIER, M., WESPI, A. "Towards a Taxonomy of Intrusion-detection Systems", *Comput. Netw.*, v. 31, n. 9, pp. 805–822, abr. 1999.

[19] LAZAREVIC, A., KUMAR, V., SRIVASTAVA, J. "Intrusion Detection: A Survey". In: Kumar, V., Srivastava, J., Lazarevic, A. (Eds.), *Managing Cyber Threats: Issues, Approaches, and Challenges*, pp. 19–78, Boston, MA, Springer US, 2005.

[20] LAZAREVIC, A., KUMAR, V., SRIVASTAVA, J. "Intrusion Detection: A Survey". In: *Managing Cyber Threats: Issues, Approaches, and Challenges*, pp. 19–78, Boston, MA, Springer US, 2005.

[21] MUKKAMALA, S., JANOSKI, G., SUNG, A. "Intrusion detection using neural networks and support vector machines". In: *Neural Networks, 2002. IJCNN '02. Proceedings of the 2002 International Joint Conference on*, v. 2, pp. 1702–1707, 2002.

[22] KIM, J., BENTLEY, P. "The Human Immune System and Network Intrusion Detection". pp. 1244–1252, 2001.

[23] MOYSEN, J., GIUPPONI, L. "From 4G to 5G: Self-organized network management meets machine learning", *Computer Communications*, v. 129, pp. 248 – 268, 2018.

[24] ©SNS TELECOM & IT. "SON (Self-Organizing Networks) in the 5G Era: 2019 – 2030 – Opportunities, Challenges, Strategies & Forecasts". Sep 2018. Disponível em: <`https://www.snstelecom.com/son`>. (Acessado em: 03 set. 2020, 15:18:10.).

[25] LIAO, Q., STANCZAK, S. "Network State Awareness and Proactive Anomaly Detection in Self-Organizing Networks". In: *2015 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6, Dec 2015.

[26] MARQUES, P. R., MAUCH, J., SHETH, N., et al. "Dissemination of Flow Specification Rules". Aug 2009.

[27] BATES, T., CHANDRA, R., KATZ, D., et al. "Multiprotocol Extensions for BGP-4". January 2007.

[28] GELMAN, A. *Bayesian Data Analysis*. Chapman and Hall/CRC.

[29] MA, W., BECK, J., LATHAM, P., et al. "Bayesian inference with probabilistic population codes", *Nature Neuroscience*, v. 9, n. 11, pp. 1432–1438, 11 2006.

[30] GUOQUAN LI, ZHENG YAN, Y. F., CHEN, H. "Data Fusion for Network Intrusion Detection: A Review", *ISecurity and Communication Networks*, v. 2018, jan. 2018.

[31] SHABANIAN, M., HOSSEINI, S. H. "Sensor Data Fusion Using Mutual Information Algorithm", *Ciência e Natura*, v. 37, n. 2, pp. 146–155, 2015.

[32] CASTANEDO, F. "A Review of Data Fusion Techniques", *The Scientific World Journal*, v. 2013, 2013.

[33] FASSINUT-MOMBOT, B., CHOQUEL, J. B. "An entropy method for multi-source data fusion". In: *Proceedings of the Third International Conference on Information Fusion*, v. 2, pp. THC5/17–THC5/23 vol.2, July 2000.

[34] MAHDIEH, S., SEYED, H. "Sensor Data Fusion Using Mutual Information Algorithm", *Ciência e Natura*, v. 37, pp. 146–155, 2015.

[35] SHAFER, G. *A Mathematical Theory of Evidence*. Princeton, Princeton University Press, 1976.

[36] NG, K.-C., ABRAMSON, B. "Uncertainty Management in Expert Systems", *IEEE Expert: Intelligent Systems and Their Applications*, v. 5, n. 2, pp. 29–48, abr. 1990.

[37] JØSANG, A., KAPLAN, L. M. "Principles of subjective networks". In: *19th International Conference on Information Fusion, FUSION 2016, Heidelberg, Germany, July 5-8, 2016*, pp. 1292–1299, 2016.

[38] YAGER, R. R. "Conditional Approach to Possibility-Probability Fusion", *IEEE Transactions on Fuzzy Systems*, v. 20, n. 1, pp. 46–56, Feb 2012.

[39] DEZERT, J., TCHAMOVA, A. "On the Validity of Dempster's Fusion Rule and its Interpretation as a Generalization of Bayesian Fusion Rule", *International Journal of Intelligent Systems*, v. 29, n. 3, pp. 223–252, 2014.

[40] GAUR, A., SCOTNEY, B. W., PARR, G. P., et al. "Evidential Sensor Data Fusion in a Smart City Environment", *OJIOT*, v. 1, n. 2, pp. 1–18, 2015.

[41] FAOUZI, N.-E. E., LEUNG, H., KURIAN, A. "Data Fusion in Intelligent Transportation Systems: Progress and Challenges - A Survey", *Inf. Fusion*, v. 12, n. 1, pp. 4–10, jan. 2011.

[42] REKHTER, Y., HARES, S., LI, T. "A Border Gateway Protocol 4 (BGP-4)". RFC 4271, jan. 2006. Disponível em: <`https://rfc-editor.org/rfc/rfc4271.txt`>.

[43] MAO, Q., HU, F., HAO, Q. "Deep Learning for Intelligent Wireless Networks: A Comprehensive Survey", *IEEE Communications Surveys Tutorials*, v. 20, n. 4, pp. 2595–2621, 2018.

[44] GOODFELLOW, I., BENGIO, Y., COURVILLE, A. *Deep Learning*. The MIT Press, 2016.

[45] SILVA, R. S., MEIXNER, C. C., GUIMARÃES, R. S., et al. "REPEL: A Strategic Approach for Defending 5G Control Plane from DDoS Signalling Attacks", *IEEE Transactions on Network and Service Management*, pp. 1–1, 2020.

[46] BIKOS, A. N., SKLAVOS, N. "LTE/SAE Security Issues on 4G Wireless Networks", *IEEE Security Privacy*, v. 11, n. 2, pp. 55–62, 2013.

[47] ©3GPP. *System architecture for the 5G System (5GS).* Technical Specification (TS) 23501, 3rd Generation Partnership Project (3GPP), 09 2020. Disponível em: <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3144>. Version 16.6.0.

[48] BLAINE, G. "Inside the Mirai Malware That Powers IoT Botnets". Nov 2016. Disponível em: <https://www.a10networks.com/blog/inside-the-mirai-malware-that-powers-iot-botnets/>. (Acessado em: 10 dez. 2020, 13:01:58.).

[49] SHAIK, A., BORGAONKAR, R., ASOKAN, N., et al. "Practical Attacks Against Privacy and Availability in 4G/LTE Mobile Communication Systems", *ArXiv*, v. abs/1510.07563, 2015.

[50] ERNSBERGER, D., GEORGE, K., ARUMUGAM, S. "Security Study and Monitoring of LTE Networks", *Journal of ICT Standardization*, v. 7, pp. 43–60, 01 2019.

[51] ©3GPP. *Evolved Universal Terrestrial Radio Access Network (E-UTRAN); S1 Application Protocol (S1AP).* Technical Specification (TS) 23401, 3rd Generation Partnership Project (3GPP), 09 2018. Disponível em: <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2446>. Version 15.3.0.

[52] KLEINROCK, L. *Theory, Volume 1, Queueing Systems.* New York, NY, USA, Wiley-Interscience, 1975. ISBN: 0471491101.

[53] OSBORNE, M. *An introduction to game theory.* New York, NY [u.a.], Oxford Univ. Press, 2004.

[54] LIANG, X., XIAO, Y. "Game theory for network security", *IEEE Communications Surveys & Tutorials*, v. 15, n. 1, pp. 472–486, 2012.

[55] DASGUPTA, D. "Immunity-based Intrusion Detection System: a General Framework". In: *Proc. of the 22nd NISSC*, v. 1, pp. 147–160, 1999.

[56] KIM, J., BENTLEY, P. J. "Towards an Artificial Immune system for Network Intrusion Detection: an Investigation of Clonal Selection with a Negative Selection Operator". In: *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, v. 2, pp. 1244–1252, 2001.

[57] IGBE, O., DARWISH, I., SAADAWI, T. "Distributed Network Intrusion Detection Systems: An Artificial Immune System Approach". In: *2016 IEEE First International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)*, pp. 101–106, jun 2016.

[58] SNAPP, S. R., BRENTANO, J., DIAS, G. V., et al. "DIDS (Distributed Intrusion Detection System) - Motivation, Architecture, and An Early Prototype". In: *In Proceedings of the 14th National Computer Security Conference*, pp. 167–176, 1991.

[59] CROSBIE, M., SPAFFORD, E. H. "Defending a Computer System Using Autonomous Agents", , n. 95-022, 1995.

[60] BALASUBRAMANIYAN, J. S., GARCIA-FERNANDEZ, J. O., ISACOFF, D., et al. "An Architecture for Intrusion Detection Using Autonomous Agents". In: *Computer security applications conference, 1998. Proceedings. 14th annual*, pp. 13–24. IEEE, 1998.

[61] LABIOD, H., BOUDAOUD, K., LABETOULLE, J. "Towards a New Approach for Intrusion Detection with Intelligent Agents", *NIS, Networking and Information Systems, Ingénierie des systèmes d'informations*, v. 2, n. 5-6, dez. 2000.

[62] NING, P., JAJODIA, S., WANG, X. S. "Abstraction-based Intrusion Detection in Distributed Environments", *ACM Trans. Inf. Syst. Secur.*, v. 4, n. 4, pp. 407–452, nov. 2001.

[63] CUPPENS, F., MIEGE, A. "Alert Correlation in a Cooperative Intrusion Detection Framework". In: *Proceedings 2002 IEEE Symposium on Security and Privacy*, pp. 202–215, 2002.

[64] YEGNESWARAN, V., BARFORD, P., JHA, S. "Global Intrusion Detection in the DOMINO Overlay System". .

[65] JANAKIRAMAN, R., WALDVOGEL, M., ZHANG, Q. "Indra: A Peer-to-Peer Approach to Network Intrusion Detection and Prevention". In: *12th IEEE International Workshops on Enabling Technologies (WETICE 2003), Infrastructure for Collaborative Enterprises, 9-11 June 2003,Linz, Austria*, pp. 226–231, 2003.

[66] DRESSLER, F., MUNZ, G., CARLE, G. "Attack Detection Using Cooperating Autonomous Detections Systems (CATS)", *Wilhelm-Schickard Institute of Computer Science, Computer Networks and Internet*, 2004.

[67] BASICEVIC, I., POPOVIC, M., KOVACEVIC, V. "The Use of Distributed Network-Based IDS Systems in Detection of Evasion Attacks". In: *Telecommunications 2005: Advanced Industrial Conference on Telecommunications Service Assurance with Partial and Intermittent Resources Conference E-Learning on Telecommunications Workshop (AICT / SAPIR / ELETE)*, pp. 78–82, Lisbon, jul. 2005.

[68] ©RIVERBED TECHNOLOGY. "WinPCap - The industry-standard windows packet capture library". aug 2017. Disponível em: <`https://www.winpcap.org/`>. (Acessado em: 11 Jun. 2019, 15:13:12.).

[69] ABRAHAM, A., JAIN, R., THOMAS, J., et al. "D-SCIDS: Distributed Soft Computing Intrusion Detection System", *J. Netw. Comput. Appl.*, v. 30, n. 1, pp. 81–98, jan. 2007.

[70] LUO, Y., XIANG, K., FAN, J., et al. "Distributed Intrusion Detection with Intelligent Network Interfaces for Future Networks". In: *2009 IEEE International Conference on Communications*, pp. 1–5, jun. 2009.

[71] DISSO, J. F. P., JONES, K., WILLIAMS, P., et al. "A Distributed Attack Detection and Mitigation Framework". In: *2011 IEEE 5th International Conference on Internet Multimedia Systems Architecture and Application*, pp. 1–6, dez. 2011.

[72] RAY, S. "Distributed Network Attack Detection System", *International Journal of Engineering Research & Technology - (IJERT)*, v. 3, n. 1, pp. 2520–2523, jan. 2014.

[73] SELLAMI, L., IDOUGHI, D., BAADACHE, A., et al. "A Novel Detection Intrusion Approach for Ubiquitous and Pervasive Environments", *Procedia Computer Science*, v. 94, pp. 429 – 434, 2016.

[74] FERNÁNDEZ MAIMÓ, L., HUERTAS CELDRÁN, A., GIL PÉREZ, M., et al. "Dynamic management of a deep learning-based anomaly detection system for 5G networks", *Journal of Ambient Intelligence and Humanized Computing*, May 2018.

[75] SILVA, R. S., MACEDO, E. L. C. "A cooperative approach for a global intrusion detection system for internet service providers". In: *2017 1st Cyber Security in Networking Conference (CSNet)*, pp. 1–8, 2017.

[76] SILVA, R. S., DE MORAES, L. F. M. "A cooperative approach with improved performance for a global intrusion detection systems for internet service providers", *Ann. des Télécommunications*, v. 74, n. 3-4, pp. 167–173, 2019.

[77] ©BIZATY. "Machine Learning and BGP Anomaly Detection". Jun 2020. Disponível em: <`https://www.bizety.com/2020/06/18/machine-learning-and-bgp-anomaly-detection/`>. (Acessado em: 1 Nov. 2020, 07:50:34.).

[78] KIRKPATRICK, K. "Fixing the Internet", *Commun. ACM*, v. 64, n. 8, pp. 16–17, jul. 2021.

[79] AL-MUSAWI, B., BRANCH, P., ARMITAGE, G. "BGP Anomaly Detection Techniques: A Survey", *IEEE Communications Surveys Tutorials*, v. 19, n. 1, pp. 377–396, 2017.

[80] AL-ROUSAN, N. M., TRAJKOVIĆ, L. "Machine learning models for classification of BGP anomalies". In: *2012 IEEE 13th International Conference on High Performance Switching and Routing*, pp. 103–108, 2012.

[81] ALLAHDADI, A., MORLA, R., PRIOR, R. "A Framework for BGP Abnormal Events Detection", 08 2017.

[82] DAI, X., WANG, N., WANG, W. "Application of machine learning in BGP anomaly detection", v. 1176, pp. 032015, mar 2019.

[83] KARIMI, M., JAHANSHAHI, A., MAZLOUMI, A., et al. "Border Gateway Protocol Anomaly Detection Using Neural Network". In: *2019 IEEE International Conference on Big Data (Big Data)*, pp. 6092–6094, 2019.

[84] ©RIPE NCC. "RIS Raw Data". jul 2021. Disponível em: <`http://data.ris.ripe.net/rrc04/`>. (Acessado em: 13 set. 2021, 16:17:11.).

[85] SANCHEZ, O. R., FERLIN, S., PELSSER, C., et al. "Comparing Machine Learning Algorithms for BGP Anomaly Detection Using Graph Features". Big-DAMA '19, p. 35–41, New York, NY, USA, 2019. Association for Computing Machinery.

[86] TSUDA, K., KAWANABE, M., MÜLLER, K.-R. "Clustering with the Fisher Score". In: Becker, S., Thrun, S., Obermayer, K. (Eds.), *Advances in Neural Information Processing Systems*, v. 15. MIT Press, 2003.

[87] FONSECA, P., MOTA, E. S., BENNESBY, R., et al. "BGP Dataset Generation and Feature Extraction for Anomaly Detection". In: *2019 IEEE Symposium on Computers and Communications (ISCC)*, pp. 1–6, 2019.

[88] DOULIGERIS, C., MITROKOTSA, A. "DDoS attacks and defense mechanisms: classification and state-of-the-art", *Computer Networks*, v. 44, n. 5, pp. 643–666, 2004.

[89] MIRKOVIC, J., REIHER, P. "A Taxonomy of DDoS Attack and DDoS Defense Mechanisms", *SIGCOMM Comput. Commun. Rev.*, v. 34, n. 2, pp. 39–53, abr. 2004.

[90] KHAN, M. A. "A survey of security issues for cloud computing", *Journal of Network and Computer Applications*, v. 71, pp. 11 – 29, 2016.

[91] VASANTHAAZHAGU, A. K., GNANASEKAR, J. M. "Cloud Computing Overview, Security Threats and Solutions-A Survey". ICIA-16, New York, NY, USA, 2016. Association for Computing Machinery.

[92] BANERJEE, A., MAHINDRA, R., SUNDARESAN, K., et al. "Scaling the LTE Control-plane for Future Mobile Access". In: *Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies*, CoNEXT '15, pp. 19:1–19:13, New York, NY, USA, 2015. ACM.

[93] AMOGH, P. C., VEERAMACHANENI, G., RANGISETTI, A. K., et al. "A cloud native solution for dynamic auto scaling of MME in LTE". In: *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 1–7, Oct 2017.

[94] GELENBE, E., ABDELRAHMAN, O. H., GORBIL, G. "Detection and mitigation of signaling storms in mobile networks". In: *2016 International Conference on Computing, Networking and Communications (ICNC)*, pp. 1–5, 2016.

[95] ETTIANE, R., ELKOUCH, R., CHAOUB, A. "Protection mechanisms for signaling DoS attacks on 3G mobile networks: Comparative study and future perspectives". In: *Information Science and Technology (CiSt), 2016 4th IEEE International Colloquium on*, pp. 860–866. IEEE, 2016.

[96] BASSIL, R., CHEHAB, A., ELHAJJ, I., et al. "Signaling Oriented Denial of Service on LTE Networks". In: *Proceedings of the 10th ACM International Symposium on Mobility Management and Wireless Access*, MobiWac '12, pp. 153–158, New York, NY, USA, 2012.

[97] PAVLOSKI, MIHAJLO", E. E., CAMPEGIANI, P., CZACHÓRSKI, T., et al. "Signalling Attacks in Mobile Telephony". In: *Security in Computer and Information Sciences*, pp. 130–141, Cham, 2018. Springer International Publishing.

[98] ESCUDERO-ANDREU, G., KYRIAKOPOULOS, K., FLINT, J. A., et al. "Detecting Signalling DoS Attacks on LTE Networks". In: Duong, T. Q., Vo, N.-S., Nguyen, L. K., et al. (Eds.), *Industrial Networks and Intelligent Systems*, pp. 283–301, Cham, 2019. Springer International Publishing.

[99] SATTAR, D., MATRAWY, A. "Towards Secure Slicing: Using Slice Isolation to Mitigate DDoS Attacks on 5G Core Network Slices", *CoRR*, v. abs/1901.01443, 2019.

[100] KREPS, D. M. "Nash Equilibrium". In: Eatwell, J., Milgate, M., Newman, P. (Eds.), *Game Theory*, pp. 167–177, London, Palgrave Macmillan UK, 1989.

[101] CHANDRASEKAR, K., CLEARY, G., COX, O., et al. *2017 Internet Security Threat Report.* In: Report 22, Symantec, 2017.

[102] KHALIMONENKO, A., KUPREEV, O. *DDOS attacks in Q1 2017.* In: Report Q1 2017, Kaspersky Lab, 2017.

[103] PASSERI, P. *Information Security Timelines and Statistic.* In: Report, Hackmageddon, 2017.

[104] LEINER, B. M., CERF, V. G., CLARK, D. D., et al. "A Brief History of the Internet", *SIGCOMM Comput. Commun. Rev.*, v. 39, n. 5, pp. 22–31, out. 2009.

[105] WONG, A. *Cybersecurity – Threats Challenges Opportunities.* In: Report Q4 2016, ACS - Australian Computer Society, 2016.

[106] BILGE, L., DUMITRAS, T. "Before We Knew It: An Empirical Study of Zero-day Attacks in the Real World". In: *CCS '12 Proceedings of the 2012 ACM conference on Computer and Communications Security*, CCS '12, pp. 833 – 844. ACM, ACM, 2012/// 2012.

[107] TJHAI, G. "Comprehensive Approaches of Intrusion Detection in Handling False Alarm Issue". In: *Proceedings of the Third Collaborative Research Symposium on Security, E-learning, Internet and Networking (SEIN 2007)*, pp. 53–66, 2007.

[108] BABATOPE, L. O., LAWAL, BABATUNDE, et al. "Strategic Sensor Placement for Intrusion Detection in Network-Based IDS", *International Journal of Intelligent Systems and Applications(IJISA)*, v. 6, n. 2, pp. 61–681, 2014.

[109] BASS, T. "Intrusion Detection Systems and Multisensor Data Fusion", *Commun. ACM*, v. 43, n. 4, pp. 99–105, abr. 2000.

[110] SHAH, V., AGGARWAL, A. K., CHAUBEY, N. "Performance improvement of intrusion detection with fusion of multiple sensors", *Complex & Intelligent Systems*, v. 3, n. 1, pp. 33–39, Mar 2017.

[111] PROVOST, F., KOHAVI, R. "On Applied Research in Machine Learning". In: *Machine learning*, pp. 127–132, 1998.

[112] BHUYAN, M. H., BHATTACHARYYA, D. K., KALITA, J. K. "Towards Generating Real-life Datasets for Network Intrusion Detection", *I. J. Network Security*, v. 17, pp. 683–701, 2015.

[113] N. ALVES JR., M. P. D. A., DE ALBUQUERQUE, M. P., DE ASSIS, J. T. "Submetido para TEMA Topology and Shortest Path Length Evolution of The Internet Autonomous Systems Interconnectivity". 2007.

[114] WANG, C., LI, Z., HUANG, X., et al. "Inferring the Average as Path Length of the Internet". In: *2016 IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC)*, pp. 391–395, Sept 2016.

[115] CHANG, W., MOHAISEN, A., WANG, A., et al. "Measuring Botnets in the Wild: Some New Trends", *ASIACCS 2015 - Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*, pp. 645–650, 04 2015.

[116] JØSANG, A. *Subjective Logic: A Formalism for Reasoning Under Uncertainty*. Springer Publishing Company, Incorporated, 2016.

[117] JAYNES, E. T. "Prior Probabilities", *IEEE Transactions on Systems Science and Cybernetics*, v. 4, n. 3, pp. 227–241, Sep. 1968.

[118] GHARIB, A., SHARAFALDIN, I., LASHKARI, A. H., et al. "An Evaluation Framework for Intrusion Detection Dataset". In: *2016 International Conference on Information Science and Security (ICISS)*, pp. 1–6, 2016.

[119] KAK, A. "ML, MAP, and Bayesian — The Holy Trinity of Parameter Estimation and Data Prediction". January 2017. Disponível em: <`https://engineering.purdue.edu/kak/Trinity.pdf`>. Purdue University Tutorial.

[120] SYVERSVEEN, A. "Noninformative Bayesian Priors. Interpretation And Problems With Construction And Applications." 03 1998.

[121] JAMIESON, K. G., GUPTA, M. R., KROUT, D. W. "Sequential Bayesian estimation of the probability of detection for tracking". In: *2009 12th International Conference on Information Fusion*, pp. 641–648, 2009.

[122] CASWELL, B., FOSTER, J. C., RUSSELL, R., et al. *Snort 2.0 Intrusion Detection*. Syngress Publishing, 2003. ISBN: 1931836744.

[123] MOUSTAFA, N., SLAY, J. "The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set", *Information Security Journal: A Global Perspective*, v. 25, n. 1-3, pp. 18–31, 2016.

[124] FUNG, C. J., BOUTABA, R. *Intrusion Detection Networks - A Key to Collaborative Security*. CRC Press, 2013.

[125] NEUMANN, J. C. *The Book of GNS3*. 1st ed. San Francisco, CA, USA, No Starch Press, 2014.

[126] NATH, A. *Packet Analysis with Wireshark*. Packt Publishing, 2015.

[127] BOUGUEROUA, N., MAZOUZI, S., BELAOUED, M., et al. "A Survey on Multi-Agent Based Collaborative Intrusion Detection Systems", *Journal of Artificial Intelligence and Soft Computing Research*, v. 11, n. 2, pp. 111–142, 2021.

[128] KHORASANIZADEH, H., IDRIS, N. B., MANAN, J. A. "Distributed Intrusion Detection trust management through integrity and expertise evaluation". In: *2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensic, CyberSec 2012, Kuala Lumpur, Malaysia, June 26-28, 2012*, pp. 133–138. IEEE, 2012.

[129] CORONA, I., GIACINTO, G., ROLI, F. "Adversarial attacks against intrusion detection systems: Taxonomy, solutions and open issues", *Information Sciences*, v. 239, pp. 201–225, 2013.

[130] NORDSTRÖM, O., DOVROLIS, C. "Beware of BGP attacks", *ACM SIGCOMM Computer Communication Review*, v. 34, n. 2, pp. 1–8, 2004.

[131] MARCOS, P., PREHN, L., LEAL, L., et al. "AS-Path Prepending: There is No Rose without a Thorn". IMC '20, p. 506–520, New York, NY, USA, 2020. Association for Computing Machinery.

[132] WANG, C., LI, Z., HUANG, X., et al. "Inferring the average as path length of the Internet". In: *2016 IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC)*, pp. 391–395, 2016.

[133] ©CAIDA. "CAIDA AS Rank". Disponível em: <http://as-rank.caida.org/>. (Acessado em: 21 Set. 2021, 09:26:12.).

[134] WAGNER, C., FRANÇOIS, J., STATE, R., et al. "ASMATRA: Rankin g ASs Providing Transit Service to Malware Hosters". In: *International Symposium on Integrated Network Management*, Proceedings of the 13th IFIP/IEEE International Symposium on Integrated Network Management, Ghent, Belgium, maio 2013. IEEE.

[135] SRIRAM, K., MONTGOMERY, D., BORCHERT, O., et al. "Study of BGP Peering Session Attacks and Their Impacts on Routing Performance", 2006-10-01 2006. doi: https://doi.org/10.1109/JSAC.2006.877218.

[136] FACELI, K., LORENA, A. C., GAMA, J., et al. *Inteligência artificial: uma abordagem de aprendizado de máquina*. LTC, 2011.

[137] MURTAGH, F., LEGENDRE, P. "Ward's Hierarchical Agglomerative Clustering Method: Which Algorithms Implement Ward's Criterion?" *J. Classif.*, v. 31, n. 3, pp. 274–295, out. 2014.

[138] FONSECA, P., MOTA. "BGP Feature Extractor". 2020. Disponível em: <https://github.com/ufam-lia/bgp-feature-extractor>. (Acessado em: 3 set. 2021, 18:17:11.).

[139] XU, Y., GOODACRE, R. "On splitting training and validation set: a comparative study of cross-validation, bootstrap and systematic sampling for estimating the generalization performance of supervised learning", *Journal of Analysis and Testing*, v. 2, n. 3, pp. 249–262, 2018.

[140] HANDLEY, L. "Nearly three quarters of the world will use just their smartphones to access the internet by 2025". Jan 2019. Disponível em: <smartphones-72percent-of-people-will-use-only-mobile-for-internet>. (Acessado em: 27 jul. 2020, 16:28:12.).

[141] ©ERICSSON. "In 2025, 5G networks will carry nearly half of the world's mobile data traffic". Apr 2020. Disponível em: <www.ericsson.com/en/mobility-report/reports/june-2020/mobile-data-traffic-outlook>. (Acessado em: 08 abr. 2020, 14:07:44.).

[142] HAN, Q., CHO, D. "Characterizing the technological evolution of smartphones: insights from performance benchmarks". In: *Proceedings of the 18th Annual International Conference on Electronic Commerce: e-Commerce in Smart connected World*, p. 32. ACM, 2016.

[143] AGIWAL, M., ROY, A., SAXENA, N. "Next generation 5G wireless networks: a comprehensive survey", *IEEE Communications Surveys Tutorials*, v. 18, n. 3, pp. 1617–1655, thirdquarter 2016.

[144] SHAFI, M., MOLISCH, A. F., SMITH, P. J., et al. "5G: A tutorial overview of standards, trials, challenges, deployment, and practice", *IEEE Journal on Selected Areas in Communications*, v. 35, n. 6, pp. 1201–1221, June 2017.

[145] PASSERI, P. "Information Security Timelines and Statistics". Oct 2018. Disponível em: <`https://www.hackmageddon.com/2018-master-table/`>. (Acessado em: 15 mai. 2021, 17:37:31.).

[146] SCHNEIDER, P., HORN, G. "Towards 5G Security". In: *2015 IEEE Trustcom/BigDataSE/ISPA*, v. 1, pp. 1165–1170, Aug 2015.

[147] FORSBERG, D., HORN, G., MOELLER, W.-D., et al. *LTE Security*. Wiley Publishing, 2012.

[148] KHAN, R., KUMAR, P., JAYAKODY, D. N. K., et al. "A Survey on Security and Privacy of 5G Technologies: Potential Solutions, Recent Advancements and Future Directions", *IEEE Communications Surveys & Tutorials*, pp. 1–52, July 2019.

[149] LEE, P. P. C., BU, T., WOO, T. "On the Detection of Signaling DoS Attacks on 3G Wireless Networks". In: *IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications*, pp. 1289–1297, May 2007.

[150] RATHGEB, E. P., HOHENDORF, C., NORDHOFF, M. "On the Robustness of SCTP against DoS Attacks". In: *2008 Third International Conference on Convergence and Hybrid Information Technology*, v. 2, pp. 1144–1149, Nov 2008.

[151] VIDAL, J. M., OROZCO, A. L. S., VILLALBA, L. J. G. "Mitigation of DDoS Attacks in 5G Networks: a Bio-inspired Approach", *Proc. 2nd IEEE European Symposium on Security and Privacy (EuroS&P)*, pp. 1–2, 04 2017.

[152] HENRYDOSS, J., BOULT, T. "Critical security review and study of DDoS attacks on LTE mobile network". In: *2014 IEEE Asia Pacific Conference on Wireless and Mobile*, pp. 194–200, Aug 2014.

[153] JANG, W., KIM, S. K., OH, J. H., et al. "Session-based detection of signaling DoS on LTE mobile networks", *Journal of Advances in Computer Networks*, v. 2, n. 3, 2014.

[154] JOVER, R. P. "Security attacks against the availability of LTE mobility networks: Overview and research directions". In: *2013 16th International Symposium on Wireless Personal Multimedia Communications (WPMC)*, pp. 1–9, June 2013.

[155] EINSIEDLER, H., GAVRAS, A., SELLSTEDT, P., et al. "System design for 5G converged networks". In: *2015 European Conference on Networks and Communications (EuCNC)*, pp. 391–396, June 2015.

[156] ©3GPP ORGANIZATIONAL PARTNERS. *3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; 3GPP System Architecture Evolution; CT WG1 Aspects.* Relatório Técnico 8, 3GPP, 12 2008.

[157] MAVOUNGOU, S., KADDOUM, G., TAHA, M., et al. "Survey on Threats and Attacks on Mobile Networks", *IEEE Access*, v. 4, pp. 4543–4572, 2016.

[158] BANG, J.-H., CHO, Y.-J., KANG, K. "Anomaly Detection of Network-initiated LTE Signaling Traffic in Wireless Sensor and Actuator Networks Based on a Hidden semi-Markov Model", *Comput. Secur.*, v. 65, n. C, pp. 108–120, mar. 2017.

[159] ESCUDERO-ANDREU, G., KYRIAKOPOULOS, K., FLINT, J. A., et al. "Detecting Signalling DoS Attacks on LTE Networks". In: Duong, T. Q., Vo, N.-S., Nguyen, L. K., et al. (Eds.), *Industrial Networks and Intelligent Systems*, pp. 283–301, Cham, 2019. Springer International Publishing.

[160] GUPTA, A., JHA, R. K., JAIN, S. "Attack modeling and intrusion detection system for 5G wireless communication network", *International Journal of Communication Systems*, v. 30, n. 10, pp. e3237, 2017. e3237 IJCS-16-0396.R1.

[161] ©3GPP. *5G; System Architecture for the 5G System.* Technical Specification (TS) 23.501, 3rd Generation Partnership Project (3GPP), 06 2018. Disponível em: <`https://www.etsi.org/deliver/etsi_ts/123500_123599/123501/15.02.00_60/ts_123501v150200p.pdf`>. Version 15.2.0.

[162] ©OOKLA, L. "Downdetector". Mai 2019. Disponível em: <`https://https://downdetector.com/archive/`>. (Acessado em: 16 mar. 2020, 20:04:58.).

[163] SPYRIDOPOULOS, T., KARANIKAS, G., TRYFONAS, T., et al. "A game theoretic defence framework against DoS/DDoS cyber attacks", *Computers & Security*, v. 38, pp. 39 – 50, 2013.

[164] BEDI, H. S., ROY, S., SHIVA, S. "Game theory-based defense mechanisms against DDoS attacks on TCP/TCP-friendly flows". In: *2011 IEEE Symposium on Computational Intelligence in Cyber Security (CICS)*, pp. 129–136, April 2011.

[165] HUANG, L., FENG, D., LIAN, Y., et al. "A game theory based approach to the generation of optimal DDOS defending strategy". In: *Proc. Int. Conf. Comput. Security Digit. Invest.(ComSec)*, pp. 14–20, 2014.

[166] ATTIAH, A., CHATTERJEE, M., ZOU, C. C. "A game theoretic approach to model cyber attack and defense strategies". In: *2018 IEEE International Conference on Communications (ICC)*, pp. 1–7. IEEE, 2018.

[167] WU, Q., SHIVA, S., ROY, S., et al. "On Modeling and Simulation of Game Theory-based Defense Mechanisms Against DoS and DDoS Attacks". In: *Proceedings of the 2010 Spring Simulation Multiconference*, SpringSim '10, pp. 159:1–159:8, San Diego, CA, USA, 2010. Society for Computer Simulation International.

[168] WANG, Y., MA, J., ZHANG, L., et al. "Dynamic game model of botnet DDoS attack and defense", *Security and Communication Networks*, v. 9, n. 16, pp. 3127–3140, 2016.

[169] KUMAR, B., BHUYAN, B. "Using game theory to model DoS attack and defence", *Sādhanā*, v. 44, n. 245, 2019.

[170] PUTMAN, C., ABHISHTA, A., NIEUWENHUIS, L. "Business Model of a Botnet". In: Kotenko, I., Merelli, I., Lio, P. (Eds.), *2018 26th Euromicro International conference on Parallel, Distributed, and Network-Based Processing (PDP)*, pp. 441 – 445. IEEE, jun 2018.

[171] DANO, M. "The Android IM app that brought T-Mobile's network to its knees". Oct 2020. Disponível em: <https://www.fiercewireless.com/wireless/android-im-app-brought-t-mobile-s-network-to-its-knees>. (Acessado em: 14 jul. 2021, 11:54:58.).

[172] NIKAEIN, N., MARINA, M. K., MANICKAM, S., et al. "OpenAirInterface: A Flexible Platform for 5G Research", *SIGCOMM Comput. Commun. Rev.*, v. 44, n. 5, pp. 33–38, out. 2014.

[173] SHRIVASTWA, A., SARAT, S., JACKSON, K., et al. *OpenStack: Building a Cloud Environment*. Packt Publishing, 2016. ISBN: 1787123189.

[174] ©3GPP. *Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS)*. Technical Specification (TS) 124.301, 3rd Generation Partnership Project (3GPP), 09 2009. Disponível em: <https://www.etsi.org/deliver/etsi_ts/124300_124399/124301/08.03.00_60/ts_124301v080300p.pdf>. Version 8.3.0.3.

[175] PRADOS-GARZON, J., RAMOS-MUNOZ, J. J., AMEIGEIRAS, P., et al. "Modeling and Dimensioning of a Virtualized MME for 5G Mobile Networks", *IEEE Transactions on Vehicular Technology*, v. 66, n. 5, pp. 4383–4395, May 2017.

[176] BASSIL, R., ELHAJJ, I. H., CHEHAB, A., et al. "Effects of Signaling Attacks on LTE Networks". In: *2013 27th International Conference on Advanced Information Networking and Applications Workshops*, pp. 499–504, March 2013.

[177] JERMYN, J., SALLES-LOUSTAU, G., ZONOUZ, S. "An analysis of dos attack strategies against the lte ran", *Journal of Cyber Security and Mobility*, v. 3, n. 2, pp. 159–180, 2014.

[178] SHAH, S., ISSAC, B. "Performance Comparison of Intrusion Detection Systems and Application of Machine Learning to Snort System", *Future Generation Computer Systems*, v. 80, pp. 157–170, 03 2018.

[179] AZMI, R., PISHGOO, B. "SHADuDT: Secure hypervisor-based anomaly detection using danger theory", *Computers & Security*, v. 39, pp. 268 – 288, 2013.

[180] YILMA, G. M., YOUSAF, F. Z., SCIANCALEPORE, V., et al. "On the challenges and KPIs for benchmarking open-source NFV MANO systems: OSM vs ONAP", *arXiv preprint arXiv:1904.10697*, 2019.

[181] ASSIS, F., COUTINHO, M., FILHO, J. S., et al. "IPTraf: Coleta e Detecção de Anomalias em Fluxos de Rede". In: *Anais do XXVI Workshop de Gerência e Operação de Redes e Serviços*, pp. 96–109, Porto Alegre, RS, Brasil, 2021. SBC.

Apêndice A

Cálculo da Massa Combinada

A Tabela A.1 ajuda a entender a aplicação da Equação 4.6 para obter o nível de crença combinada $m_C(I{=}1|U{=}1)$ a partir de 3 fontes de evidência $m_1(I{=}1|U{=}1)$, $m_2(I{=}1|U{=}1)$ and $m_3(I{=}1|U{=}1)$.

Tabela A.1: Cálculo do nível de crença da mensagem combinada pela Equação 4.6 para 3 fontes de evidência independentes $m_1(I{=}1|U{=}1)$, $m_2(I{=}1|U{=}1)$ e $m_3(I{=}1|U{=}1)$.

Fusion 1	$m_1(I{=}1\|U{=}1){=}PPV_1$	$m_1(\Omega){=}1{-}PPV_1$
$m_2(I{=}1\|U{=}1){=}PPV_2$	$PPV_1{\times}PPV_2$	$PPV_2(1{-}PPV_1)$
$m_2(\Omega){=}1{-}PPV_2$	$PPV_1(1{-}PPV_2)$	$(1{-}PPV_2)(1{-}PPV_1)$
Fusion 2	$m_C(I{=}1\|U{=}1){=}\alpha$	$m_C(\Omega){=}\beta$
$m_3(I{=}1\|U{=}1){=}PPV_3$	$\alpha{\times}PPV_3$	$\beta{\times}PPV_3$
$m_3(\Omega){=}1{-}PPV_3$	$\alpha(1{-}PPV_3)$	$\beta(1{-}PPV_3)$

A notação utilizada na Tabela A.1 está definida abaixo.

$m_C(I{=}1|U{=}1){=}PPV_1{\times}PPV_2{+}PPV_2(1{-}PPV_1){+}PPV_1(1{-}PPV_2){=}\alpha$
$m_C(\Omega){=}(1{-}PPV_2)(1{-}PPV_1){=}\beta$
$m_{C_1}(I{=}1|U{=}1){=}\alpha{\times}PPV_3{+}\beta{\times}PPV_3{+}\alpha(1{-}PPV_3)$
$m_{C_1}(\Omega){=}\beta(1{-}PPV_3)$

Apêndice B

Códigos dos Modelos de Aprendizado de Máquina

Este apêndice mostra as principais linhas de comando dos códigos utilizados na construção do modelo de aprendizado de máquina proposto no Capítulo 5. Os códigos completos, bem como o *dataset* utilizado para treinar o modelo de aprendizado de máquina pode ser obtido em https://github.com/renatossilva/DoctoralThesis.

B.1 Código Aglomeração K-médias

Listing B.1: Linhas de código Python do algoritmo de aglomeração 2-médias.

```
from sklearn.cluster import AgglomerativeClustering
import scipy.cluster.hierarchy as sch

cluster_2 = AgglomerativeClustering(n_clusters=2, affinity='euclidean',
linkage='ward')
cluster_2.fit_predict(X.loc[:, chosen_cols].values)
```

B.2 Código Aglomeração Hierárquica

Listing B.2: Linhas de código Python do algoritmo de aglomeração hierárquica.

```
from sklearn.cluster import AgglomerativeClustering
import scipy.cluster.hierarchy as sch

dendrogram = sch.dendrogram(sch.linkage(X_both_sample, method='ward'))
```

B.3 Código Rede Neural

Listing B.3: Linhas de código Python do algoritmo para a escolha do melhor modelo de aprendizado.

```
from tensorflow import keras

def model_builder(hp):

  dense_outer = hp.Int('units_outer', min_value=16, max_value=128, step=16)
  dense_inner = hp.Int('units_inner', min_value=16, max_value=128, step=16)
  dense_act = hp.Choice('dense_activation', values=['relu', 'tanh', 'sigmoid'],
                        default='relu')
  dropout_outer = hp.Choice('dropout_outer', values=[0.1, 0.2, 0.3, 0.4, 0.5])
  dropout_inner = hp.Choice('dropout_inner', values=[0.1, 0.2, 0.3, 0.4, 0.5])
  l_rate = hp.Choice('learning_rate',
                     values = [1e-2, 1.5e-2, 1e-3, 1.5e-3, 1e-4, 1.5e-4])

  model = keras.Sequential([keras.layers.Dense(dense_outer,
                                               activation=dense_act,
                                               input_shape=(X_train.shape[-1],)),
                            keras.layers.Dropout(dropout_outer),
                            keras.layers.Dense(dense_inner,
                                               activation=dense_act),
                            keras.layers.Dropout(dropout_inner),
                            keras.layers.Dense(1,
                                               activation='sigmoid')])
  model.compile(
      optimizer=keras.optimizers.Adam(learning_rate=l_rate),
      loss=keras.losses.BinaryCrossentropy(),
      metrics=METRICS)

  return model
```