Universidade Estadual de Campinas
Faculdade de Engenharia Elétrica e de Computação
DECOM – Departamento de Comunicações

INFERINDO A FONTE E O DESTINO DO TRÁFEGO ANÔMALO EM REDES DE COMPUTADORES USANDO CORRELAÇÃO ESPAÇO-TEMPORAL

Autor(a): Alexandre de Aguiar Amaral

Orientador: Prof. Dr. Leonardo de Souza Mendes

Co-Orientador: Prof. Dr. Mario Lemes Proença Junior

Dissertação de Mestrado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para a obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: **Telecomunicações e Telemática.**

Banca Examinadora

Prof. Dr. Leonardo de Souza Mendes — DECOM/FEEC/UNICAMP
Prof. Dr. Rodolfo Miranda de Barros — DC/UEL
Prof. Dr. Maurício Ferreira Magalhães — DCA/FEEC/UNICAMP

Campinas – SP
Agosto/2011

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

Am13i

Amaral, Alexandre de Aguiar
Inferindo a fonte e o destino do tráfego anômalo em redes de computadores usando correlação espaço-temporal / Alexandre de Aguiar Amaral. --Campinas, SP: [s.n.], 2011.

Orientadores: Leonardo de Souza Mendes, Mario Lemes Proença Junior.
Dissertação de Mestrado - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Redes de computação – Administração. 2. Anomalias. 3. Alarmes. 4. Telecomunicações - Tráfego. I. Mendes, Leonardo de Souza. II. Proença Junior, Mário Lemes. III. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. IV. Título.

Título em Inglês: Inferring the source and destination of the anomalous traffic in networks using spatio-temporal correlation
Palavras-chave em Inglês: Computer networks - Management, Anomaly, Alarm, Telecommunication – Traffic
Área de concentração: Telecomunicações e Telemática
Titulação: Mestre em Engenharia Elétrica
Banca examinadora: Rodolfo Miranda de Barros, Maurício Ferreira Magalhães
Data da defesa: 24-08-2011
Programa de Pós Graduação: Engenharia Elétrica

COMISSÃO JULGADORA - TESE DE MESTRADO

Candidato: Alexandre de Aguiar Amaral

Data da Defesa: 24 de agosto de 2011

Título da Tese: "Inferindo a fonte e o destino do tráfego anômalo em redes de computadores usando correlação espaço-temporal"

Prof. Dr. Leonardo de Souza Mendes (Presidente): _____

Prof. Dr. Rodolfo Miranda de Barros: _____

Prof. Dr. Maurício Ferreira Magalhães: _____

Resumo

Estratégias voltadas para a detecção de anomalias em redes de computadores emitem alarmes como forma de notificação ao administrador de rede. Esses alarmes são essenciais para a gerência de rede, pois são evidências de uma anormalidade. Entretanto, uma única anomalia pode gerar um número excessivo de alarmes, tornando a inspeção manual inviável. Nesta dissertação, é apresentado um sistema de correlação de alarmes automatizado, divido em três camadas, que obtém os alarmes primitivos e apresenta ao administrador de rede uma visão global do cenário afetado pela anomalia. A camada de pré-processamento faz a compressão dos alarmes utilizando seus atributos espaciais e temporais, os quais são reduzidos a um único alarme denominado DLA (Alarme em Nível de Equipamento). A camada de correlação busca, através dos DLAs e de informações sobre a topologia da rede, inferir o caminho de propagação da anomalia, sua origem e destino. A camada de apresentação provê a visualização do caminho e elementos de redes afetados pela propagação da anomalia. O sistema apresentado nesta dissertação foi aplicado em diversos cenários que apresentavam anomalias reais detectadas na rede da Universidade Estadual de Londrina. Foi demonstrada sua capacidade de identificar, de forma automatizada, o caminho de propagação do tráfego anômalo, proporcionando informações úteis e corretas ao administrador de rede para o diagnóstico do problema.

Palavras-chave: Redes de computação – Gerência, Anomalias, Alarmes, Telecomunicações - Tráfego.

Abstract

Anomaly detection systems for computer networks send alarms in order to notify the network administrator. These alarms are essential for network management because they are evidences of an abnormality. However, a single anomaly may generate an excessive volume of alarms, making the manual inspection unfeasible. In this work, it is presented an automated alarm correlation system divided into three layers, which obtains raw alarms and presents to network administrator a global view of the scenario affected by the anomaly. In the preprocessing layer, it is performed the alarm compression using their spatial and temporal attributes, which are reduced to a unique alarm named DLA (Device Level Alarm). The correlation layer aims to infer the anomaly propagation path and its origin and destination using DLAs and network topology information. The presentation layer provides the visualization of the path and network elements affected by the anomaly propagation through the network. The presented system was applied in various scenarios that had real anomalies detected on the State University of Londrina network. It demonstrated its ability to identify in an automated manner the anomalous traffic propagation path, providing useful and accurate information to the network administrator to diagnose the problem.

Keywords: Computer networks - Management, Anomaly, Alarm, Telecommunication – Traffic.

Aos meus pais, José e Aparecida

Agradecimentos

Ao grande Deus pela vida, saúde e força.

Aos meus inestimáveis pais, José e Aparecida, dois anjos que lutaram arduamente para que hoje eu pudesse estar aqui. Jamais me esquecerei de tudo que vocês fizeram por mim. Vocês são minha maior fonte de inspiração.

Ao meu irmão Claudinei, a minha cunhada Rosângela e aos meus lindos sobrinhos Larissa Milena e Kauã Felipe, amo vocês.

Ao tio Odair por me acolher quando cheguei a Campinas. Somos muito gratos a ti.

A minha querida namorada Ana Paula Malheiro pelo carinho, apoio e compreensão em todo tempo. Obrigado pelas dicas e sugestões que contribuíram muito para esse trabalho.

Ao meu orientador Prof. Leonardo de Souza Mendes pela confiança, orientação e as oportunidades concedidas durante esses anos.

Ao meu co-orientador Prof. Mário Lemes Proença Jr. pela oportunidade de estarmos trabalhando durante todos esses anos, pelas orientações e por me ajudar chegar até aqui.

Ao amigo Bruno Zarpelão pelas correções e pelas ricas dicas e sugestões. Muito obrigado, cresci muito contigo.

Aos amigos do LaRCom, Rodrigo Miani, Dherik, Eduardo Zanoni, Ricardo Tajiri, Felipe, Márlon, Liniquer, Everton, André Panhan e Henrique pelos momentos e convivência.

A amiga Marta pelas sugestões e dicas para o texto desse trabalho, muito grato.

Aos amigos da casa M-09, Thiagão e André pelos momentos legais que temos vividos durante esses anos de mestrado.

Sumário

Lista de Figuras xvii
Lista de Tabelas xxi
Lista de Abreviações xxiii
Lista de Símbolos xxvii
Capítulo 1 31
Introdução 31
1.1 Estrutura da dissertação 37
Capítulo 2 39
Anomalias 39
2.1 Definições e classificação das anomalias 39
2.1.1 Monitoramento ativo 39
2.1.2 Monitoramento passivo 40
2.1.3 Anomalia de volume 41
2.1.3.1 Comportamento não-usual 41
2.1.3.2 Comportamento malicioso 42
2.2 Métodos de detecção de anomalias 46
Capítulo 3 51
Correlação de alarmes 51
3.1 Alarmes 51
3.1.1 Classificação dos alarmes 52
3.1.2 Atributos de um alarme 53
3.1.3 Metadados 53
3.2 Correlação de Alarmes 54
3.2.1 Tipos de operação 56
3.2.1.1 Compressão de alarmes 56
3.2.1.2 Supressão seletiva 56

3.2.1.3 Filtragem 57
3.2.1.4 Generalização 57
3.2.1.5 Especialização 58
3.2.1.6 Contagem 58
3.2.1.7 Escalação 58
3.2.1.8 Priorização 59
3.2.1.9 Clusterização 59
3.2.1.10 Normalização 60
3.2.2 Resumo das operações de correlação 61
3.3 Métodos para correlação de alarmes 62
3.3.1 Correlação baseada em regras 62
3.3.2 Correlação baseada em *codebook* 62
3.3.3 Correlação baseada em casos 64
3.3.4 Redes neurais 65
3.3.5 Grafo de dependência 65
3.3.6 Rede Bayesiana 66
3.3.7 Soluções híbridas 66
Capítulo 4 69
Trabalhos relacionados 69
4.1 Redução de alarmes 69
4.2 Localização da causa raiz 71
4.3 Entropia 72
4.4 Tomografia de rede 74
Capítulo 5 79
Proposta para correlação de alarmes 79
5.1 Sistema de correlação de alarmes 79
5.2 Sistema gerador de alarmes 81

5.3 Camada de pré-processamento 84
5.3.1 Classificação dos alarmes para identificar a propagação da anomalia 85
5.3.2 Redução do volume de alarmes 85
5.4 Camada de correlação 89
5.4.1 Modelagem da rede 89
5.4.2 Matriz de dependência 90
5.4.3 Medida de incerteza 92
5.4.4 Cálculo da probabilidade de propagação da anomalia nos enlaces 98
5.4.5 Identificando a origem e destino do tráfego anômalo 98
5.5 Camada de apresentação 101
Capítulo 6 103
Implementação e resultados 103
6.1 Infraestrutura de monitoramento 103
6.2 Implementação na camada de pré-processamento 105
6.3 Ferramenta APV 107
6.4 Ambiente de rede monitorado 109
6.5 Resultados 111
6.5.1 Estudo de caso 1 111
6.5.2 Estudo de caso 2 115
6.5.3 Estudo de caso 3 117
6.5.4 Estudo de caso 4 122
6.5.5 Estudo de caso 5 127
Conclusão 131
Referências Bibliográficas 135

Lista de Figuras

Figura 2.1 – Estatística de ocorrências de ataques de negação de serviço. Fonte [15]. 43
Figura 2.2 - Ataque de negação de serviço. 44
Figura 2.3 - Estabelecimento de uma conexão TCP. 45
Figura 3.1 – Fragmento contendo exemplos de alarmes. 52
Figura 4.1 – Funcionamento dos métodos de tomografia de rede 75
Figura 5.1 – Visão geral do sistema de correlação. 80
Figura 5.2 – Modo de funcionamento do sistema de alarmes. 82
Figura 5.3 - Diagrama de atividades para o algoritmo de histerese [13]. 83
Figura 5.4 – Evolução dos trabalhos. 84
Figura 5.5 - Classificação dos alarmes gerados para os objetos SNMP. 85
Figura 5.6 - Exemplos de DLAs. 88
Figura 5.7 – Grafo de dependência com probabilidades nas arestas. 90
Figura 5.8 – Rede afetada por uma atividade anômala originada na Internet que tinha como destino o servidor *web*. 92
Figura 5.9 - (a) Um cenário com dois possíveis destinos e (b) um cenário com duas possíveis fontes causados por alarmes espúrios. 93
Figura 5.10 – Entropia de Shannon x probabilidade. 95
Figura 5.11 – Grafo ponderado com dois possíveis caminhos. 96
Figura 5.12 - Entropia de Tsallis x Probabilidade com $q = 2$. 97
Figura 5.13 – Possíveis cenários causados por alarmes espúrios. 99
Figura 6.1 – Módulos da ferramenta GBA. 103
Figura 6.2 – Implementação das camadas do sistema de correlação e interação com a ferramenta GBA. 106
Figura 6.3 - Diagrama de componentes da ferramenta GBA com os novos módulos. 107
Figura 6.4 – Ferramenta APV. (1) barra de Ferramenta, (2) área de plotagem, (3) tabela com informações detalhadas sobre o evento anômalo. 109

Figura 6.5 - Rede monitorada da Universidade Estadual de Londrina. 109
Figura 6.6 – Estudo de caso 1: Alarmes emitidos para *S1.* 111
Figura 6.7 – Estudo de caso 1: Alarmes emitidos para o *Firewall.* 112
Figura 6.8 – Estudo de caso 1: Alarmes emitidos para *S2.* 112
Figura 6.9 – Estudo de caso 1: Alarmes emitidos para o servidor *Proxy.* 112
Figura 6.10 – Estudo de caso 1: Alarmes emitidos para *S2.* 112
Figura 6.11 – DLAs do estudo de caso 1 113
Figura 6.12 - Visão global da rede afetado pela anomalia do estudo de caso 1 114
Figura 6.13 – Estudo de caso 2: Alarmes emitidos para *switch S1.* 115
Figura 6.14 – Estudo de caso 2: Alarmes emitidos para o *Firewall.* 115
Figura 6.15 – Estudo de caso 2: Alarmes emitidos para *S2* 115
Figura 6.16 - Estudo de caso 2: Alarmes emitidos para o *S2* indicando a propagação para o servidor *Proxy* 116
Figura 6.17 - Estudo de caso 2: Alarmes emitidos para o *S2,* notificando a propagação para o servidor *Web*. 116
Figura 6.18 – DLAs do estudo de caso 2 116
Figura 6.19 – Cenário de rede afetado pela anomalia no estudo de casos 2. 117
Figura 6.20 – Estudo de caso 3: Alarmes emitidos para *S2* indicando que a anomalia partiu de *S3.* 118
Figura 6.21 – Estudo de caso 3: Alarmes emitidos para o servidor *Proxy.* 118
Figura 6.22 – Estudo de caso 3: Alarmes emitidos para *S2* indicando que o tráfego anômalo partiu do servidor *Proxy.* 118
Figura 6.23 – Estudo de caso 3: Alarmes emitidos para *S2* indicando que o tráfego anômalo foi para o *Firewall*. 119
Figura 6.24 – Estudo de caso 3: Alarmes emitidos para o equipamento *Firewall.* 119
Figura 6.25 – Estudo de caso 3: Alarmes emitidos para *S1.* 119
Figura 6.26 – DLAs do estudo de caso 3 119
Figura 6.27 - Cenário com duas prováveis fontes de anomalia. 120

Figura 6.28 – Informações dos caminhos gerados pelo sistema de correlação para o caso de estudo 3. 121
Figura 6.29 – Visão holística da rede afetada pelo evento anômalo do estudo de caso 3. 122
Figura 6.30 – Estudo de caso 4: Alarmes emitidos para *S1.* 123
Figura 6.31 – Estudo de caso 4: Alarmes emitidos para *S2,* na porta 3001, indicando que o tráfego anômalo partiu do *Firewall.* 123
Figura 6.32 – Estudo de caso 4: Alarmes emitidos para *S2* na porta 3011. 123
Figura 6.33 – Estudo de caso 4: Alarmes emitidos para *S2* na porta 3015. 123
Figura 6.34 – Estudo de caso 4: Alarmes emitidos para *S2,* na porta 5001. 124
Figura 6.35 – DLAs do estudo de caso 4. 124
Figura 6.36 - Cenário com duas prováveis fontes de anomalia do estudo de caso 4. 125
Figura 6.37 – Informações dos caminhos gerados pelo sistema de correlação para o caso de estudo 4. 126
Figura 6.38 – Visão holística da rede afetada pelo evento anômalo do estudo de caso 4. 126
Figura 6.39 – Equipamentos e caminho de propagação do tráfego anômalo. 127
Figura 6.40 – Estudo de caso 5: Alarmes emitidos para *S2* indicando que partiu da sub-rede ATI. 128
Figura 6.41– Estudo de caso 5: Alarmes emitidos para *S2* sendo um indicador de que o tráfego está partindo de *S2* para o *Firewall.* 128
Figura 6.42– Estudo de caso 5: Alarmes emitidos para o *Firewall.* 128
Figura 6.43– Estudo de caso 5: Alarmes emitidos para *S1.* 128
Figura 6.45 – Visão global da rede do estudo de caso 5. 129
Figura 6.44 – DLAs do estudo de caso 5. 129

Lista de Tabelas

Tabela 3.1 – Resumo dos principais tipos de operações utilizados na correlação de alarmes. 61

Tabela 5.1 - Algoritmo para traçar o caminho percorrido pela anomalia. 101

Tabela 6.1 – Informações dos equipamentos monitorados..... 110

Lista de Abreviações

ACK	*Acknowledgment*
AD	*Administrative Domain*
ADS	*Anomaly Detection System*
AIR	*Active Integrated fault Reasoning*
AM	*Agent Manager*
API	*Application Programming Interface*
APV	*Anomaly Propagation View*
AR	*Auto-Regressivo*
AS	*Autonomous System*
BLGBA	*Baseline para Gerenciamento de Backbone Automático*
CBR	*Case-Based Reasoning*
CSI	*Computer Security Institute*
DLA	*Device Level Alarm*
DNS	*Domain Name System*
DDoS	*Distributed Denial of Service*
DoS	*Denial of Service*
DSNS	*Assinatura Digital do Segmento de Rede*
DWDM	*Dense Wavelenght Division Multiplexing*
EA	*Evidence Accumulation*
EJB	*Enterprise Java Beans*
FEC	*Fuzzy Event Correlation*
FTP	*File Transfer Protocol*

GBA	*Gerenciamento de Backbone Automático*
HIS	*Human Immune System*
HTTP	*HyperText Transfer Protocol*
ICMP	*Internet Control Message Protocol*
IDMEF	*Intrusion Detection Message Exchange Format*
IDS	*Intrusion Detection System*
IETF	*Internet Engineering Task Force*
IGMP	*Internet Group Management Protocol*
IHU	*Increment Hypothesis Updating*
IP	*Internet Protocol*
IPFIX	*IP Flow Information Export*
ISP	*Internet Service Provider*
JUNG	*Java Universal Network/Graph*
MIB	*Management Information Base*
NDG	*Network Dependency Graph*
NMS	*Network Management System*
OD	*Origem-Destino*
OSPF	*Open Shortest Path First*
PCA	*Principal Component Analysis*
PNG	*Portable Network Graphic*
QoE	*Quality of Experience*
QoS	*Quality of Service*
RBR	*Rule-Based Reasoning*
RFC	*Request for Comments*
SDH	*Synchronous Digital Hierarchy*

SLA	*Service Level Agreement*
SSC	*Sub-Space Clustering*
SSH	*Secure Shell*
SOM	*Self Organizing Map*
SYN	*Synchronize Sequence Number*
SNMP	*Simple Network Management Protocol*
TCP	*Transmission Control Protocol*
TES	*Traffic Entropy Spectrum*
TRA	*Taxa de Redução de Alarmes*
TM	*TM - Traffic Matrix*
UDP	*User Datagram Protocol*

Lista de Símbolos

D	Matriz de dependência
$AD_{inicial}$	Número de alarmes por dispositivo antes da operação de compressão
$AD_{comprimidos}$	Número de alarmes por dispositivo após a operação de compressão
$AT_{inicial}$	Número total de alarmes que chegou à camada de pré-processamento
$AT_{comprimidos}$	Número total de alarmes após a operação de compressão
TRA	Taxa de redução de alarmes após operação de compressão
$\mathcal{G}$	Grafo, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
$\mathcal{V}$	Conjunto de vértices do grafo
$\mathcal{E}$	Conjunto de arestas do grafo
v_i	Vértice do grafo
v_j	Vértice do grafo
$N(v_i)$	Vizinhos do vértice v_i
Gr_i	Maior valor da amostra utilizado para a geração do DSNS
Sm_i	Menor valor da amostra utilizado para a geração do DSNS
h	Diferença entre o maior (Gr_i) e o menor (Sm_i)
bl_i	Valor do DSNS no instante i
δ	Número de violações permitidas dentro do intervalo de histerese
W	Conjunto de dispositivos monitorados na rede
O	Conjunto de todos os objetos SNMP monitorados em cada equipamento
O_{ki}	Conjunto dos objetos SNMP monitorados no equipamento afetado pela anomalia
X	Conjunto de dispositivos afetados pela anomalia
$\#X$	Número de total de dispositivos afetados pela anomalia
λ	Conjunto de enlaces afetados pela anomalia

z_x^{in}	Grau de entrada de um vértice x
z_x^{out}	Grau de saída de um vértice x
$p_{r \to w}$	Probabilidade de propagação da anomalia pelo enlace que conecta o dispositivo r e w
$\boldsymbol{H}(X)$	Entropia clássica
q	Parâmetro entrópico da entropia generalizada
$\boldsymbol{H}_q(X)$	Entropia generalizada
$\boldsymbol{H}(X)_q^{Max}$	Valor máximo da entropia generalizada
S	Conjunto dos dispositivos candidatos a ser fonte da anomalia
$\#S$	Número de total de dispositivos candidatos a serem fontes da anomalia
F	Conjunto dos dispositivos candidatos a ser destino da anomalia
$\#F$	Número de total de dispositivos candidatos a serem destino da anomalia
P	Conjunto dos possíveis caminhos de propagação da anomalia
γ	Vetor que contém as probabilidades dos enlaces afetados pela anomalia
P_s	Caminho afetado pela anomalia
s_q	Nó inicial do caminho
f_t	Nó final do caminho
x_i	Dispositivo afetado pela anomalia
x_{i+u}	Dispositivo sucessor no caminho de propagação
MPP	Caminho mais provável de propagação

Trabalhos afins publicados pelo autor

1. AMARAL, Alexandre Aguiar; ZARPELÃO, Bruno Bogaz; MENDES, Leonardo de Souza; PROENÇA JR., Mario Lemes; RODRIGUES, Joel José Puga Coelho. **Analysing Network-Wide Anomalies Using Dependency Graphs and Baseline**. Proc IEEE International Conf. on Software, Telecommunications and Computer Networks (SoftCOM 2010), 2010.

Capítulo 1

Introdução

O importante papel das redes de telecomunicações nos diversos segmentos da sociedade faz do gerenciamento de redes uma tarefa vital, tanto para os provedores de serviços de telecomunicações quanto para seus usuários. O gerenciamento de redes tem se tornado uma tarefa cada vez mais desafiadora, pois atreladas à evolução e crescimento das redes de telecomunicações, tanto em tamanho quanto em complexidade, estão presentes anomalias que são responsáveis pelo comprometimento da qualidade e disponibilidade dos serviços prestados. Os efeitos colaterais de uma anomalia podem trazer muitos prejuízos, pois podem afetar a QoS (*Quality of Service*) e a QoE (*Quality of Experience*) dos usuários, além de causarem a violação de SLAs (*Service Level Agreement*) [75].

Anomalias de rede são situações nas quais a operação de uma rede apresenta um desvio do comportamento normal causado por eventos como falhas de equipamentos e ações maliciosas/intrusivas como os ataques de negação de serviço (DoS - *Denial of Service*) e crimes cibernéticos [44]. No contexto de gerenciamento de redes, estratégias que buscam identificar esses eventos anormais são denominadas sistemas de detecção de anomalias (ADS - *Anomaly Detection System*) [9].

Sistemas de detecção de anomalias emitem alarmes, tradicionalmente em formato de texto, como forma de notificação ao administrador quando alguma atividade anormal é detectada na rede [9] [60] [65]. Alarmes são fundamentais para o gerenciamento de uma rede, pois são indicadores de que existe uma anormalidade e que medidas deverão ser tomadas [32].

A ocorrência de uma única anomalia pode alterar o comportamento de todos os elementos de rede no seu caminho de propagação [24]. Esse fenômeno pode produzir um

número expressivo de alarmes, sobrecarregando o administrador de rede [56]. Em uma rede de médio e grande porte, esse número pode chegar a milhares de alarmes diariamente [65]. É inviável para o administrador de rede realizar a inspeção manual desses alarmes em um curto espaço de tempo.

Além da alta taxa de alarmes gerada na ocorrência de uma anomalia, outros agravantes podem dificultar a análise por parte do administrador:

- As informações contidas nos alarmes são, muitas vezes, incompreensíveis ou pouco descritivas [67], por não haver um tratamento prévio para filtrá-los ou traduzi-los para uma linguagem mais acessível [66];
- Alarmes podem ser gerados por diferentes fontes, cada uma com seu formato e atributos específicos [65].

O processo de investigação de cada mensagem contida no alarme, o entendimento da correlação entre elas e a decisão de quais medidas serão aplicadas são fatores que estão diretamente ligados à eficiência e à precisão no diagnóstico do problema. O grande volume de alarmes, com diferentes formatos, oriundos de diversas fontes pode levar o administrador a efetuar interpretações equivocadas, as quais implicam no comprometimento da restauração da operação normal da rede [3]. Esses fatores demonstram a necessidade de soluções automatizadas que minimizem a intervenção humana nesse processo complexo. Reduzir a intervenção humana pode resultar na identificação do problema de forma mais ágil e eficiente.

Nesse contexto, surge a correlação de alarmes. A correlação de alarmes é um processo que combina os diversos alarmes gerados inicialmente, denominados alarmes primitivos, dando a eles um novo significado [67]. Busca-se, através da correlação, identificar e/ou produzir alarmes mais significativos [3], oferecendo informações mais concisas para o propósito de localização e diagnóstico do incidente detectado na rede [24].

Em linhas gerais, as propostas para correlação de alarmes têm sido direcionadas para dois problemas principais: redução do volume de alarmes e identificação da origem do problema. Compressão e filtragem são algumas das técnicas utilizadas para redução do volume

de alarmes. A compressão consiste no agrupamento dos múltiplos alarmes emitidos por um mesmo evento transformando-os em um único alarme [45]. No processo de filtragem, é realizada a seleção de alarmes baseada em regras previamente especificadas. Essas regras são aplicadas com um propósito seletivo, no qual alarmes indesejáveis são descartados [70].

Dado o crescimento das redes, a estratégia de apenas reduzir o volume de alarmes não é suficiente. Segundo Wallin [67], um administrador humano pode ler e compreender de 15 a 30 alarmes por minuto. Uma abordagem proposta recentemente por Costa *et al.* [56], obteve uma taxa máxima de redução de alarmes de 70%. Esse método foi aplicado a um conjunto de 2,4 milhões de alarmes, gerados pela rede em um intervalo de quinze dias, com uma média de 160 mil alarmes diários. Considerando a taxa de redução de 70%, o número de alarmes emitidos diariamente seria reduzido de 160 mil para 48 mil. Supondo um cenário hipotético, em que o administrador de rede trabalhasse 24 horas sem interrupções e assumindo ainda o melhor caso em que ele investigaria 30 alarmes por minuto, um total de 43200 (1440 x 30) alarmes seriam investigados por dia, um número inferior a 48 mil.

A busca pela causa da anomalia ou causa raiz, responsável pela geração dos alarmes, tem sido outra vertente das pesquisas na área de correlação de alarmes [32]. Entretanto, grande parte delas tem sido direcionada apenas para anomalias geradas por falhas de *hardware* e *software* de redes [1] [7] [28] [78]. Um ambiente de rede real está suscetível a uma diversidade de outras anomalias, como os ataques DoS (*Denial of Service*) e DDoS (*Distributed Denial of Service*), *alpha flows* e *flash crowd.* Essas anomalias são denominadas anomalias de volume, responsáveis por variações bruscas no comportamento do tráfego da rede e que causam sérios danos à operação normal da rede [75].

Isso mostra que a identificação da fonte e destino da anomalia de volume e dos elementos de rede afetados pela sua propagação são informações imprescindíveis para a gerência. Porém, na prática, a aquisição dessas informações através da correlação de alarmes não é uma tarefa simples. A ocorrência de uma única anomalia pode ocasionar um fenômeno conhecido como tempestade de alarmes, podendo gerar muitos alarmes redundantes, aumentando a complexidade da localização da causa da anomalia. Esse problema é classificado como um problema NP-Completo [28] [46].

Somados aos alarmes redundantes, a ocorrência de alarmes falsos e alarmes perdidos, chamados espúrios, tornam o processo de identificação da fonte e do destino da anomalia ainda mais complexo. Os sistemas de detecção de anomalias estão propensos a emitirem alarmes falsos [60][62]. Além disso, em ambientes de redes reais, alarmes perdidos são inevitáveis [78]. Esses alarmes podem ser perdidos por uma interrupção de comunicação (por exemplo, indisponibilidade de um enlace) entre o elemento monitorado e o ponto de gerenciamento [47]. Esses alarmes espúrios podem aumentar a incerteza da análise, gerando falsas hipóteses sobre a origem e o destino da anomalia. Para Sun *et al.* [38], os alarmes espúrios são um dos principais contribuintes para o aumento da complexidade e redução da eficiência e precisão de um sistema de correlação de alarmes.

O sistema de correlação de alarmes é, portanto, considerado uma peça fundamental de um NMS (*Network Management System*) [9] [30] [71], cujo desenvolvimento deve considerar os seguintes fatores:

- ✓ **Mínima intervenção humana:** um sistema precisa reduzir o envolvimento do administrador humano na análise dos alarmes, especialmente nas camadas onde grandes quantidades de dados são processadas [56] [71]. Esse requisito é imprescindível tanto para a redução da carga de trabalho do administrador, quanto para redução do tempo de diagnóstico e solução do problema;
- ✓ **Habilidade de identificar um maior número de anomalias:** são necessárias soluções que estejam aptas a acompanhar as mutações e o surgimento de novas ameaças responsáveis pela degradação do desempenho e serviços da rede;
- ✓ **Capacidade de resolver as incertezas geradas por alarmes espúrios:** um dos requisitos importantes de um sistema de correlação é a habilidade de conduzir a inferência mesmo em condições na qual o conjunto de alarmes contenha dados incompletos/incertos. O fato de negligenciar a existência de alarmes falsos e perdidos implica na inviabilização do uso do sistema na prática. Portanto, o grande desafio é o desenvolvimento de estratégias capazes de mitigar a incerteza

gerada por alarmes espúrios, visando o não comprometimento da precisão e o resultado da análise [34];

- ✓ **Meios que facilitem o entendimento do problema ocorrido na rede:** a forma na qual é apresentado o resultado do processo de correlação deve permitir que o administrador interprete rapidamente o comportamento e o impacto da anomalia na rede. São necessárias estratégias que facilitem a interpretação do resultado, diferentes da forma tradicional, a qual utiliza a forma textual.

Este trabalho tem o objetivo de apresentar um sistema automatizado para o processo de correlação de alarmes. O objetivo é automatizar o processo de correlação, minimizando a intervenção humana e prover uma visão holística do impacto da anomalia na rede. Como infraestrutura para caracterização de tráfego e geração de alarmes, é utilizado o modelo BLGBA (*Baseline* para Gerenciamento de *Backbone* Automático), o qual é aplicado sobre dados históricos coletados em objetos SNMP (*Simple Network Management System*) para construir um *baseline* denominado DSNS (*Digital Signature of Network Segment* - Assinatura Digital do Segmento de Rede). Os dados coletados dos objetos SNMP são utilizados pelo sistema de alarmes proposto por [12] e [43], o qual detecta mudanças não esperadas no comportamento dos dispositivos de redes.

O sistema de correlação é divido em três camadas: (1) camada de pré-processamento, (2) camada de correlação e (3) camada de apresentação, descritas a seguir:

- ✓ **Pré-processamento:** responsável pela compressão dos alarmes primitivos. Para tal, é explorado o atributo espacial e temporal pertinentes a cada alarme. Baseando-se nesses atributos, alarmes disparados pelo mesmo dispositivo na ocorrência da anomalia são comprimidos, reduzindo-se a um único alarme, denominado neste trabalho de DLA. O resultado desse processo reduz o número de alarmes para cada dispositivo afetado pelo evento anômalo, minimizando a quantidade de alarmes no contexto global para a análise posterior;

- ✓ **Camada de correlação:** responsável por correlacionar os alarmes previamente gerados pela camada de pré-processamento. Busca-se, reunindo os DLAs e informações sobre a topologia da rede, produzir a melhor explicação sobre o incidente ocorrido na rede. Essa camada implementa um método para inferir o caminho de propagação da anomalia, sua origem e seu destino. Uma medida baseada na entropia generalizada de Tsallis [16] é proposta visando a utilização do método em cenários com alarmes espúrios;
- ✓ **Camada de apresentação:** responsável por apresentar uma visão global da rede. O impacto causado pela anomalia é apresentado de forma gráfica, possibilitando a visualização da propagação da anomalia e os elementos de rede afetados, facilitando a análise por parte dos administradores de rede.

As principais contribuições desta dissertação são:

- ✓ Proposta de um sistema de correlação de alarmes para inferir a fonte e destino de anomalias de volume.
- ✓ Uma medida baseada na entropia generalizada de Tsallis que, a partir da utilização de probabilidades *a priori*, é aplicada para inferir o caminho percorrido pela anomalia em cenários onde existem possíveis caminhos gerados por alarmes espúrios;
- ✓ Desenvolvimento da ferramenta multiplataforma APV (*Anomaly Propagation View*), que fornece ao administrador de rede a visualização do caminho e elementos de redes afetados pela propagação da anomalia;
- ✓ Resultados da aplicação da solução sobre dados reais coletados na rede da Universidade Estadual de Londrina (UEL).

1.1 Estrutura da dissertação

O restante desta dissertação é estruturado em seis capítulos. O capítulo 2 contém os conceitos referentes à anomalia, tipos de anomalias, suas possíveis causas e abordagens recentes propostas para detectá-las.

No capítulo 3, serão apresentadas as definições de alarmes, correlação de alarmes e os seus tipos. Métodos propostos para correlação de alarmes também serão apresentados, bem como suas vantagens e desvantagens.

No capítulo 4, trabalhos relacionados serão apresentados. Iniciamos com trabalhos voltados para correlação de alarmes, soluções de tomografia de redes e, por fim, trabalhos que utilizam entropia.

No capítulo 5, será apresentado o sistema de correlação de alarmes. Inicialmente, é dada uma visão geral de sua arquitetura e, posteriormente, a descrição dos seus componentes é apresentada de uma forma detalhada.

A implementação e resultados serão apresentados e discutidos no capitulo 6. Serão apresentados estudos de casos detalhados mostrando a aplicação da proposta com dados reais obtidos da rede da Universidade Estadual de Londrina.

Finalmente, as considerações finais deste trabalho serão apresentadas no capitulo 7.

Capítulo 2

Anomalias

2.1 Definições e classificação das anomalias

O tráfego de uma rede pode ser classificado segundo o seu comportamento como normal ou anômalo. O primeiro refere-se a um comportamento estável e previsível. O segundo refere-se a um desvio abrupto e imprevisível causado por eventos inesperados, tais como a falhas, erros de configurações nos dispositivos de rede e atividades relacionadas à segurança, como ataques [39] . Esses eventos, responsáveis pela alteração do comportamento do tráfego e operação de uma rede, são denominados de anomalias [44].

A detecção de anomalias refere-se ao problema de identificar padrões de dados não condizentes a um comportamento previamente esperado [39]. Uma anomalia pode ser observada através do aumento no número de pacotes, número de *bytes*, uma concentração de pacotes em torno de uma determinada porta, etc. [63]. Esses dados podem ser obtidos de duas formas: monitoramento ativo/intrusivo e monitoramento passivo/não intrusivo.

2.1.1 Monitoramento ativo

No monitoramento ativo, pacotes são injetados com o propósito de verificar o estado dos elementos de rede. Esses pacotes são denominados de *probes* (sondas) enviados por monitores conhecidos na literatura como *probes station* ou *beacon* [35] [59]. Um *beacon* é um dispositivo de rede que pode atuar no monitoramento ativo, enviando sondas e, no monitoramento passivo, coletando informações dos elementos monitorados. Esse tipo de monitoramento realiza

medições fim-a-fim, sendo possível obter informações e características do tráfego, tais como perda e atraso de pacotes. No que diz respeito às ferramentas de medições, *ping* e *traceroute* são provavelmente as mais populares. Para propósitos mais específicos, requisições HTTP também são utilizadas, a fim de obter informações relativas à disponibilidade e tempo de resposta de servidores [42].

A vantagem do monitoramento ativo está na quantidade de informações que pode ser coletada sobre o comportamento do tráfego e dos elementos de redes. Essas medidas não se restringem apenas ao AS (*Autonomous System*), onde está a estação de gerência, mas possibilita obter informações através de medições entre domínios. Outro benefício é o fato de que não é preciso instalar agentes em cada dispositivo de rede, como é o caso dos agentes SNMP [76]. Por outro lado, a injeção de pacotes realizada pelo monitoramento ativo pode modificar as propriedades que se deseja medir, tais como *jitter* e latência em caminhos fim-a-fim [22]. Outra desvantagem é que muitos ISPs (*Internet Service Provider*) restringem o uso do monitoramento ativo por razões de privacidade e segurança.

2.1.2 Monitoramento passivo

No monitoramento passivo, agentes são incorporados a elementos de rede como roteadores e switches para inspecionar o tráfego que flui através desses equipamentos [76]. Essa inspeção pode ser simples, visando apenas a contabilização de pacotes e/ou *bytes* no tráfego ou mais sofisticadas, como a extração de dados dos cabeçalhos de pacote transmitidos, como endereços IP de origem e destino, portas de origem e de destino, dentre outras.

O monitoramento passivo do tráfego pode ser realizado de diversas formas. Uma das mais comuns é através do protocolo SNMP. O SNMP é largamente empregado e suportado pela grande maioria dos equipamentos de rede [44]. A infraestrutura exigida para a sua utilização é mínima, representando baixos custos adicionais, o que o torna uma opção atrativa e uma fonte ideal de dados para a detecção de anomalias.

NetFlow [57] e IPFIX (*IP Flow Information Export*) [11] têm sido alternativas para a coleta de informações no monitoramento passivo. Embora o NetFlow seja uma opção, esse é um

protocolo proprietário desenvolvido pela Cisco e, consequentemente, não é suportado por todos dispositivos de rede. Um grupo de trabalho do IETF (*Internet Engineering Task Force*), iniciado em 2001, definiu um protocolo aberto para a exportação de fluxo denominado IPFIX, o qual é baseado na nona versão do NetFlow.

A vantagem do monitoramento passivo é capacidade de monitorar um conjunto de enlaces dentro do domínio administrativo utilizando um único ponto de monitoramento [64]. Porém, se faz necessária a instalação de agentes em todos os dispositivos que se deseja monitorar [76].

2.1.3 Anomalia de volume

Anomalias de volume são aquelas caracterizadas por variações bruscas no tráfego [54]. Essas anomalias se propagam pela rede e seu efeito colateral pode resultar em congestionamento, atrasos, sérios danos na operação e/ou disponibilidade de serviços e violação de SLA (*Service Level Agreement* - Acordo de Nível de Serviço) em um curto espaço de tempo [63]. As variações de tráfego que caracterizam as anomalias de volume são causadas por uma diversidade de eventos inesperados, tais como os ataques DoS e DDoS, *flash crowd* e *alpha flows*.

De uma forma geral, as anomalias de volumes podem ser agrupadas em duas diferentes categorias [5] [63]: comportamento não usual e comportamento malicioso. A seguir será detalhada cada categoria, apresentando anomalias pertinentes a cada uma delas.

2.1.3.1 Comportamento não-usual

O primeiro tipo de anomalia que vamos abordar é o *flash crowd*. Essas anomalias são caracterizadas pelo grande volume de tráfego, gerado por inúmeras fontes e direcionado para um único destino na rede [31]. Comumente, esses eventos estão relacionados a requisições feitas a servidores *Web*. Um determinado evento inesperado, como o ataque terrorista de 11 de setembro de 2001 nos Estados Unidos, pode resultar no acesso de milhões de pessoas em todo mundo a um portal de notícias levando à indisponibilidade do serviço.

Como destacado por Lakhina *et al*. [5] e Jung *et al.* [31], a ocorrência de *flash crowd* na rede pode resultar em:

- ✓ Requisições de alguns usuários podem ser descartadas, dada a limitação dos servidores;
- ✓ Caso a requisição seja aceita, a entrega da informação pode ocorrer com atrasos significativos causados pela perda de pacotes e tentativas de retransmissão;
- ✓ Pela grande taxa de requisições, podem ocorrer congestionamentos ou a parada total de uma rede, impactando no acesso de outros recursos disponíveis, tais como email, FTP (*File Transfer Procotol*) e SSH (*Secure Shell*).

Um tráfego anômalo também pode ser resultado de uma alta taxa inesperada de dados transmitidos entre pontos na rede. Esse evento é denominado de *alpha flows* [5]. A diferença entre um *alpha flows* e *flash crowd* está no fato de que o primeiro consiste na geração de um grande volume de dados transferidos entre um único nó fonte e um único nó de destino, ao passo que no segundo existe um grande número de fontes enviando requisições para um único destino. Tanto o *flash crowd* quanto o *alpha flows* são eventos não usuais observados na rede e diferem dos ataques por não terem como propósito o comprometimento ou roubo de informação.

2.1.3.2 Comportamento malicioso

Mesmo com diversas soluções e medidas utilizadas para evitar eventos anômalos causados por ataques, eles continuam a causar sérios danos às corporações. Uma rede está suscetível a um grande conjunto de ataques. Porém, nesta seção iremos nos ater aos ataques causadores de anomalias de volume, em particular, os ataques de negação de serviços.

Um ataque de negação de serviço é definido como qualquer ato de origem maliciosa e intencional com o objetivo de tornar indisponível ou inutilizável um recurso ou serviço oferecido pela rede.

Este tipo de ataque ocorre há muito tempo e ainda continua responsável por causar danos e perdas financeiras, gerados pela indisponibilidade de um recurso na rede. A Figura 2.1 mostra o número de ocorrências desses ataques nos últimos anos. Esse estudo, realizado pela CSI (*Computer Security Institute*) em 2009 [15], envolvendo as agências governamentais, instituições financeiras, educacionais, dentre outras organizações, mostrou que houve uma redução do número de ocorrência de atividades voltadas para negação de serviços entre 2006 e 2008. Contudo, as ocorrências voltaram a crescer em 2009.

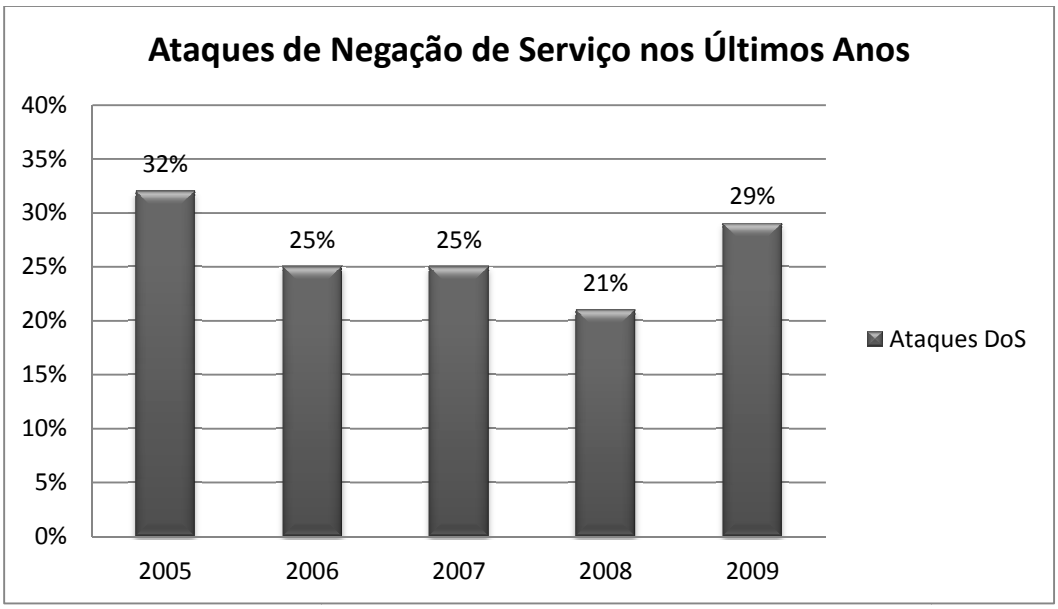

Figura 2.1 – Estatística de ocorrências de ataques de negação de serviço. Fonte [15].

Como destacado em [51], há uma tendência de que esse número aumente de forma mais acentuada nos próximos anos. Em 2010, ataques de negação de serviço foram disparados contra as empresas Visa e Mastercard, como forma de protesto pelo bloqueio que ambas as empresas realizaram nas doações feitas por anônimos ao site Wikileaks [51]. Diferente da forma usual, em que máquinas de usuários são invadidas e utilizadas de forma ilegal para realização do ataque, anônimos de diversos lugares do mundo aderiram ao ataque de maneira voluntária. Isso indica que a forma tradicional de protestos está, gradativamente, adquirindo um novo caráter através do meio virtual, denominado de *cyber*-protesto. Os *cyber*-protestos serão muito mais comuns e,

possivelmente, os principais responsáveis pelo aumento de ataques de negação de serviço nos próximos anos [51].

Os ataques de negação de serviço são classificados em dois tipos, segundo o número de fontes geradoras do tráfego anômalo. O primeiro corresponde ao DoS (*Denial of Service*), no qual uma única fonte é responsável por gerar um grande número de requisições a um recurso de rede. O segundo, refere-se a um ataque distribuído, conhecido como DDoS (*Distributed Denial of Service*), onde é utilizado um grande número de máquinas, conhecidas como máquinas *zumbis*. As máquinas *zumbis* são máquinas infectadas por *softwares* maliciosos que permitem que os criminosos tenham controle sobre elas para a realização do ataque. Para Rahmani *et al.* [27], dentre as anomalias, o DDoS é considerada uma das ameaças mais graves para as redes.

A forma tradicional de um ataque de negação de serviço é realizada em duas etapas:

I. Obtenção do controle de uma máquina vulnerável na rede. Essas informações podem ser obtidas através de ferramentas especializadas para varredura em busca de brechas de segurança causadas por alguma vulnerabilidade. Um *software* malicioso, tradicionalmente um *trojan*, é instalado na máquina, possibilitando ao atacante ter o controle remotamente de seus recursos;
II. O ataque é orquestrado pelo criminoso, como ilustrado na Figura 2.2, no qual é requisitado às maquinas infectadas (*zumbis*) que façam inúmeras requisições, que culminará na inundação (*flooding*) e esgotamento do recurso prestado pelo alvo.

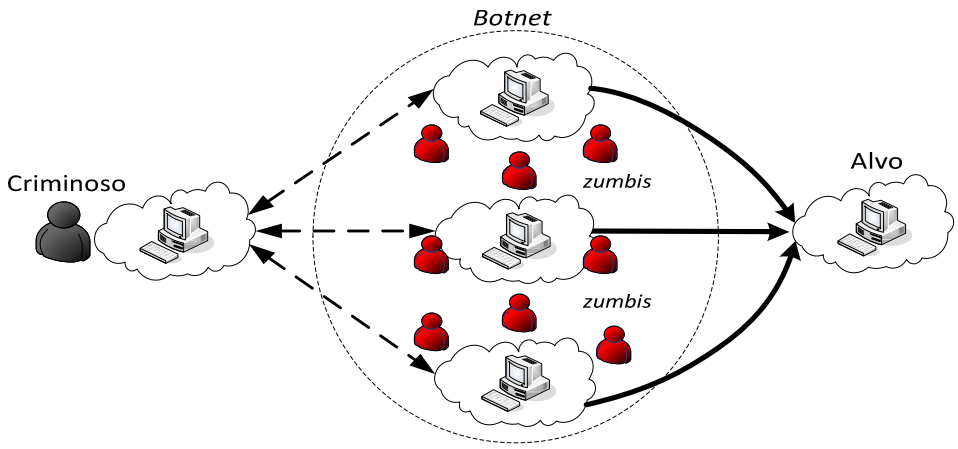

Figura 2.2 - Ataque de negação de serviço.

Os ataques mais conhecidos de negação de serviços são: UDP *flood*, SYN *flood* e ICMP *flood*. Um ataque UDP *flood* é considerado um ataque simples de preparar. Isso se deve à forma de funcionamento do protocolo UDP (*User Datagram Protocol*), que não é orientada a conexão. Comumente, o criminoso determina que as máquinas *zumbis* enviem pacotes UDP para portas aleatórias do alvo. Ao receber um pacote UDP, o alvo busca, pelo recurso que está utilizando, a porta solicitada. Ao identificar que nenhuma aplicação está aguardando pacotes na porta, é enviado um pacote ICMP informando ao requisitante que o destino não pode ser alcançado. Quando esse processo é efetuado em larga escala, os recursos do alvo são saturados podendo culminar na interrupção dos serviços prestado.

O ataque SYN *flood* explora o modelo de estabelecimento de conexões do protocolo TCP (*Transport Control Protocol*), o *handshake* de três vias (3-*way handshake*). Como o nome sugere, o ataque utiliza as mensagens de sincronização SYN (*synchronization*). A Figura 2.3 mostra as mensagens trocadas entre ambas as partes para iniciar a troca de informação. Um nó cliente envia um pacote SYN (1) ao nó servidor que responde com SYN/ACK (2) ao cliente e espera que o cliente confirme o estabelecimento da conexão pelo envido do ACK (3). O ponto chave do ataque consiste em não efetuar o passo (3) por parte do cliente. Ao enviar a mensagem (2), recursos são reservados pelo servidor para o cliente e fica aguardando a sua resposta. Quando inúmeras conexões incompletas são realizadas simultaneamente por inúmeras máquinas *zumbis*, o servidor começa a ser sobrecarregado, dada a quantidade finita de recursos.

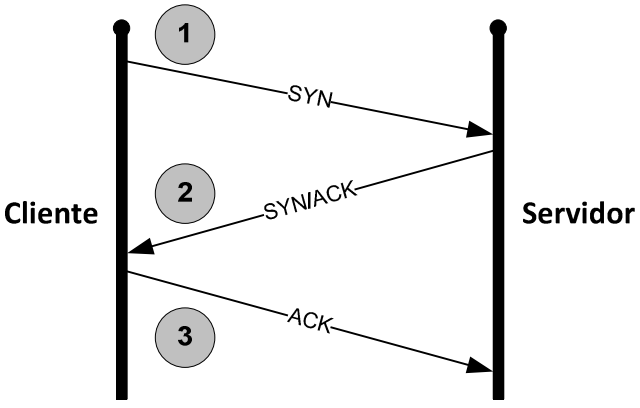

Figura 2.3 - Estabelecimento de uma conexão TCP.

O ataque ICMP *flood* tem o mesmo objetivo dos anteriores, porém o protocolo utilizado é o ICMP (*Internet Control Management Protocol*). Nesse ataque, é enviado ao alvo grande quantidade de pacotes com tamanho da carga útil (*payload*) consideravelmente grande. Dessa forma, quando esse tráfego ICMP é gerado, o caminho entre o atacante e o alvo pode ficar congestionado implicando, diretamente, na perda de pacotes. O servidor, ao receber uma mensagem ICMP do tipo *Echo Request*, responde com a mensagem ICMP *Echo Reply*, ficando sobrecarregado quando inúmeras requisições são realizadas simultaneamente pelas máquinas *zumbis*.

2.2 Métodos de detecção de anomalias

A detecção de anomalias tem sido extensivamente estudada na última década e muitas soluções tem sido propostas. O objetivo desta seção é apresentar trabalhos recentes voltados para detecção de anomalias de volume.

O grande volume de dados gerados pelo crescimento das redes tem sido considerado um dos principais fatores que dificultam a detecção da anomalia. Isso está diretamente ligado a questões de processamento e armazenamento. Por esses fatores, faz-se necessária a redução do volume de dados. Para esse propósito, tem se aplicado a técnica PCA (*Principal Component Analysis*). A aplicação dessa técnica teve origem no processamento de imagens e tem sido amplamente utilizada no contexto de detecção de anomalias [9].

Callegari *et al.* [14], apresentaram uma solução constituída de três módulos principais a fim de diagnosticar anomalias de volume. O primeiro módulo é responsável por obter os dados via NetFlow, formatá-los e gravá-los em arquivos de textos. O segundo módulo é responsável pela construção de séries temporais. O terceiro módulo aplica a técnica PCA, buscando a redução da dimensionalidade dos dados coletados a fim de definir o comportamento normal do tráfego. A detecção da anomalia é realizada através da comparação dos dados produzidos pelo terceiro módulo com um limiar previamente definido. Quando o tráfego excede o limiar, alarmes são disparados.

Um das formas mais usuais utilizadas para detecção de anomalias consiste na observação do volume de dados do tráfego que excede um limiar considerado normal. Outra estratégia baseia-se na distribuição de características do tráfego de rede. O ponto central dessa abordagem é a utilização de uma medida capaz de extrair propriedades do tráfego, fornecendo dados para detecção e classificação dos eventos anômalos na rede. A entropia tem sido considerada uma medida eficaz para esse propósito [5].

Ziviani *et al.* [10] propuseram um método de detecção de anomalias utilizando a entropia generalizada de Tsallis. Eles mostraram que a utilização da entropia generalizada apresentou melhores resultados do que a entropia clássica de Shannon. A razão disso é que a entropia generalizada possui um parâmetro denominado de parâmetro entrópico que possibilita ao administrador de rede ajustar a sensibilidade de detecção para os diferentes tipos de anomalias.

Bernhard *et al.* [68] apresentaram o método TES (*Traffic Entropy Spectrum*), que utiliza a entropia generalizada para identificar as mudanças na distribuição das características do fluxo de rede, a fim de detectar as anomalias de volume. As características do fluxo utilizadas por Bernhard *et al.* foram: endereço IP de origem e destino e a porta de origem e destino. Foram avaliados os fluxos TCP (*Transmission Control Protocol*), UDP (*User Datagram Protocol*) e ICMP (*Internet Control Message Protocol*). Além da detecção, o método proposto possibilita a captura e visualização gráfica das características do tráfego.

Utilizando conceitos de teoria da informação e entropia, Rahmani *et al.* [27] propuseram um modelo de detecção de anomalias de volume, em particular aquelas causadas por ataques DDoS. Essa solução parte do princípio de coerência entre a quantidade de pacotes e o número de fluxos IP recebidos por uma rede. A fim de delinear o fator coerência, os autores exemplificam a diferença entre uma anomalia gerada por um ataque DDoS e por um *flash crowd.* Em um ataque DDoS, há uma diferença brusca entre o número de pacotes e o número de fluxo recebidos pela rede, pois cada ponto utilizado no ataque pode gerar um grande volume de pacotes em um único fluxo destinado ao ponto alvo. No caso do *flash crowd*, a quantidade de pacotes recebidos pela rede aumenta em decorrência do número de fluxos. Dessa forma, a proposta busca identificar a distribuição de probabilidade conjunta entre o número de fluxos IP

e o volume de pacotes e, posteriormente, aplicar a entropia conjunta a fim de quantificar o grau de coerência entre o número de pacotes e o número de fluxos.

Em uma rede, os fluxos de tráfego são comumente representados por uma matriz, denominada de matriz de tráfego (TM - *Traffic Matrix*). Ela armazena dados referentes ao tráfego transmitido entre cada par de pontos de entrada e saída da rede, conhecido como pares Origem-Destino (OD). Um método baseado em cálculos estatísticos, proposto por Fillatre *et al.* [39], busca detectar as anomalias de volume, utilizando as informações dos enlaces contidas na matriz TM e obtidas através de medições do protocolo SNMP e informações de roteamento.

A proposta apresentada por Du *et al*. [54], centra-se na detecção de anomalias utilizando séries temporais e na identificação do caminho de propagação da anomalias. A fim de detectar a anomalia, o tráfego de rede é decomposto em três componentes: componente de tendência, componente auto-regressivo (AR) e componente de ruído. O componente de tendência tem por objetivo identificar as mudanças do volume de tráfego na série temporal. O componente AR identifica as mudanças do volume de tráfego na série temporal juntamente com o componente de ruído, onde é assumido o ruído branco. Este trabalho propõe o algoritmo *shortest-path-first* baseado no algoritmo de Dijkstra, que utiliza os custos obtidos através do protocolo OSPF (*Open Shortest Path First*) para traçar o caminho de propagação da anomalia.

Enquanto muitas propostas são voltadas para detecção de anomalias no tráfego de entrada, isto é, no tráfego que vem da Internet para o AD (*Administrative Domain*), a abordagem proposta por Limthong *et al.* [37] visa a detecção de anomalias no tráfego de saída. A justificativa é que a detecção dos eventos anômalos, gerados internamente, possibilita ao administrador identificar as causas da anomalia em sua rede e aplicar medidas para que elas não afetem outras redes. A solução proposta é baseada na técnica de processamento de sinais, na qual é aplicada *wavelets* para detecção de anomalias de volume, como *flash crowd* e ataques DoS.

Um *framework* biologicamente inspirado é apresentado por Hashim *et al.* [21] para detecção de eventos anômalos resultantes de ataques DoS, DDoS e *worms*. A proposta se baseia em dois campos de estudo da biologia: imunologia e epidemiologia. O *framework* é constituído de dois componentes principais. O primeiro utiliza os princípios de funcionamento do sistema imunológico humano (HIS - *Human Immune System*), em particular a teoria do perigo, para

detectar ações maliciosas na rede. O segundo tem como propósito conter a propagação da atividade anômala pela rede através de estratégias utilizadas na epidemiologia.

Um método de clusterização não supervisionado é proposto por Casas *et al.*[53]. A solução é divida em três estágios. No primeiro estágio, os fluxos de redes são capturados em intervalos de tempo com tamanho fixo e são agrupados utilizando informações dos pacotes como endereços de origem e destino. O fluxo do primeiro estágio são entradas para o segundo, que consiste na utilização dos algoritmos SSC (*Sub-Space Clustering*), EA (*Evidence Accumulation*) e clusterização baseada em densidade para detecção e caracterização do tráfego anômalo. O último estágio tem como objetivo construir as assinaturas das anomalias detectadas.

Capítulo 3

Correlação de alarmes

3.1 Alarmes

No contexto das redes de telecomunicações, um alarme é definido como um tipo especial de notificação projetado para informar a existência de mudança no estado de um elemento monitorado [30] [34] [67].

Diferentes nomenclaturas são alternativamente utilizadas para substituir o termo alarme. São encontradas na literatura o termo evento, sintomas, hipóteses e evidências [20] [78]. Desses, o termo evento é mais comum, apesar de alguns autores, tais como Bellec *et al.* [30] e Steinder *et al.* [46], fazerem distinções entre um evento e um alarme.

Alarmes podem ser gerados por sistemas de detecção de anomalias (ADS), sistemas de detecção de intrusão (IDS), ou mesmo por equipamentos de redes [65]. Esses alarmes são denominados na literatura como alarmes primitivos, gerados fora do processo de correlação. Alarmes que recebem tratamento dentro do processo de correlação são comumente denominados de meta-alarmes [55]. O fato é que independentemente do alarme, ele requer uma investigação incluindo possíveis ações [67].

Um alarme é comumente uma mensagem curta em formato de texto [30], importante para o gerenciamento de uma rede, pois é um indicador de que existe uma anormalidade e que, possivelmente, medidas deverão ser tomadas [32].

Exemplos de alarmes são apresentados na Figura 3.1, os quais foram obtidos do sistema de alarmes proposto por Zarpelão *et al.* [12]. A figura mostra alarmes gerados para o dispositivo

firewall, com o IP 189.xx.xx.2 na porta 3001. Cada linha contém um alarme com o *timestamp* e o nome do objeto SNMP para o qual foi identificada a anomalia.

```
Device: Firewall - IP Address: 189.xx.xx.2- Port:3001

      03May2011;05:12:52;SNMP_OBJECT:ifInOctets;
      03May2011;05:13:11;SNMP_OBJECT:ifInOctets;
      03May2011;05:35:47;SNMP_OBJECT:ifInOctets;
      03May2011;06:03:28;SNMP_OBJECT:ifInOctets;
      03May2011;06:04:06;SNMP_OBJECT:ifInOctets;
```

Figura 3.1 – Fragmento contendo exemplos de alarmes.

3.1.1 Classificação dos alarmes

Em um contexto mais específico, onde alarmes são gerados por falhas de redes, eles podem ser classificados em duas categorias: *físicos* e *lógicos* [17]. Alarmes físicos referem-se a falhas de *hardwares*, como a queda de um enlace. Alarmes lógicos são referentes a erros estatísticos resultantes de falhas que culminam na degradação da operação normal da rede, como os congestionamentos.

Alarmes podem ser falsos ou ainda perdidos. Alarmes falsos são comumente divididos em *falsos positivos* e *falsos negativos*. Falso positivo é um alarme que indica uma anormalidade mesmo que essa não exista. No contexto de detecção de anomalias, um falso alarme refere-se à notificações que erroneamente indicam a existência de uma anomalia no tráfego de rede, mesmo esse sendo um tráfego normal. Falsos negativos ocorrem em situações nas quais há uma anomalia ocorrendo na rede, mas o sistema de detecção classifica o comportamento como normal. Alarmes falsos são prejudiciais ao processo de correlação, pois fornecem informações enganosas sobre os elementos afetados pelo evento anômalo [78].

Alarmes perdidos são aqueles gerados por um elemento em um determinado ponto da rede na ocorrência de uma anormalidade, que não chegam à central de armazenamento ou ponto de gerenciamento, devido a uma interrupção na comunicação (por exemplo, indisponibilidade de um enlace) [47]. Esses alarmes podem resultar na perda de informações importantes para o diagnóstico do problema [17].

Nesta dissertação, utilizaremos a terminologia alarmes espúrios, que engloba os alarmes falsos e perdidos.

3.1.2 Atributos de um alarme

Um alarme pode possuir diversos atributos descrevendo o evento que o disparou. Para Bouloutas *et al.* [1], um alarme perfeito seria aquele contendo os 5*Ws*:

- ✓ *Quem:* Identificação do elemento de rede afetado pela anomalia;
- ✓ *O que:* Provável causa da anomalia*;*
- ✓ *Onde*: A descrição da posição do dispositivo na rede afetado pela anomalia;
- ✓ *Quando:* O *timestamp* indicando o momento em que o alarme foi disparado;
- ✓ *Por que:* Descrição da causa do problema.

Se todos esses atributos fossem fornecidos pelo alarme, a correlação seria evidentemente facilitada. Entretanto, os alarmes atuais normalmente não fornecem esses atributos [3]. A qualidade dos alarmes tem sido considerada um dos obstáculos para a correlação de alarmes [66] [67]. Os atributos *onde* e *quando* são considerados informações mínimas que um alarme deve possuir. *Onde* tradicionalmente é dado pelo endereço IP (*Internet Protocol*) e porta do dispositivo e *quando* é dado pela data e hora em que o problema foi detectado. A correlação explorando esses dois atributos, a qual considera o alarme no espaço-tempo, é conhecida na literatura como correlação espaço-temporal.

3.1.3 Metadados

As informações contidas nos alarmes primitivos são pouco informativas e insuficientes, em muitos casos, para identificação do problema responsável pela sua geração [65]. Para preencher essa lacuna, dados adicionais são utilizados com objetivo de agregar informações ao processo de correlação. Essas informações são denominadas de metadados [71]. Exemplos

comuns de metadados importantes no contexto de correlação de alarmes são as informações da topologia e das dependências entre os elementos de redes.

Para alcançar o objetivo deste trabalho, utilizamos informações da topologia da rede e dos objetos SNMP. Através de um conjunto de objetos (descrito na seção 5.3.1), são extraídas informações para identificar o caminho de propagação do tráfego anômalo na rede. Alarmes disparados para cada objeto indicarão se o tráfego anômalo ingressou ao equipamento, partiu dele ou se ambas as situações ocorreram.

3.2 Correlação de Alarmes

A correlação de alarmes é um procedimento de interpretação conceitual em que um novo significado é atribuído a um conjunto de alarmes disparados durante um intervalo de tempo [3] [17]. Na literatura, o termo correlação de evento é usado como sinônimo [45] [71].

O objetivo principal da correlação é a redução do volume de alarmes [17] [19]. Para Sadoddin e Ghorbani [60], a correlação vai muito além da redução do número de alarmes. Segundo os autores, é um processo que visa a identificação da origem e destino dos eventos anômalos observados na rede, que são informações difíceis de obter analisando individualmente os alarmes primitivos. A razão disso é, que embora os alarmes primitivos contenham alguns atributos, eles geralmente não trazem informações explícitas sobre a sua natureza e a origem do problema, pois são gerados para cada dispositivo de forma isolada. Dessa forma, faz-se necessária a correlação, que busca por conexões lógicas entre os alarmes a fim de descrever o problema [3].

As primeiras pesquisas e soluções na área de correlação de alarmes tiveram início nos anos 80. Naquela época, os sistemas de correlação eram, exclusivamente, voltados para o gerenciamento e identificação da causa das falhas relacionadas à telefonia [34]. Com o passar dos anos, os sistemas de correlação começaram a ser usados para gerenciar dispositivos de rede. A necessidade de metodologias para a correlação surgiu devido ao crescimento das redes em tamanho e complexidade e pela limitação humana de processar grande volume de dados [28].

Desde então, muitas soluções têm sido propostas, mas ainda estão longe da solução ideal esperada pelos administradores de rede [29] [67].

Uma das razões é que o projeto de um sistema eficaz de correlação de alarmes é complexo, como é destacado por Martin-Flatin *et al.* [34] e Wallin *et al.* [65]:

- ✓ A qualidade das informações providas pelos alarmes primitivos é muito baixa, dificultando a correlação;
- ✓ Falta de padronização dos alarmes quando há várias fontes de dados;
- ✓ Existência de alarmes espúrios que podem gerar muitas incertezas, ambiguidades e falsas hipóteses sobre a causa e origem do problema;
- ✓ É gerado um grande volume de alarmes, requerendo grande capacidade de processamento.

Outro fator é que as soluções têm focado em dois pontos principais: redução do volume de alarmes e localização de falhas. A problemática reside no fato de que mesmo as técnicas de redução de alarmes com as mais altas taxas podem ser insuficientes para minimizar o grande volume de alarmes gerado. Um segundo ponto é que eventos anômalos podem ser causados por diferentes fatores, não exclusivamente por falhas. Para Thottan *et al.* [44], o número de anomalias geradas por ações maliciosas é maior do que as geradas por falhas, pois os *hardwares e softwares* de redes já estão bem consolidados e menos propícios a falhas.

O desafio das soluções de correlação deverá ser ainda maior. Martin-Flatin *et al.* [34] relatam que, no início da década de 90, a quantidade de alarmes era em torno de 5 a 10 alarmes por segundo. Dez anos depois, é citado o caso da AT&T, que processou em 2001 cerca de 10 a 100 alarmes por segundo, com casos que chegaram até 1.000 alarmes por segundo. Os autores destacam que o processamento necessário para correlacionar o volume de alarmes das futuras redes será um dos grandes desafios das soluções de correlação.

3.2.1 Tipos de operação

Nesta seção serão apresentados os principais tipos de operação aplicados à correlação de alarmes. Algumas das operações podem ser estendidas e aplicadas a diversos cenários, enquanto outras possuem aplicações mais restritas. Será evidenciada a falta de padronização das terminologias utilizadas para as operações. Além dos termos, há opiniões divergentes sobre se as operações são um tipo de correlação ou um componente da correlação de alarmes [67]. Para Zurutuza *et al.* [74], a pluralidade de terminologias utilizadas na área de correlação de alarmes é uma deficiência encontrada na literatura e se faz necessária uma padronização. Neste trabalho, buscaremos utilizar os termos mais comuns utilizados na literatura.

3.2.1.1 Compressão de alarmes

A compressão de alarmes consiste em substituir alarmes similares por um único alarme como pode ser visto em (3.1). Há divergências sobre o que é considerado similar. Alguns consideram que um alarme é similar quando todos os atributos (por exemplo, IP e Porta de destino) dos alarmes são similares entre si, exceto o atributo *timestamp*, ao passo que para outros, dois alarmes são considerados similares se são resultados de um mesmo evento [60]. Utilizaremos aqui e nas demais operações uma representação simbólica baseada em [3].

$$\text{Compressão: } \langle A_1, A_2, \dots, A_n \rangle \longmapsto A \tag{3.1}$$

O termo compactação é utilizado por [23] para a mesma operação. Já para Sadoddin e Ghorbani [60], essa operação é denominada agregação.

3.2.1.2 Supressão seletiva

Supressão seletiva ou simplesmente supressão é a operação que tem por finalidade inibir alarmes dentro de um determinado contexto de correlação. Diferente da compressão, em que os

alarmes são descartados, a supressão utiliza critérios para suprimir temporariamente os alarmes [3] [45]. Como apresentado em (3.2), pelo critério C os alarmes $A_1, A_2, \dots, A_n$ não serão exibidos.

$$\text{Supressão: } \langle A_1, A_2, \dots, A_n | \, \mathrm{C} \rangle \longmapsto \emptyset \tag{3.2}$$

Para Gürer [17], a supressão é a omissão de alarme de baixa prioridade, na presença de um alarme com uma prioridade maior.

3.2.1.3 Filtragem

A filtragem de alarmes consiste na utilização de parâmetros para selecionar, suprimir, agrupar, priorizar e descartar um alarme ou um conjunto de alarmes que não estejam de acordo com um padrão especificado. Em (3.3), temos que se o valor de um parâmetro $\mathrm{p}(A_k)$ de A_k não for atendido ele é descartado. Exemplos de parâmetros são: tempo, tipo, prioridade, etc.

$$\text{Filtragem: } \langle \mathrm{p}(A_1), \mathrm{p}(A_2), \dots, \mathrm{p}(A_n) \neq \mathrm{V} \rangle \longmapsto \emptyset \tag{3.3}$$

3.2.1.4 Generalização

A generalização consiste na substituição de um alarme por um correspondente na sua superclasse. O objetivo dessa operação é a de prover um alarme mais representativo dado um conjunto de alarmes. Como podemos observar em (3.4), a generalização consiste na substituição dos alarmes pertencentes a uma classe $\mathrm{c}(A_k)$ por um alarme na superclasse que, denominamos de A^+.

$$\text{Generalização: } \langle \mathrm{c}(A_1), \mathrm{c}(\, A_2), \dots, \mathrm{c}(A_n) \mid \mathrm{G} \rangle \longmapsto A^+ \tag{3.4}$$

Na prática, a generalização é útil para identificar a causa do evento anômalo responsável pelos alarmes emitidos. Por exemplo, suponhamos que os alarmes $A_1, A_2, \dots, A_n$ pertencentes a classe DoS estão sendo emitidos pelo servidor *Web*. Em vez de reportar todos os alarmes ao

administrador, apenas um alarme A^+ seria enviado indicando a existência de um ataque DoS na rede, isto é, o evento responsável pelo geração dos alarmes e não os alarmes propriamente ditos.

3.2.1.5 Especialização

A operação de especialização emprega o mesmo raciocínio utilizado pela generalização, mas de uma forma inversa. A especialização busca gerar alarmes mais especializados através de um alarme genérico [23], como pode ser visto em (3.5).

$$\text{Especialização: } \langle A^+ \mid \text{E} \rangle \longmapsto A_1, A_2, \dots, A_n \tag{3.5}$$

3.2.1.6 Contagem

Na operação de contagem, um novo alarme é gerado cada vez que o número de ocorrências de um determinado tipo de alarme atinge um limite pré-definido [3] [17]. Nesse caso, há perda de dados, pois alarmes que não atingirem o limiar são descartados pelo processo de correlação [23]. A operação de contagem é demonstrada em (3.6). Dado um conjunto de alarmes, busca-se verificar a ocorrência daqueles que são similares. Quando a quantidade da ocorrência de um determinado alarme A_k é maior do que um limiar L, ou seja, ($n \times A_k >$ L) um novo alarme A_k^* é criado.

$$\text{Contagem: } \langle n \times A_1, n \times A_2, \dots, n \times A_n > \text{L} \rangle \longmapsto A_1^*, A_2^*, \dots, A_m^* \tag{3.6}$$

Para Pouget e Dacier [23], a operação de contagem é denominada *thresholding*.

3.2.1.7 Escalação

A operação de escalação tem por objetivo incrementar o valor de determinado(s) parâmetro(s) de um alarme. O parâmetro de prioridade é normalmente o mais utilizado [3]. De uma forma geral, a escalação pode ser representada em (3.7). O procedimento M é aplicado sobre cada alarme A_k com parâmetro (p) onde um valor maior é adicionado a cada parâmetro resultando em (p^+).

$$\text{Escalação: } \langle A_1(\text{p}), A_2(\text{p}), \dots, A_n(\text{p}) \mid \text{M} \rangle \mapsto A_1(\text{p}^+), A_2(\text{p}^+), \dots, A_n(\text{p}^+) \quad (3.7)$$

Para a mesma operação, Pouget e Dacier [23] emprega o termo modificação.

3.2.1.8 Priorização

A operação de priorização refere-se ao processo de seleção de alarmes segundo critérios de prioridade [60]. O funcionamento da priorização é representado em (3.8). Uma vez que alarmes são gerados, eles são examinados pela operação P que, através de atributos de interesses, prioriza os alarmes para serem correlacionados. Os benefícios imediatos dessa operação são a redução do número de alarmes e consequentemente a redução de processamento.

$$\text{Priorização: } \langle A_1, A_2, \dots, A_n \mid \text{P} \rangle \mapsto A_k \quad (3.8)$$

A priorização assume um papel importante nos cenários de rede em que várias fontes de alarmes são utilizadas. Sob a perspectiva do gerenciamento de redes, determinados alarmes podem requerer maior atenção do que outros. Por exemplo, alarmes podem ser priorizados com base no nível de ameaça que eles representam para rede, segundo políticas adotadas.

3.2.1.9 Clusterização

A clusterização busca dividir um conjunto de dados em *clusters*, de modo que os dados pertencentes a cada *cluster* sejam semelhantes uns aos outros. Como sugerido por Emad [3], a clusterização pode ser representada em (3.9). Dado um conjunto de alarmes $A_1, A_2, \dots, A_n$ uma determinada operação 0 pode empregar operações booleanas como ∧ (E), ∨ (OU) e ¬ (NOT) a fim de criar *clusters* $C_1, C_2, \dots, C_m$ com alarmes que possuam similaridade ou dissimilaridade, baseado na investigação dos atributos de cada alarme.

$$\text{Clusterização: } \langle A_1, A_2, \dots, A_n \mid \text{O}(\wedge, \vee, \neg) \rangle \mapsto C_1, C_2, \dots, C_m \quad (3.9)$$

Exemplos de atributos dos alarmes que poderiam ser explorados na clusterização são: IP de origem, IP de destino, porta de origem, porta de destino, tipo e tempo de geração do alarme [29] [62].

3.2.1.10 Normalização

Em um ambiente de redes, há uma variedade de tipos e formatos de alarmes. Em busca de maior segurança e detecção de uma gama maior de atividades anômalas, tem sido propostas soluções híbridas envolvendo os sistemas de detecção de anomalias (ADS) e os sistemas de detecção de intrusão (IDS). Nesse tipo de abordagem, o ADS auxilia na detecção de anomalias novas ou desconhecidas, enquanto o IDS detecta aquelas já conhecidas [9]. Além disso, a rede é heterogênea, constituída de dispositivos de diferentes fabricantes que emitem diferentes tipos de alarmes [60] [65].

A utilização de fontes alternativas de dados pode implicar no enriquecimento da correlação. Em contrapartida, para esse cenário heterogêneo é necessário que alguma estratégia seja aplicada para obter um formato comum para os alarmes emitidos pelas diversas fontes antes de serem correlacionados [24] [55]. Esse procedimento é denominado normalização [71]. Como mostrado em (3.10), dado um conjunto de alarmes de entrada, com diferentes tipos/formatos $\text{Ą}a_1, \tilde{\text{ą}}_2, \text{aĂ}_3, \text{a}\bar{\alpha}_4, \ldots, \text{A}_n$, é aplicado sobre ele uma operação N que disponibiliza na saída um conjunto $A_1, A_2, A_3, A_4, \ldots, A_n$ padronizado.

$$\text{Normalização: } \langle \text{Ą}a_1, \tilde{\text{ą}}_2, \text{aĂ}_3, \text{a}\bar{\alpha}_4, \ldots, \text{A}_n | \text{ N} \rangle \longmapsto A_1, A_2, A_3, A_4, \ldots, A_n \qquad (3.10)$$

Nos últimos anos, tem se visto um engajamento buscando a padronização dos alarmes emitidos pelos diferentes IDS [71]. Em 2007, foi definido o protocolo IDMEF (*Intrusion Detection Message Exchange Format*) pela IETF (*Internet Engineering Task Force*). O IDMEF foi concebido tendo como objetivo principal a unificação dos formatos de dados para possibilitar a interoperabilidade entre os diferentes IDSs, comerciais e *open source* [58].

3.2.2 Resumo das operações de correlação

A Tabela 3.1 apresenta um resumo das operações de correlação analisando a característica de perda de informação. A operação é considerada sem perdas quando nenhum alarme é perdido na observação da saída do processo [23].

Tabela 3.1 – Resumo dos principais tipos de operações utilizados na correlação de alarmes.

Operação	*Objetivo*	*Perda de informação*
Compressão	Reduzir um conjunto de alarmes similares para um único alarme	Sim
Supressão	Inibir temporariamente um alarme ou um conjunto de alarmes através de critérios	Não
Filtragem	Seleção de um alarme ou um determinado conjunto baseado em determinados parâmetro/atributos	Depende do tipo de filtro
Generalização	Substituir alarmes por seus respectivos correspondentes na superclasse	Sim
Especialização	Substituir alarmes por alarmes mais específicos na subclasse	Não
Contagem	Gerar um novo alarme quando o número de ocorrências de um determinado tipo atinge um limite pré-definido	Não
Escalação	Atribuir um valor maior a determinado(s) parâmetro(s) do alarme	Não
Priorização	Dar prioridade a determinados alarmes segundo critérios pré-estabelecidos	Sim
Clusterização	Agrupar alarmes baseando em similaridade	Não
Normalização	Transformar alarmes com diferentes tipos em um formato comum	Não

3.3 Métodos para correlação de alarmes

Ao longo de trinta anos de pesquisa, diferentes abordagens têm sido propostas baseadas em diferentes áreas da computação, tais como inteligência artificial, redes neurais, teoria dos grafos e teoria da informação. Nesta seção serão apresentadas as principais soluções propostas, mostrando suas vantagens e desvantagens.

3.3.1 Correlação baseada em regras

Uma das primeiras abordagens na área de correlação de alarmes foi a correlação baseada em regras (RBR – *Rule-Based Reasoning*) [45]. As regras predefinidas especificam as relações de uma determinada condição e sua ação correspondente. A condição, nesse contexto, são os alarmes recebidos. Nesse caso, quando alarmes são emitidos, as regras para os correspondentes alarmes são aplicadas a fim de disparar e/ou sugerir uma ação. Nessa abordagem, são utilizadas regras do tipo SE-ENTÃO que se assemelha com a linguagem natural [34]. Por exemplo, "SE alarme A e B ocorrer, ENTÃO envie um email ao administrador de rede alertando sobre a queda do servidor de DNS". Comumente, o conjunto de regras é armazenado em um banco de dados.

As facilidades de associação de regras com uma linguagem simples faz o RBR ser extensivamente utilizado na prática. Entretanto, para domínios administrativos grandes e ambientes dinâmicos esse modelo se torna improfícuo. As críticas sobre esse modelo referem-se ao alto custo de manutenção/atualização de regras, a impossibilidade de ajuste às mudanças do domínio e escalabilidade [23]. Para Martin-Flatin *et al.* [34], a grande desvantagem das abordagens utilizando RBR se dá pelo fato delas não poderem aprender com eventos históricos e não se adaptarem a situações que não foram previamente codificadas.

3.3.2 Correlação baseada em *codebook*

Em linhas gerais, *codebook* é uma representação da relação causa e efeito. No contexto da correlação de alarmes, é utilizado para representar a relação dos eventos anômalos (causa) e

alarmes (efeito). Por essa definição, pode-se notar que a correlação com *codebook* se assemelha muito à abordagem baseada em regras. O que a difere é que na correlação baseada em regras os alarmes são tratados separadamente ao passo que com *codebook* todos os alarmes são agrupados em um vetor, combinando alarmes com a respectiva assinatura do problema [38] [41].

A técnica *codebook* é descrita em [80]. Como o nome sugere, códigos são gerados para identificar um determinado evento anômalo. Os códigos são constituídos de 0 e 1, gerados a partir de modelos que mapeiam os incidentes ocorridos na rede e os alarmes gerados por eles. O grafo de causalidade tem sido um dos modelos mais aplicados para tal finalidade. Nele é representada a relação de causa-efeito entre um evento anômalo e seus sintomas (alarmes) [46]. A partir do grafo, é gerado o *codebook* que é uma matriz, também conhecida como matriz de correlação.

As entradas da matriz são os valores binários 0 e 1. As linhas da matriz representam os alarmes e as colunas o problema (causa da anomalia, por exemplo, ataque, falhas). Assim, temos que uma entrada 1 na *iésima* posição indica que um alarme a_i foi gerado por um problema p_j [46]. Em síntese, a matriz é formada por um conjunto de vetor (es) de problema(s). Um problema p_i é descrito pelos sintomas (alarmes) que ele emite, isto é, $p_i = (s_1, s_2, \ldots s_n)$.

Depois que a matriz (*codebook*) é construída, o código de um incidente p_k ocorrido é comparado com todos os códigos da matriz, buscando o código correspondente mais próximo. A busca pelo código mais próximo ou de menor distância é dado por uma medida denominada distância de Hamming [23] [69].

As soluções usando *codebook* são apropriadas para casos com única fonte de anomalia, mas não são capazes de lidar com anomalias causadas por múltiplas fontes [80]. Outra desvantagem apontada por Sun *et al.* [38] é que um *codebook* é construído com base na experiência do administrador de rede, tornando necessário que exista um conhecimento especializado para especificar o conjunto de regras. Dessa forma, quando há alterações na rede, é necessária a reconstrução do *codebook*, que para algumas redes pode ser um trabalho árduo para o administrador [38].

3.3.3 Correlação baseada em casos

O paradigma CBR (*Case-Based Reasoning* – Raciocínio Baseado em Casos) é uma técnica de inteligência artificial que se baseia em três princípios fundamentais [34]:

- ✓ Resolução de problemas usando conhecimento de casos já vivenciados;
- ✓ Uma abordagem de conhecimento incremental, no qual a resolução de problemas passados pode ser utilizada para solução de problemas futuros;
- ✓ Capacidade de se adaptar a problemas inéditos.

Como o nome sugere, o modelo CBR utiliza-se de casos. Um caso é constituído de informações referentes ao problema, da possível solução, de resultados da aplicação dessa solução e possíveis atributos que podem ser utilizados na busca de atributos similares de outros casos [34]. Quando uma solução é eficaz para solucionar um determinado problema, ela é armazenada em uma biblioteca para ser utilizada na resolução de problemas posteriores.

Uma analogia do modo de funcionamento de sistemas baseado em casos pode ser feita ao comportamento do cérebro humano que, na ocorrência de um problema não vivenciado, busca, por intermédio de experiências passadas, meios para resolvê-los.

As vantagens de utilizar a correlação baseada em casos é que, no surgimento de novos problemas, as experiências são utilizadas para solucioná-los, e posteriormente, partes úteis são armazenadas [17]. Isso implica que a base de conhecimento vai sendo incrementada no surgimento e resolução de novos problemas. Diferente da limitação de aprendizagem do modelo baseado em regras, um sistema baseado em modelo tem a capacidade de tratar problemas desconhecidos pela capacidade de usar analogia [8]. Outra vantagem refere-se à possibilidade de serem aplicados em cenários dinâmicos, onde há mudanças de configurações de redes, por exemplo.

Por outro lado, para correlação de alarmes em tempo real, esse modelo pode ser ineficiente [46]. Isso se deve a fato da incapacidade de um sistema CBR de solucionar novos problemas, quando não há alguma solução anterior/experiências armazenadas na biblioteca de

casos. Nessas circunstâncias, o tempo gasto para construir a biblioteca de casos pode ser consideravelmente grande [8].

3.3.4 Redes neurais

Inspirada no cérebro humano, uma rede neural artificial consiste de unidades (neurônios) e conexões direcionadas e ponderadas entre elas [23]. Cada neurônio é ligado a seus vizinhos através de conexões sinápticas. No processo de aprendizado de uma rede neural, ajustes são feitos nos pesos das conexões entre os neurônios até que a rede apresente resultados desejados.

Entre as vantagens das redes neurais, está a capacidade de aprendizado e a resistência a inconsistências nos dados (alarmes) de entrada, comuns no processo de correlação [17] [23]. A utilização de redes neurais na correlação de alarmes é uma alternativa atrativa dada a sua capacidade de extrair informações de dados complexos e imprecisos [40]. A desvantagem das redes neurais é que elas podem consumir um grande período de tempo na fase de treinamento [7] [46].

3.3.5 Grafo de dependência

Uma rede de telecomunicações é constituída de diversos componentes interconectados. Devido a essa característica, um evento anômalo ocorrido em um elemento de rede particular pode resultar em um grande número de alarmes dada a propagação da anomalia para seus elementos dependentes [1]. Através do conhecimento dessas dependências, é possível extrair informações para o processo de localização da origem do problema, bem como identificar a propagação da anomalia e o seu efeito na rede [28]. Essas dependências de rede podem ser representadas através de um grafo de dependência [1]. A sigla NDG (*Network Dependency Graph*) é utilizada para denominar um grafo de dependência [7].

Os grafos de dependência são considerados um modelo simples para a correlação de alarmes. Na literatura, muitas soluções têm sido propostas utilizando grafos de dependências para diagnosticar problemas, sendo a maioria delas para identificação da origem do problema

[7]. Na maioria das propostas, é assumida a hipótese de que as dependências são estáticas. Entretanto, em alguns ambientes de redes isso não é verdade [2]. Ambientes de redes dinâmicos estão aumentando consideravelmente, em decorrência do crescimento do número de dispositivos portáteis conectados à rede [34]. São necessárias mais pesquisas nessa direção, sendo assim um dos problemas em aberto na área de correlação de alarmes [34] [46].

3.3.6 Rede Bayesiana

As redes Bayesianas, também conhecidas como redes probabilísticas ou redes de causalidade, são importantes para a representação do conhecimento no campo da Inteligência Artificial [18]. Uma rede Bayesiana é um grafo acíclico dirigido, constituído de nós que representam variáveis aleatórias. Probabilidades condicionais são associadas às arestas e indicam a existência de uma relação de casualidade direta entre os nós conectados [20]. Na rede bayesiana, a informação de um nó depende da informação do seu nó predecessor, conhecido, também, de nó pai. Isso significa que o nó pai tem uma influência causal sobre o nó filho.

Uma rede Bayesiana é considerada uma boa alternativa para representação de conhecimentos probabilísticos que, no contexto de redes, pode representar os relacionamentos de causa e efeitos entre os seus elementos [18]. Para a correlação de alarmes, essa tem sido uma abordagem atrativa pela capacidade de lidar com as incertezas intrínsecas à correlação de alarmes [17] [23]. Uma das principais críticas à correlação através de uma rede Bayesiana é a dificuldade de identificar as relações de casualidade entre as variáveis que se deseja avaliar [20]. Esse é considerado um problema NP-Completo [46].

3.3.7 Soluções híbridas

Pela falta de uma solução que possa ser aplicada a todos ambientes de redes e tipos de anomalias, algumas soluções híbridas têm sido propostas com intuito de atender uma gama maior de problemas e em cenários de redes diversificados [45].

Hanemann [4] propôs uma solução híbrida utilizando raciocínio baseado em regras RBR (*Rule-Based Reasoning*) e raciocínio baseado em casos CBR (*Case-Based Reasoning*). A solução proposta por Marilly *et al*. [41] é constituída pela combinação de redes neurais e técnicas de processamento de sinal. Gürer *et al*. [17] propuseram uma solução que emprega redes neurais e técnicas de raciocínio baseado em casos. Zheng e Qian [80] combinaram *codebook* e a técnica IHU (*Increment Hypothesis Updating*) para resolver o problema de alarmes gerados por múltiplas fontes anômalas, em particular as anomalias geradas por falhas.

Capítulo 4

Trabalhos relacionados

4.1 Redução de alarmes

Pelo fato da ocorrência de um evento anômalo gerar um grande volume de alarmes, soluções têm sido propostas com o objetivo de reduzi-los antes de serem enviados ao administrador de rede. Essa é umas das principais vertentes de pesquisas na área de correlação de alarmes, com propostas utilizando as mais diferentes áreas da computação. Algumas das soluções encontradas na literatura serão apresentadas a seguir.

O trabalho proposto por Chyssler *et al*. [70] apresenta uma arquitetura para redução e correlação de alarmes. Essa solução é voltada para automatizar o processo de correlação, auxiliando o administrador de rede a manipular o grande volume de alarmes gerados pelos IDSs. As operações de filtragem e compressão são previamente utilizadas antes do processo de correlação. Dois tipos de filtragem são propostos: estática e adaptativa. A filtragem estática é modelada por regras pré-determinadas. A filtragem adaptativa é realizada por um classificador Bayesiano Ingênuo (Naive Bayes - NB). Tanto a filtragem estática quanto a adaptativa são aplicadas para descartar alarmes reportados pelo IDS que não correspondem a atividades maliciosas. IDSs, comumente, emitem mensagens informando sobre erros de configurações, por exemplo, as quais podem não ser de interesse do administrador de rede. Após a filtragem, é aplicada a compressão, na qual alarmes com o mesmo endereço IP de origem, destino e assinatura disparados dentro da mesma janela de tempo são agrupados. Como resultado, Chyssler *et al*. mostram que através da filtragem e compressão é possível reduzir, consideravelmente, o volume de alarmes.

Al-Mamory e Zhang [62] aplicaram a clusterização para reduzir o volume de alarmes. O algoritmo proposto é baseado na técnica de clusterização *Nearest Neighbor* (vizinho mais próximo). Os atributos dos alarmes utilizados na clusterização são: IP de origem, IP de destino, porta de origem, porta de destino, data e hora de geração do alarme e o tipo o alarme. Essa solução usa o conceito de alarme *generalizado*. Um alarme *generalizado* é aquele que caracteriza um *cluster*. No surgimento de um novo alarme, é calculada a medida de distância dos seus atributos para os atributos do alarme generalizado de cada *cluster*. O novo alarme é agrupado ao *cluster* que possui a distância mais próxima do alarme generalizado. Segundo os autores, a taxa média de redução obtida pela solução foi de 74% do volume total de alarmes.

Com o objetivo de minimizar a quantidade de alarmes gerados por falhas dos dispositivos de rede, uma abordagem é proposta por Bellec e Kechadi [29]. Essa proposta é divida em duas fases: pré-processamento e clusterização. Na fase de pré-processamento, são realizadas as operações de eliminação e filtragem de alarmes. Alarmes não condizentes com o formato esperado e duplicado são eliminados, visando a redução da quantidade e a melhora na qualidade dos alarmes que serão analisados posteriormente. O tipo de filtragem aplicado neste trabalho consiste em separar alarmes que compartilham o mesmo conteúdo em diferentes conjuntos, onde esses conjuntos serão classificados com o objetivo de obter aqueles com maior importância para a fase de clusterização. O conteúdo é definido pela mensagem e o ID da fonte que emitiu o alarme. Na fase de clusterização, é proposto um algoritmo denominado FEC_k (*Fuzzy Event Correlation*). O objetivo do algoritmo consiste em gerar *clusters* baseado nos alarmes com maior relevância, escolhidos previamente na fase de pré-processamento. O algoritmo gera *clusters* contendo informações pertinentes às possíveis causas da falha. No entanto, o último passo, que consiste na verificação de qual é a mais provável fonte do problema fica a cargo do administrador de rede.

Mais recentemente, um sistema que utiliza redes neurais e algoritmo de clusterização foi desenvolvido por Tjhai *et al.* [25]. O objetivo central do sistema está em identificar os alarmes redundantes e com baixa prioridade gerados pelo sistema de detecção, bem como se esses são verdadeiros ou falsos. A solução proposta é realizada em duas etapas. A primeira etapa consiste na utilização de uma rede SOM (*Self Organizing Map*) para classificar e agregar alarmes que

possuem o mesmo *timestamp* e um mesmo endereço IP de origem e de destino, sendo consideradas indicações de uma mesma atividade anômala. A primeira fase resulta no agrupamento dos alarmes redundantes denominado de *cluster*. Na segunda fase, o algoritmo K-*means* é aplicado para rotular os alarmes pertencentes aos *clusters* produzidos na primeira classificação como alarmes verdadeiros ou falsos. Os autores enfatizam que esse passo é importante para que o número de alarmes falsos seja minimizado antes de ser apresentado ao administrador de rede.

4.2 Localização da causa raiz

O processo de inferir a fonte do problema a partir da observação de um conjunto de evidências (alarmes) é denominado na literatura como localização da causa raiz [46] [62]. Sob a perspectiva da gerência de rede, são necessárias técnicas que busquem descrever as razões da ocorrência do incidente. A rápida descoberta da fonte da anomalia traz dentre outros benefícios, a redução dos efeitos da degradação dos serviços prestados pela rede e minimização da insatisfação dos usuários. Entretanto, esta é uma tarefa complexa [2] [46]. A seguir apresentaremos estratégias propostas para a correlação de alarmes voltada para a localização da causa raiz.

Um dos trabalhos pioneiros na área de correlação de alarmes, voltado para localização da causa raiz, foi proposto por Katzela e Schwartz [28]. Nessa proposta, os elementos de rede e suas respectivas dependências são modelados por um grafo de dependência de rede (NDG – *Network Dependency Graph*). Um peso é dado para as arestas do NDG. Os pesos nas arestas representam a probabilidade de um incidente ocorrido em um determinado dispositivo de rede se propagar para seus vizinhos. De igual modo, uma probabilidade condicional conhecida *a priori* é associada a cada nó que representa a probabilidade do elemento de rede falhar. Esse trabalho mostra que a localização da falha é um problema NP-Completo. Assim, um algoritmo heurístico é proposto. Os resultados mostram que essa solução é capaz de localizar a origem da falha, porém alarmes espúrios não são tratados nessa solução.

Mohamed e Basir [7] propuseram um sistema de correlação de alarmes distribuído, voltado para localização da causa raiz de falhas. Nessa abordagem, cada agente inteligente é responsável por um determinado domínio de rede monitorada. O agente é responsável por coletar e correlacionar os alarmes gerados em seu domínio. Essa solução é baseada na teoria da evidência proposta por Dempster-Shafer. O agente considera cada alarme coletado como uma peça ou parte da evidência. De forma individual e independente, os agentes inteligentes relatam seus alarmes para um gestor de alto nível denominado AM (*Agent Manager*). O AM busca correlacionar os alarmes recebidos de forma individual pelos agentes a fim de inferir a real causa raiz do problema.

A solução proposta por Liu *et al.* [72] tem por objetivo auxiliar os administradores e engenheiros de redes a localizarem falhas em redes híbridas. Diferente de outros trabalhos voltados para redes IP, essa abordagem busca a correlação de alarmes de redes DWDM (*Dense Wavelenght Division Multiplexing*), SDH (*Synchronous Digital Hierarchy*) e IP.

A proposta de Tang *et al.* [78] consiste na combinação do monitoramento passivo e ativo para localização de falhas. Os autores destacam que embora a abordagem ativa seja mais eficiente para localização de falhas, pelo tempo de identificação e maior imunidade a ruídos da rede, elas não são escaláveis de forma a identificar falhas simultâneas na rede. Por outro lado, a localização de falhas através da abordagem passiva causa menos sobrecarga ao tráfego da rede, mas leva mais tempo para identificar os sintomas de falhas. Explorando as vantagens e características de cada monitoramento eles propuseram um *framework* denominado de AIR (*Active Integrated fault Reasoning*).

4.3 Entropia

O uso do conceito de entropia em diversas áreas do gerenciamento de redes (detecção de anomalia, correlação de alarmes, tomografia de redes, etc.) será apresentado nesta seção.

Ekaette e Far [20] propuseram um *framework* para automatizar o gerenciamento de falhas. É proposto um gerenciamento distribuído que utiliza agentes inteligentes capazes de identificar atividades anômalas causadas por falhas em seus domínios. Esses agentes são responsáveis pela

detecção e correlação de alarmes a fim de apresentar um diagnóstico do incidente ocorrido. Pela forma em que os agentes estão distribuídos, diversas hipóteses de falhas pode ser obervadas pelos *n*-agentes. Nesse trabalho, o conceito de entropia é aplicado como uma medida de quantificação do grau de incerteza do resultado dos *n*-agentes. A hipótese mais provável para o equipamento responsável pela falha é dada pelo menor valor de entropia.

Marnerides *et al.* [6] desenvolveram um método para detecção de anomalias em uma rede autônoma. Segundo eles, uma rede autônoma é aquela que emprega mecanismos automatizados tais como, autoadaptação, autoconfiguração e autoproteção a fim de minimizar a necessidade de configuração estática e intervenção do administrador de rede. Eles enfatizam que a detecção do tráfego anômalo nessas redes é uma tarefa desafiadora, pois no início do surgimento da anomalia, a rede deve se comportar de forma inteligente e utilizar meios que minimizem os seus efeitos. Uma medida utilizando entropia é aplicada para estimar se o fluxo de dados é anômalo. A entropia é calculada utilizando probabilidades de ocorrência de determinados atributos de um fluxo como, por exemplo, o número de pacotes. Esse processo é executado periodicamente, visando identificar se uma atividade anômala está se iniciando na rede. Para isso, uma comparação é feita entre o novo valor de entropia e o previamente calculado.

Um método para detectar anomalias usando entropia foi recentemente desenvolvido por Wang e Wu [75]. Esse método utiliza as informações obtidas dos pacotes IP: endereços de origem, destino e portas de origem e destino. Através da distribuição dessas informações, busca-se detectar as anomalias, em particular os ataques DDoS, *worm* e *port scan*. Podemos tomar como exemplo a distribuição das informações obtidas de pacotes de um fluxo anômalo gerado por um ataque DDoS. Nesse caso, a medição do fluxo destinado ao nó alvo mostraria uma concentração de pacotes com diferentes endereços IP e portas de origem, mas com o mesmo endereço e porta de destino. Antes de passar a monitorar a rede, a proposta passa por uma fase de treinamento. Nessa fase, os dados do fluxo são capturados via NetFlow e armazenados em uma base de dados. As probabilidades para o cálculo da entropia são geradas através de cálculos estatísticos aplicados aos dados armazenados. A entropia é computada durante um período de medição em intervalos de tempo. Um limiar é criado através do cálculo da média e variância dos valores de

entropia. A partir desse ponto se encerra o treinamento. Durante o monitoramento da rede, o valor de entropia é comparado com o limiar criado no treinamento e alarmes são emitidos quando esse valor está acima do valor definido pelo limiar.

Diagnosticar falhas em redes usando *adaptive probing* foi o objetivo de Natu e Sethi [42]. A técnica apresentada utiliza *probes* como *ping* e *traceroute* a fim de localizar elementos de redes responsáveis pela(s) falha(s). A detecção de falhas é realizada da seguinte maneira. Ferramentas como *ping* ou *traceroute* são utilizadas para verificar o estado do dispositivo de rede. Como essas ferramentas geram tráfego adicional na rede, é preciso otimizar esta medição. É realizado, então, um processo de medição adaptativa, no qual as áreas com maior probabilidade de falhas são mais frequentemente inspecionadas. Para tanto, é utilizado um critério baseado no conceito de entropia. São escolhidos os dispositivos que possuem menor incerteza/menor entropia, ou seja, aqueles dispositivos que possuem maiores chances de serem causadores de falhas. A partir da seleção do dispositivo com menor valor de entropia é definida a estratégia de medição.

4.4 Tomografia de rede

Metodologias que buscam avaliar e predizer as características internas da rede como perda, atraso de pacotes e estimativa da taxa de dados em um fluxo são fundamentais para a gerência de uma rede. Porém, no cenário heterogêneo e distribuído da Internet, a aquisição dessas informações se torna um processo desafiador. Dessa forma, as soluções tradicionais voltadas para um único domínio administrativo não são suficientes para a gerência de uma rede distribuída na Internet. Com esse propósito, surgiu a tomografia de rede, sendo um campo emergente de pesquisa [19]. O termo tomografia de rede foi cunhado por Vardi [79], em 1996, devido à semelhança entre a inferência de rede e a tomografia médica.

Métodos tomográficos utilizam pontos de medições fisicamente distribuídos na rede, responsáveis pelo envio de mensagens de sondagens (*probes*), também chamados de mensagens de teste, a fim de realizar as medições dos elementos de redes. No contexto da tomografia de redes, esses pontos são denominados de *beacon* [35][59].

A Figura 4.1 mostra o funcionamento do método de tomografia apresentado em [59]. Nela é demonstrado o procedimento de inspeção do enlace $l_3 = (N_2, N_3)$. O *beacon* envia um par de mensagens de sondagem (por exemplo, ICMP *echo request*) quase que simultâneos para N_2 e N_3. Apenas uma das mensagens atravessa o enlace e chega em N_3. Cada ponto N_2 e N_3 envia uma resposta para o *beacon* (ICMP *echo reply*).

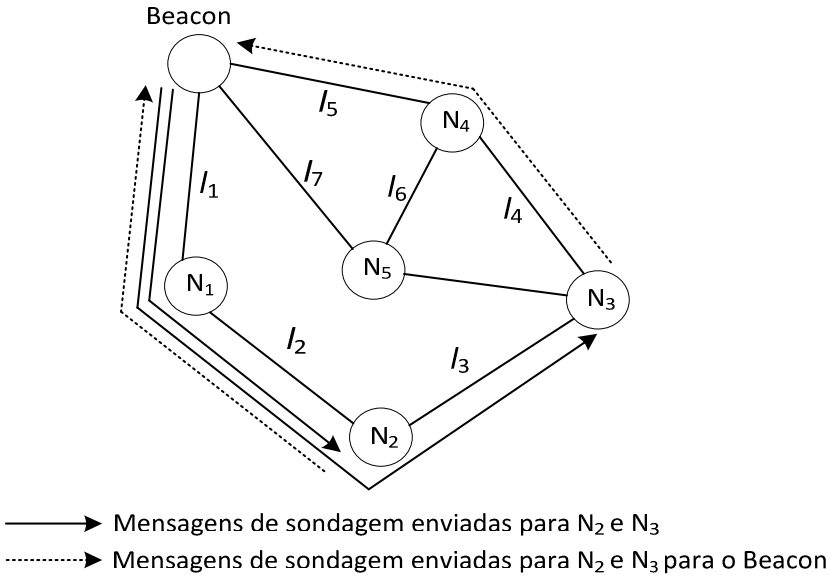

Figura 4.1 – Funcionamento dos métodos de tomografia de rede

Como proposto por Moulierac e Molnar [35], caso fosse desejável verificar o atraso do pacote transmitido pelo enlace l_3, poderíamos fazê-lo da seguinte forma: ao enviar as mensagens para N_2 e N_3, o *beacon* armazenaria o tempo de envio, e.g., t_1 e t_2. Ao receber a resposta em um tempo t_1' e t_2', o atraso seria calculado pelo *beacon* utilizando (4.1).

$$ATRASO(n_2, n_3) = \frac{\left|(t_2' - t_2) - (t_1' - t_1)\right|}{2} \quad (4.1)$$

Utilizando-se da tomografia de rede, Agrawal *et al.* [61] propuseram um método para realizar a inferência em nível de enlace com o objetivo de identificar enlaces anômalos responsáveis por atrasos e perdas de pacotes. Essa solução utiliza o monitoramento passivo, no qual medidores pontuais, estrategicamente distribuídos, são utilizados para obter dados sobre a performance do tráfego que passa por eles. O método aplicado nesse trabalho é denominado na

literatura como método *tomográfico booleano.* Esse método utiliza rótulos para a identificação do estado do enlace. Por exemplo, o estado de um enlace pode ser considerado como operante ou inoperante. Se todos os enlaces de um caminho estão operantes, esse caminho é rotulado como *"bom"* (valor booleano 0); por outro lado, quando ao menos um dos enlaces do caminho está inoperante, este é rotulado como "*ruim*" (valor booleano 1) [61].

No trabalho proposto por Barford *et al*. [52], é apresentado um *framework* para o monitoramento, detecção e localização de anomalias na rede. Nesse *framework* há um algoritmo que busca as anomalias nos caminhos utilizando *probes* juntamente com um limiar pré-definido. Quando uma anomalia é detectada, ou seja, os dados coletados são maiores que os do limiar predefinido, um alarme é gerado. Outro algoritmo desse *framework* é responsável pela determinação de qual (is) caminho (s) deve(m) ser verificado (s) em um determinado momento. O objetivo desse algoritmo é assegurar que todos os enlaces sejam frequentemente monitorados, a fim de encontrar as anomalias o mais rápido possível. Um terceiro componente do *framework* é um algoritmo de localização que visa identificar os enlaces que possuem um tráfego anômalo, disparado quando uma anomalia é detectada no caminho.

Huang *et al*. [77] propuseram uma solução voltada para detectar e identificar a origem de falhas na rede. O método aplicado utiliza o monitoramento ativo em que os *beacons* enviam mensagens através das ferramentas *ping* e *traceroute* para coletar informações sobre o estado dos elementos de redes. As informações coletadas de todos os monitores formam a entrada para o algoritmo NetDiagnoser, que busca identificar a origem da falha causada por problemas em enlaces ou nos dispositivos monitorados. Esse processo toma como entrada a topologia da rede conhecida *a priori*, e o conjunto de caminhos que falharam, resultando em um conjunto das possíveis localizações das falhas na rede monitorada.

No trabalho realizado por Rizzo *et al.* [73], os autores destacam pontos negativos pertinentes às abordagens tomográficas. Pelo fato dos métodos de tomografia terem que realizar a soma de todas as combinações possíveis de atrasos internos obtidos na rede, esses cálculos se tornam complexos com um tempo computacional não polinomial. Essa complexidade computacional e problemas de escalabilidade das soluções têm sido um das grandes limitações para se utilizar a tomografia de redes em tempo real. Rizzo *et al.* mostraram que é possível lidar

com esses problemas, de modo que a tomografia possa ser aplicada, na prática, no cenários de grandes redes. Um dos resultados mais importantes deste trabalho é a utilização de uma nova versão modificada do algoritmo *message-passing*, que possibilita os cálculos de forma muito mais rápida através da recursividade.

Capítulo 5

Proposta para correlação de alarmes

5.1 Sistema de correlação de alarmes

O sistema de correlação de alarmes é uma peça fundamental de um NMS (*Network Management System*) [9] [30] [71]. Porém, projetar um sistema de correlação de alarmes que seja eficaz na prática, é uma tarefa complexa [65]. Cada solução pode apresentar melhores resultados do que outras em circunstâncias particulares, não havendo uma solução genérica capaz de resolver todos os problemas pertinentes à correlação e aplicável a qualquer tipo de rede [34]. Isso se dá pelas diferenças e particularidades de cada rede, tais como a topologia, tipos de alarmes, tipos de anomalias e suas causas.

Não há métricas voltadas para a comparação das soluções de correlação de alarmes [45]. No entanto, alguns questionamentos podem ser feitos a fim de identificar pontos importantes para o desenvolvimento de um sistema de correlação de alarmes:

- Qual é a relevância das informações providas pelo sistema para diagnosticar a anomalia?
- O sistema automatiza o processo, dispensando a necessidade da intervenção humana?
- Qual é o custo de implantação do sistema?
- Qual/Quais o(s) tipo(s) de alarme(s) esperado(s) como entrada do sistema? Alarmes espúrios são esperados ou deve-se assumir que todos alarmes serão corretamente gerados?
- Como o resultado da correlação é apresentado ao administrador?
- Quais tipos de anomalias o sistema aborda?

Nesta dissertação é apresentado um sistema para correlação de alarmes, voltado para inferir a origem e o destino da anomalia. A Figura 5.1 apresenta uma visão do sistema, o qual é divido em três camadas: (1) camada de pré-processamento; (2) camada de correlação; (3) camada de apresentação.

A camada de pré-processamento é responsável pela compressão dos alarmes primitivos utilizando o atributo espacial e temporal de cada alarme. Através desses atributos, alarmes disparados pelo mesmo dispositivo na ocorrência da anomalia são reduzidos a um único alarme denominado DLA (*Device Level Alarm* - Alarme em Nível de Equipamento). Utilizando os DLAs produzidos pela camada de pré-processamento e informações sobre a topologia de rede, a camada de correlação implementa um método para inferir o caminho de propagação da anomalia, bem como sua origem e destino. O resultado da inferência da camada de correlação é exibido na camada de apresentação. Essa camada fornece uma visão global da rede de forma gráfica, possibilitando a visualização e um rápido entendimento do impacto causado pela propagação da anomalia na rede.

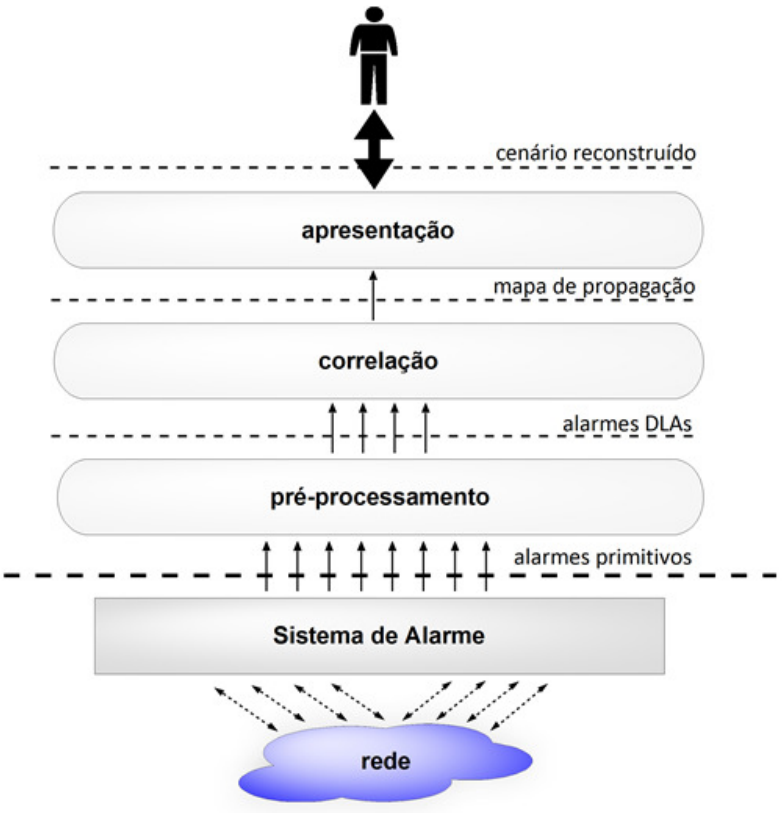

Figura 5.1 – Visão geral do sistema de correlação.

A divisão do sistema de correlação em camadas se deu por dois fatores principais. O primeiro se deve ao fato de que a correlação de alarmes é uma tarefa complexa para ser realizada em única fase. O segundo fator refere-se à utilização do sistema em diferentes redes com pequenos ajustes em determinada(s) camada(s). Formatos e tipos de alarmes podem variar de uma rede para outra, dado os diferentes equipamentos e sistemas de alarmes como é mostrado em [67]. Sendo assim, modificações na camada de pré-processamento são necessárias para atender os diferentes alarmes de uma rede em particular, porém essas modificações se tornam transparentes para as camadas superiores, de correlação e apresentação.

5.2 Sistema gerador de alarmes

O sistema gerador de alarmes utilizado neste trabalho foi inicialmente proposto por Proença Junior [43] e é mostrado na Figura 5.2. Nesse sistema de alarmes, dados coletados de um objeto SNMP são analisados a fim de detectar alterações inesperadas no comportamento da rede. Os dados coletados dos objetos SNMP são caracterizados através do modelo BLGBA (*Baseline para* Gerenciamento de *Backbone* Automático) proposto por Proença Junior [43], gerando perfis de comportamento normal, denominados DSNS (*Digital Signature of Network Segment* - Assinatura Digital do Segmento de Rede) ou *baseline*.

Para geração do DSNS, o modelo BLGBA utiliza uma variação do cálculo da moda. O valor esperado para cada segundo do dia é baseado nos respectivos segundos de semanas anteriores obtidos de dados históricos. Os valores coletados em semanas anteriores são agrupados, segundo suas frequências, em cinco classes baseando-se na diferença entre o maior Gr_i e o menor valor Sm_i da amostra. Através da divisão dessa diferença por cinco, é obtida a amplitude h entre as classes, como é mostrado em (5.1).

$$h = \frac{(Gr_i - Sm_i)}{5} \tag{5.1}$$

Posteriormente, os limites de cada classe, representados por L_{Ck} são obtidos através de (5.2), onde Ck representa a k-ésima classe, com $k = \{1, \dots, 5\}$.

$$L_{Ck} = Sm_i + (h \times k) \tag{5.2}$$

Por fim, o maior elemento da classe com frequência acumulada maior ou igual a 80%, denominado de bl_i, é escolhido para constituir o DSNS.

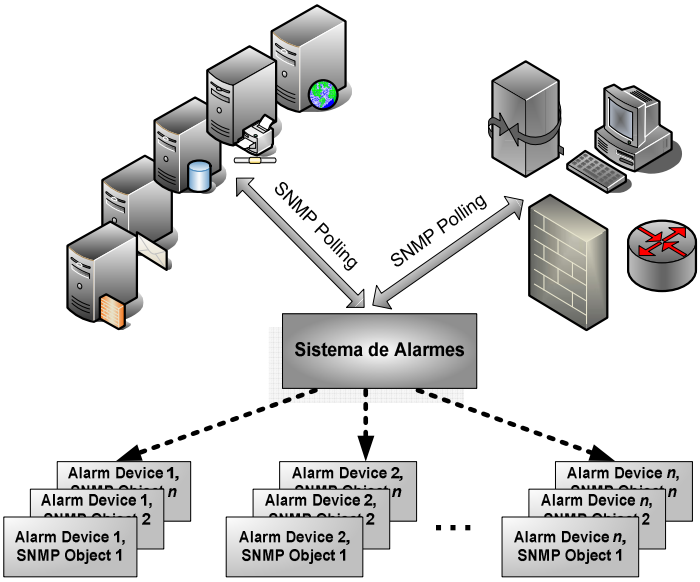

Figura 5.2 – Modo de funcionamento do sistema de alarmes.

Após a caracterização do tráfego, os DSNS são comparados com os dados reais obtidos do tráfego de rede. Essa comparação é realizada através de um algoritmo de histerese [13], representado pelo diagrama de atividades na Figura 5.3. O algoritmo utiliza um parâmetro $\boldsymbol{\delta}$ que representa o número de violações toleradas na comparação dos dados reais com o DSNS. Quando o valor de leitura é superior ao valor do DSNS é gerado um evento tipo 1. Na identificação do evento 1, um contador temporal pré-definido denominado de *intervalo de histerese* é iniciado. Ao identificar o evento 1, a leitura que ultrapassou o DSNS passa a ser o novo limite. Quando é superado o DSNS e o limite corrente, a ocorrência de um evento 2 é identificada. O alarme é gerado quando a contagem de eventos 2 em um único *intervalo de histerese* é maior do que o valor do $\boldsymbol{\delta}$. Cada alarme disparado contém os seguintes atributos: Nome do objeto SNMP, endereço IP e a porta do dispositivo e *timestamp*.

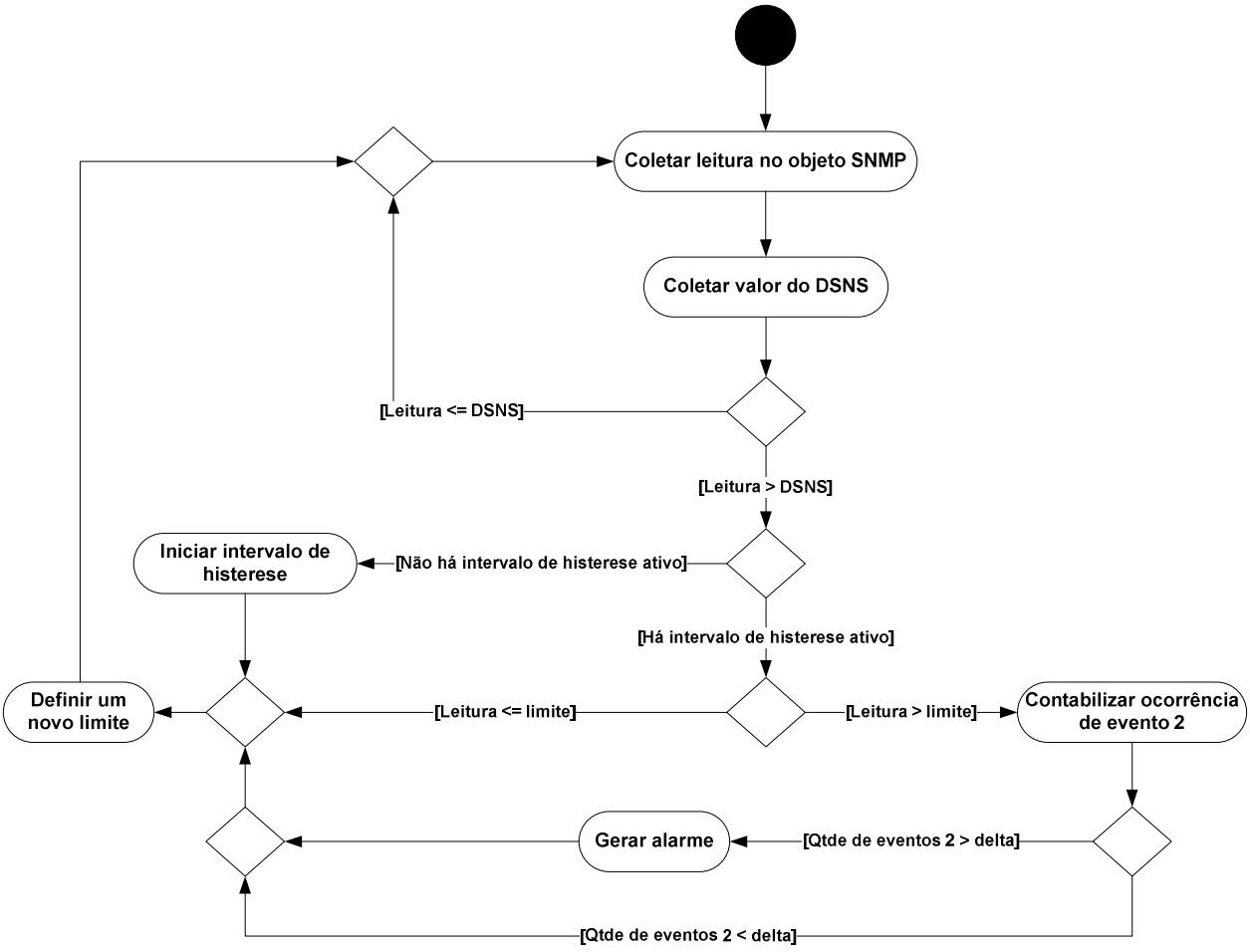

Figura 5.3 - Diagrama de atividades para o algoritmo de histerese [13].

O sistema de alarmes monitora um conjunto de objetos SNMP, oferecendo diferentes informações sobre o equipamento monitorado. Dentre os objetos monitorados, neste trabalho foram utilizados os seguintes: (i) *ifInOctets* e *ifOutOctets*, que contêm a quantidade de octetos recebidos e enviados, respectivamente, pela interface de rede. (ii) *ipInReceives* e *ipOutRequests*, que contêm o número de datagramas IP recebidos e enviados e (iii) *tcpInSegs* e *tcpOutSegs*, que contêm o número total de segmentos TCP recebidos e enviados.

Essa dissertação é a evolução dos trabalhos de Proença Junior [43] e Zarpelão [13] como mostra a Figura 5.4. Em 2005 foi desenvolvido o modelo de caracterização e geração de *baselines* para segmentos de redes proposto por Proença Junior [43]. Em 2010, Zarpelão [13] desenvolveu um sistema de detecção de anomalias que utiliza os *baselines* proposto por Proença Junior para detectar eventos anômalos na rede. Embora o objetivo principal de Zarpelão fosse detectar as

anomalias de volume, ele propôs também uma modelo inicial para identificar a propagação da anomalia na rede. Nosso trabalho apresenta um sistema de correlação de alarmes que, através dos alarmes gerados pelo sistema de detecção de anomalias proposto por Zarpelão [13], busca traçar o caminho de propagação do tráfego anômalo identificando sua fonte e destino. Diferente do trabalho de Zarpelão, nosso trabalho considera a existência de alarmes espúrios e utiliza uma medida baseada na entropia generalizada para identificar o caminho mais provável de propagação da anomalia.

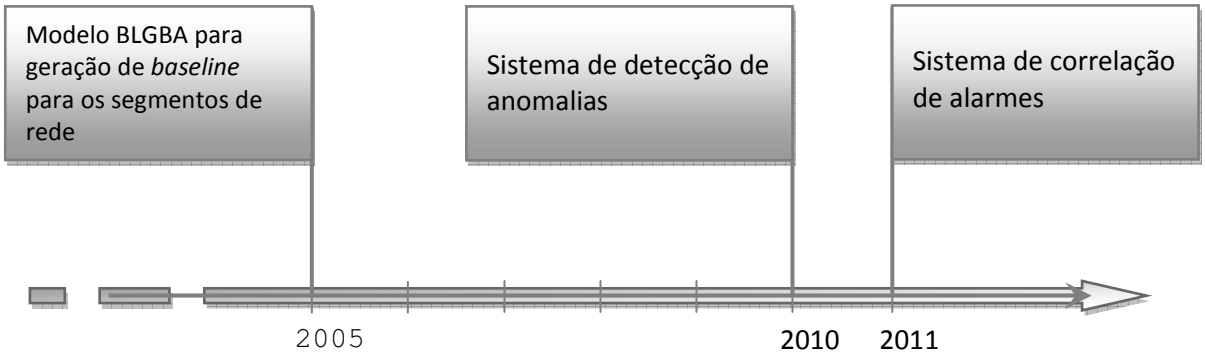

Figura 5.4 – Evolução dos trabalhos.

5.3 Camada de pré-processamento

A Figura 5.1 mostra o sistema de alarmes atuando como um coletor de dados da rede que detecta a existência de uma anomalia na rede. Vale salientar que, na prática, podem ser utilizados vários pontos de monitoramento e sistemas de alarmes geograficamente distribuídos na rede. Quando uma discrepância é detectada no tráfego normal de rede, alarmes são emitidos com o propósito de notificação. Esses alarmes são conhecidos no contexto de correlação de alarmes como alarmes primitivos e podem chegar a centenas de milhares diariamente em uma rede de médio e grande porte [29] [32] [34].

O primeiro obstáculo para a correlação de alarmes está relacionado à alta taxa de alarmes primitivos consecutivos que são gerados na ocorrência de uma anomalia. Esse volume de alarmes é resultado de dois fatores principais. Primeiro, os efeitos da anomalia serão observados enquanto uma medida não for tomada para resolvê-la. O segundo fator é a propagação da

anomalia pela rede [32]. Pela característica de persistência da anomalia, isto é, ela só desaparece quando solucionada, o sistema de alarme continua emitindo alarmes referentes ao mesmo problema. Dessa forma, esses alarmes se tornam dados redundantes, pois sinalizam a mesma anomalia e não acrescentam um valor semântico para a correlação. Assim, é plausível dizer que a remoção desses alarmes é impulsionada pela necessidade de minimizar o número de alarmes que serão analisados posteriormente, pois, segundo Tjhai *et al.* [25], são supérfluos e não acrescentam nada ao processo de correlação. Mesmo que haja um grande volume de alarmes chegando à camada de pré-processamento, apenas uma fração desses alarmes é realmente importante para a correlação. Isso significa que um sistema de correlação de alarmes deve ser capaz de filtrar alarmes relevantes e de significância para o objetivo da proposta.

5.3.1 Classificação dos alarmes para identificar a propagação da anomalia

Para o propósito deste trabalho, os alarmes disparados para os objetos SNMP são divididos em dois grupos. A Figura 5.5 apresenta os grupos e seus objetos. O grupo 1 (G1) contém os objetos relacionados aos alarmes classificados como alarmes de entrada. Alarmes disparados para os objetos do G1 são indicadores de que o tráfego anômalo ingressou no dispositivo. Por outro lado, os objetos pertencentes ao grupo 2 (G2) são classificados como alarmes de saída e indicam que o tráfego anômalo partiu do dispositivo.

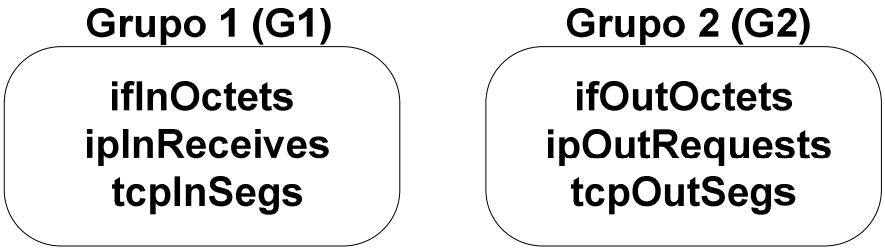

Figura 5.5 - Classificação dos alarmes gerados para os objetos SNMP.

5.3.2 Redução do volume de alarmes

A fim de reduzir o volume de alarmes, exploramos dois atributos pertinentes a cada alarme: *quando* e *onde*. O atributo *onde* contém a descrição da localização do elemento na rede

monitorada para o qual o alarme foi disparado. Tradicionalmente, essa informação é dada pelo endereço IP (*Internet Protocol*) e porta do dispositivo. *Quando* refere-se ao momento em que o problema foi detectado no elemento de rede [1]. Logo, o centro da nossa solução é a noção de similaridade entre os alarmes em termos de suas informações espaciais e temporais, isto é, com base nos atributos *onde* e *quando*, alarmes disparados pelo mesmo dispositivo para um evento anômalo são comprimidos, sendo reduzidos para um único alarme.

Seja W um conjunto, contendo x elementos de redes afetados por uma anomalia. Cada $w_k \in W$ possui um conjunto O de objetos SNMP contendo um ou mais alarmes como é apresentado a seguir

$$w_k = \{O_{k1}, O_{k2}, \dots, O_{kn}\} \tag{5.3}$$

onde

$$O_{ki} = \{a_{i1}, \dots, a_{im}\} \tag{5.4}$$

para $k = \{1, \dots, x\}$ e $i = \{1, \dots, n\}$, onde n é o número de objetos monitorados em cada dispositivo e m o número de alarmes gerados para cada objeto.

Analisando individualmente cada conjunto O_{ki} e partindo do pressuposto de que os alarmes estão sendo gerados pelo mesmo evento anômalo, observamos que os alarmes a_{ij}, para $j = \{2, \dots, m\}$ são alarmes redundantes, pois referem ao mesmo problema e não agregam informações para a correlação. Se os alarmes indicam a mesma anomalia, eles podem ser comprimidos resultando em um alarme com maior conteúdo semântico e, portanto mais significativo. Essa medida reduz o volume de alarmes para cada dispositivo afetado pelo tráfego anômalo, que diretamente minimiza o conjunto de alarmes como um todo.

Com esse objetivo, utilizamos a operação de compressão de alarmes. Como descrito na seção 3.2.1.1, a compressão reduz várias ocorrências do mesmo alarme em um único alarme. Em nosso contexto, a compressão é realizada da seguinte forma. Para cada O_{ki} é obtido um único alarme utilizando o critério de menor *timestamp*, isto é, o alarme a_{i1} é o primeiro alarme disparado para o objeto quando o dispositivo foi afetado pela anomalia. Nessas circunstâncias, temos que o número $AD_{inicial}$ (Alarmes por Dispositivo), antes da compressão era

$$AD_{inicial} = \sum_{i=1}^{n} \#O_{ki} \tag{5.5}$$

e após a compressão, o número de alarmes de cada O_{ki} é reduzido para um único alarme (a_{i1}), onde a quantidade de alarmes $AD_{comprimidos}$ para cada dispositivo é minimizada para

$$AD_{comprimidos} = \sum_{i=1}^{n} 1 = n_k \tag{5.6}$$

O número de alarmes que chega à camada de pré-processamento $AT_{inicial}$(Alarmes Total) é dado por

$$AT_{inicial} = \sum_{k=1}^{x} \sum_{i=1}^{n} \#O_{ki} \tag{5.7}$$

minimizado para

$$AT_{comprimidos} = \sum_{k=1}^{x} AD_{comprimidos} \tag{5.8}$$

A TRA (Taxa de Redução de Alarmes) é obtida pela razão entre o número de alarmes resultante da compressão sobre o número total de alarmes inicialmente dado, ou seja,

$$\mathbf{TRA} = \left(1 - \frac{AT_{comprimidos}}{AT_{inicial}}\right) \times 100 \tag{5.9}$$

Após a compressão, buscamos agregar mais informações aos alarmes resultantes de cada dispositivo afetado pela anomalia. Os alarmes primitivos representados pelos objetos SNMP dão lugar aos DLAs.

Cada DLA passa a representar o comportamento do dispositivo diante da atividade anômala. Através dele, temos conhecimento se o tráfego anômalo ingressou no dispositivo ou partiu dele, ou se dentro de um Δ_t o tráfego anômalo chegou e partiu dele.

```
1: @G1
2: Roteador, Firewall
3: @G2
4: Firewall, Switch
5: @G1-G2
6: Switch, Proxy
...
```

Figura 5.6 **-** Exemplos de DLAs.

Cada par de linhas n_i e n_{i+1}, onde $n \in \mathbb{N}$ e ímpar, representado na Figura 5.6, indica um DLA. As anotações @G1, @G2 e @G1-G2 possuem os seguintes significados:

- ✓ @G1: alarmes que pertencem ao grupo 1, indicando que a anomalia ingressou no dispositivo. As linhas 1 e 2 da Figura 5.6 mostram que foram emitidos alarmes do grupo 1 para uma porta do equipamento *Firewall* que o conecta ao Roteador;
- ✓ @G2: alarmes que pertencem ao grupo 2, indicando que a anomalia partiu do dispositivo. As linhas 3 e 4 da Figura 5.6 mostram que foram emitidos alarmes do grupo 2 para uma porta do equipamento *Firewall* que o conecta ao *Switch*;
- ✓ @G1-G2: alarmes que pertencem aos grupos 1 e 2, indicando que dentro de uma janela de tempo um tráfego anômalo chegou e saiu do dispositivo. As linhas 5 e 6 da Figura 5.6 mostram que foram emitidos alarmes dos grupos 1 e 2 para uma mesma porta do *Switch* que o conecta ao servidor *Proxy*.

O objetivo de se utilizar a representação @Gx em vez dos nomes dos objetos SNMP (por exemplo, *ifInOctets* e *tcpOutSegs*) é por motivos de generalização. Embora nesta dissertação o protocolo SNMP seja utilizado como fonte de dados, outras fontes de dados são utilizadas para a detecção de anomalias. Caso a fonte de dados seja alterada e os alarmes emitidos sejam diferentes, esse efeito não irá se propagar para as camadas superiores do sistema de correlação.

5.4 Camada de correlação

Essa camada é responsável por correlacionar os alarmes previamente gerados pela camada de pré-processamento. Através dos DLAs e informações sobre a topologia da rede, busca-se produzir a melhor explicação sobre o incidente ocorrido na rede. A modelagem da rede, uma medida baseada em entropia para minimizar o impacto dos alarmes espúrios (alarmes falsos e perdidos) na correlação, e o método para identificar o caminho de propagação da anomalia serão apresentados nesta seção.

5.4.1 Modelagem da rede

As redes de telecomunicações são constituídas de elementos dependentes entre si. Um incidente ocorrido em um dispositivo x_k pode diretamente afetar os dispositivos $x_{k+1}, x_{k+2} \dots x_n$, dado seus relacionamentos de dependências [7] [28] [32]. O funcionamento efetivo de uma rede depende do funcionamento das sub-redes que depende do funcionamento dos dispositivos que as compõem [1]. A fim de representar essas dependências, um grafo de dependência é utilizado. Grafos de dependência têm sido aplicados em diversas áreas de conhecimento como forma de representação de relacionamentos ou dependências entre objetos. O interesse na utilização dos grafos de dependências ocorre devido a sua simplicidade para modelagem dos problemas relacionados à identificação das possíveis causas e diagnósticos de eventos anômalos ocorridos na rede [1] [2].

O conhecimento das dependências entre os dispositivos de redes traz informações importantes para auxiliar na resolução de uma variedade de problemas de redes, como a correlação de alarmes, identificação da propagação da anomalia pela rede e localização da origem do problema [2][28].

Neste trabalho, a rede é modelada por um grafo direcionado $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. O conjunto finito e não vazio de vértices $\mathcal{V}$ representa os elementos de redes, enquanto o conjunto de arestas $\mathcal{E}$ representa os enlaces de comunicações entre eles. Cada aresta no grafo é representada por um par ordenado (v_i, v_j), onde v_i e v_j são elementos distintos de $\mathcal{V}$, i.e., $v_i \neq v_j$. Em cada aresta (v_i, v_j) é atribuída uma probabilidade como é mostrado na Figura 5.7. O peso na aresta indica a

probabilidade de uma anomalia observada em v_i se propagar para os dispositivos dependentes ou vizinhos de v_i. A vizinhança de v_i é denotada por $N(v_i)$. O cálculo das probabilidades será apresentado na seção 5.4.4

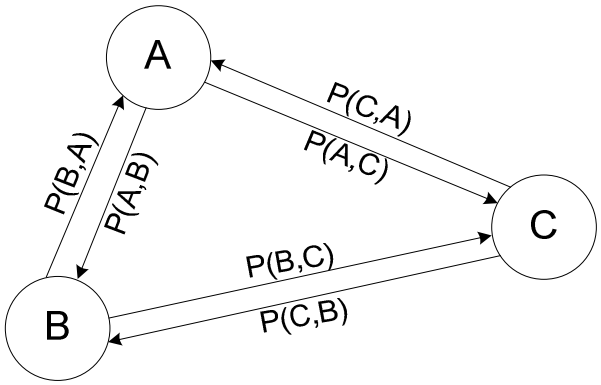

Figura 5.7 – Grafo de dependência com probabilidades nas arestas.

É definido $X = \{x_0, x_1, \dots, x_n\}$, $X \subseteq \mathcal{V}$, o conjunto de dispositivos de redes afetados por um evento anômalo em uma determinada janela de tempo. O número de total de dispositivos afetados pela anomalia é dado por $n_x = \#X$. Um conjunto $\lambda = \{\lambda_1, \lambda_2, \dots, \lambda_n\}$ é definido como um subconjunto de $\mathcal{E}$, onde λ_k para $k = \{1, \dots, n\}$ representam os enlaces que conectam os elementos de X. No cenário de propagação da anomalia pela rede, λ_k representa o enlace anômalo entre x_i e x_j.

5.4.2 Matriz de dependência

Para a modelagem do problema, a estrutura do grafo $\mathcal{G}$ é representada por uma matriz de dependência, denominada de matriz $\mathcal{D}$. Essa matriz irá representar os elementos X afetados pela anomalia, sendo definida em (5.10).

$$\mathcal{D} = (d_{i,j})_{n_{(x+1)} \times n_{(x+1)}} \tag{5.10}$$

As entradas da matriz $\mathcal{D}$ são definidas da seguinte forma. Cada $d_{i,j}$ pode assumir um valor representado pela dupla $< w_{ij}, p(x_i, x_j) >$, onde

$$w_{ij} = \begin{cases} \mu & \text{se } (x_i, x_j) \in \lambda, \\ 0 & \text{caso contrário.} \end{cases} \tag{5.11}$$

e $p(x_i, x_j)$ é a probabilidade atribuída a cada enlace anômalo λ_k que conecta x_i e x_j. $\mu \neq 0$ indica que o tráfego anômalo está saindo de x_i para x_j.

As informações do grau de entrada e o grau de saída dos elementos são utilizados no processo de inferência da fonte e destino da anomalia. Para armazenar o grau de entrada e saída de cada nó, uma linha e uma coluna são adicionadas na matriz $\mathcal{D}$.

O grau de saída de x_k pode ser obtido por (5.12). É realizada a soma das entradas na linha k da matriz $\mathcal{D}$, onde $w_{kj} \neq 0$ $(0 \leq j \leq n_x)$.

$$z_x^{out} = \sum_{\forall 0 \leq j \leq n_x \mid w_{kj} \neq 0} 1 \tag{5.12}$$

De modo similar, a soma das entradas da coluna k da matriz $\mathcal{D}$ fornece o grau de entrada de x_k, com $w_{ik} \neq 0$ $(0 \leq i \leq n_x)$, como é mostrado em (5.13).

$$z_x^{in} = \sum_{\forall 0 \leq i \leq n_x \mid w_{ik} \neq 0} 1 \tag{5.13}$$

Um nó x_k é considerado uma fonte quando seu grau de entrada é zero, isto é, $z_x^{in} = 0$ e seu grau de saída $z_x^{out} \geq 1$. Essas informações indicam que x_k não têm predecessores (nós anteriores). Por outro lado, x_k é considerado destino quando $z_x^{out} = 0$ e $z_x^{in} \geq 1$. Em (5.14), a linha $n_{(x+1)}$ chamada S e a coluna $n_{(x+1)}$ F representam o grau de entrada e saída de cada x_k, respectivamente.

$$\mathcal{D} = \begin{array}{c} \\ \\ \\ \\ S \rightarrow \end{array} \begin{array}{c} \begin{array}{cccc} & & & \quad F \downarrow \end{array} \\ \left[\begin{array}{cccc|c} d_{00} & d_{01} & \cdots & d_{0n_x} & z_{x_0}^{out} \\ d_{10} & d_{11} & \cdots & d_{1n_x} & z_{x_1}^{out} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ d_{n_x0} & d_{n_x1} & \cdots & d_{n_xn_x} & z_{x_{n_x}}^{out} \\ \hline z_{x_0}^{in} & z_{x_1}^{in} & \cdots & z_{x_{n_x}}^{in} & \end{array}\right] \end{array} \tag{5.14}$$

5.4.3 Medida de incerteza

A complexidade da correlação de alarmes está ligada a incerteza, ambiguidade e incompletude, devido a alarmes com informações de baixa qualidade e incompletos [34]. Isso implica que para um mesmo conjunto de alarmes pode haver mais de uma explicação plausível para a causa do problema. Esse cenário se torna mais complexo quando é levada em consideração a existência de alarmes espúrios [46] [78], responsáveis pela incompletude de dados.

Embora conscientes de que a incerteza gerada por alarmes espúrios pode inviabilizar a utilização das soluções voltadas para a localização da fonte do problema, muitos autores têm evitado o desenvolvimento de soluções que estejam aptas a tratá-los, dado o aumento da complexidade da análise [26]. Em um ambiente de rede real, alarmes espúrios são inevitáveis [78]. Sendo assim, a confiabilidade e a precisão da correlação dependem fortemente do tratamento dos alarmes espúrios, pois caso sejam negligenciados, a qualidade da correlação pode ser substancialmente degradada.

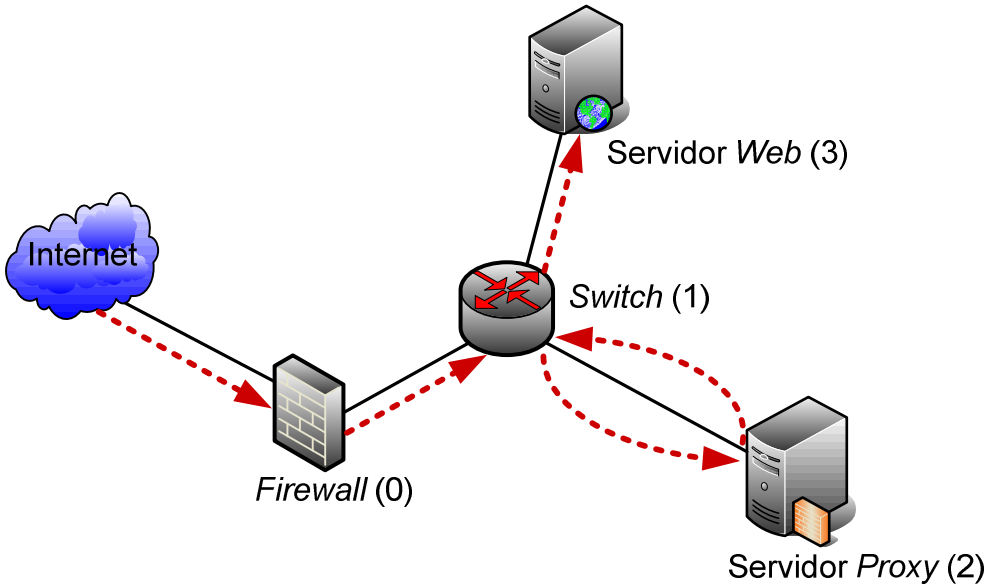

Figura 5.8 – Rede afetada por uma atividade anômala originada na Internet que tinha como destino o servidor *web*.

A fim de mostrar o impacto de alarmes espúrios no processo de inferência da fonte e o destino da anomalia, consideramos um cenário de rede ilustrado na Figura 5.8, no qual há a

ocorrência de uma anomalia. O tráfego anômalo, nesse exemplo, afetou os equipamentos *firewall*, *switch* e servidor *proxy* que compõem o caminho entre um ponto na Internet e o servidor *web*. A Figura 5.9 (a) apresenta um caso onde um alarme não foi gerado (falso-negativo), deixando de indicar a volta do tráfego anômalo do servidor *proxy* (2) para o *switch* (1). Analisando os alarmes, temos dois prováveis caminhos anômalos. O primeiro P_1, de 0 a 3 e o P_2 de 0 a 2. É importante destacar que, embora haja um único tráfego anômalo originado na Internet e destinado ao servidor *web*, dois possíveis caminhos são observados no processo de correlação, tendo como causa a não geração de um alarme. Nesse caso, a existência de mais de uma explicação para o mesmo conjunto de dados decorre do fato de que esse conjunto de alarmes possui alarmes espúrios.

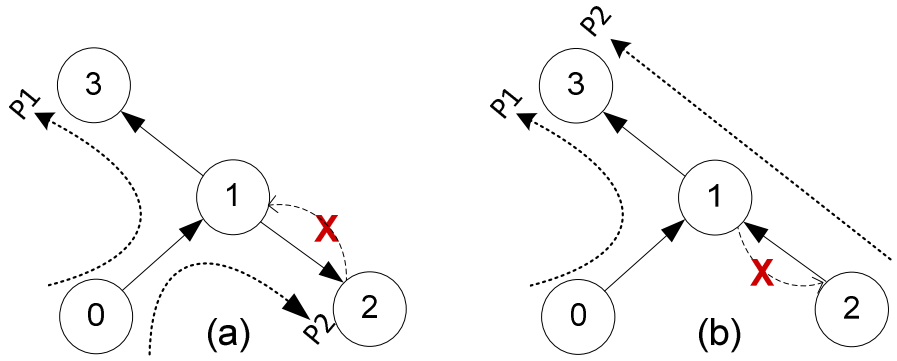

Figura 5.9 - (a) Um cenário com dois possíveis destinos e (b) um cenário com duas possíveis fontes causados por alarmes espúrios.

A Figura 5.9 (b) apresenta outro cenário em que alarmes não foram gerados, ignorando a ida do tráfego anômalo do *switch* (1) para o servidor *proxy* (2). Diferente da Figura 5.9 (a), na qual existem dois possíveis destinos, na Figura 5.9(b), a não geração de um único alarme resultou em duas possíveis fontes. Nesse caso, existem também dois possíveis caminhos, P_1 partindo de 0 a 3 e P_2 de 2 a 3. Os exemplos mostram que a ocorrência de alarmes espúrios no processo de correlação de alarmes pode aumentar a incerteza e a complexidade do processo de identificação da verdadeira fonte e destino do tráfego anômalo. Dessa forma, é razoável afirmar

que quando uma solução não leva em conta a possibilidade de existência de alarmes espúrios, sua aplicação na prática pode se tornar inviável.

Nesta dissertação, é apresentada uma medida baseada no conceito de entropia com o objetivo de minimizar as incertezas geradas no processo de identificação da origem e destino da anomalia, decorrentes da existência de alarmes espúrios no conjunto de dados. Medidas utilizando entropia têm sido amplamente aplicadas para resolução de problemas em diferentes áreas de conhecimento, tais como a física, teoria da informação, compressão de dados, detecção de anomalias e economia [69].

O conceito de entropia foi introduzido na termodinâmica pelo físico Rudolf Clausius, em 1865, como uma medida de ordem/caos. Em 1948, Claude E. Shannon apresentou o conceito de entropia como uma medida de incerteza. No contexto da teoria da informação, o conceito de entropia tem sido aplicado como uma de medida de quantificação de informação e incerteza [69].

Shannon definiu a entropia da seguinte forma. Seja X uma variável aleatória discreta, com probabilidades $p_1, p_2, \ldots, p_n$. A entropia é matematicamente expressada como

$$\boldsymbol{H}(p_1, p_2, \ldots, p_n) = \boldsymbol{H}(\mathrm{p}) = -\sum_{i=1}^{n} p(x_i) \log p(x_i) = -\sum_{i=1}^{n} p_i \log p_i \qquad (5.15)$$

A Figura 5.10 mostra a relação entre os valores de probabilidades no intervalo $0 \leq p \leq 1$ e o valor de entropia de Shannon. O valor $p = 0{,}5$ resulta no valor de entropia $H(p) = 1$, sendo denominado de ponto máximo de incerteza [69]. Quando a probabilidade $p = 1$, temos que $H(p) = 0$, ou seja, não há incerteza sobre o evento.

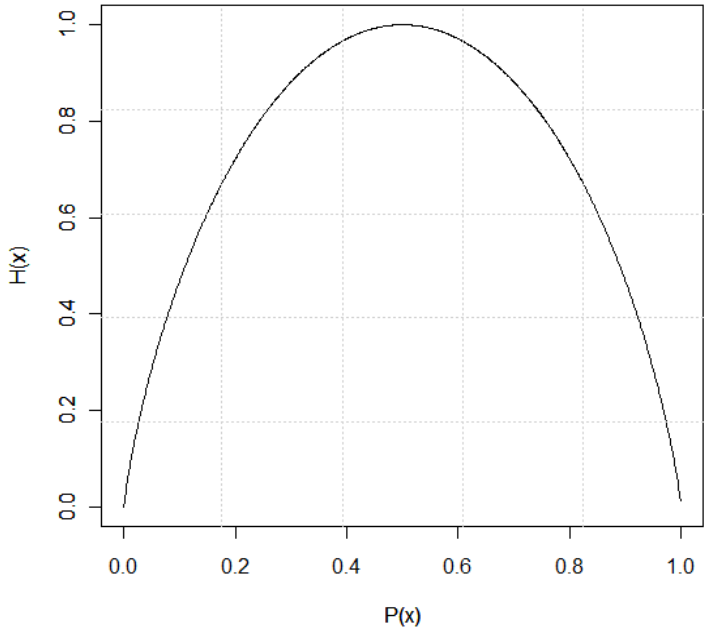

Figura 5.10 – Entropia de Shannon x probabilidade.

Propostas na área de detecção de anomalia, correlação de alarmes e tomografia de redes têm aplicado entropia como forma de mensurar incertezas de diversos aspectos. Porém, no contexto deste trabalho, a entropia de Shannon, também denominada de entropia clássica, não é suficiente para resolver o problema de incerteza gerada por alarmes espúrios. Busca-se uma medida que, a partir da utilização de probabilidades *a priori*, possa ser aplicada para inferir o caminho percorrido pela anomalia, em cenários onde existem possíveis caminhos gerados por alarmes espúrios.

A fim de mostrar que a entropia de Shannon não atende as necessidades do propósito deste trabalho é mostrado o seguinte exemplo. Tomamos o exemplo apresentado na Figura 5.9 (b). Suponhamos que as probabilidades de propagação dos enlaces, mostradas na Figura 5.11, sejam: $l_{(0,1)} = 0{,}71$, $l_{(2,1)} = 0{,}11$ e $l_{(1,3)} = 0{,}28$. Temos que o caminho P_1 possui os enlaces $l_{(0,1)}$ e $l_{(1,3)}$ e P_2 os enlaces $l_{(2,1)}$ e $l_{(1,3)}$. Utilizando o logaritmo na base dez temos que a entropia/incerteza de $P_1 \cong 0{,}26$ e $P_2 \cong 0{,}26$. Mesmo que P_1 possua o $l_{(0,1)} = 0{,}71$ com valor muito superior ao $l_{(2,1)} = 0{,}11$ de P_2, o valor de entropia é o mesmo. Isso pode ser visualizado

no gráfico ilustrado na Figura 5.10, onde a relação probabilidade *x* entropia mostra que os pequenos valores de probabilidade (próximos de zero) ou valores altos (próximos de um) pouco influenciam no valor da entropia. Essa característica da entropia de Shannon é considerada por Ziviani *et al.* [10] como uma limitação para sua aplicação de forma mais generalizada.

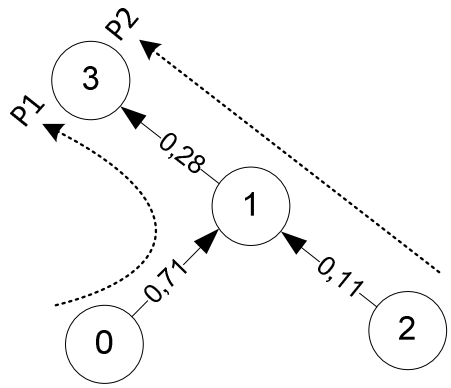

Figura 5.11 – Grafo ponderado com dois possíveis caminhos.

Em 1988, um físico brasileiro chamado Constantino Tsallis [16] propôs o conceito de entropia *não extensiva*, generalizando a entropia clássica ou a entropia de Shannon. A entropia de Tsallis é definida da seguinte forma.

Seja X uma variável aleatória discreta, com probabilidades $p_1, p_2, \ldots, p_N$, onde $0 \leq p \leq 1$, a entropia de Tsallis é formalmente expressada como

$$\boldsymbol{H}_q(X) = \frac{1}{q-1}\left(1 - \sum_{i=1}^{N} {p_i}^q\right) \tag{5.16}$$

onde o valor de $\boldsymbol{H}_q(X)$ varia de 0 ao valor máximo

$$\boldsymbol{H}(X)_q^{Max} = \frac{1 - N^{1-q}}{q-1} \tag{5.17}$$

Como podemos observar em (5.16), na entropia generalizada temos o parâmetro q denominado de parâmetro entrópico [10]. O parâmetro q indicará os valores de probabilidades

que terão maior influência no resultado da entropia, diferente da entropia clássica, na qual as probabilidades altas ou baixas exercem pouca influência no resultado. No caso da entropia generalizada, podem ser utilizados valores para q de modo que as probabilidades altas ou baixas tenham maior influência no valor final de entropia. Para $q > 1$ as probabilidades maiores têm uma contribuição maior para o valor da entropia do que as probabilidades de menores valores, ao passo que $q < 1$ temos o oposto [68]. Em [10], é demonstrado que quando $q \rightarrow 1$ temos que a entropia de Tsallis é equivalente a entropia de Shannon.

Neste trabalho, o conceito de entropia é interpretado como a incerteza/certeza associada a cada possível caminho de ser o mais provável percorrido pelo tráfego anômalo. Nesse contexto, o menor valor de entropia indica o caminho mais provável percorrido pela anomalia. Em termos de probabilidade temos que quando $p(x) \rightarrow 1$ a incerteza sobre a ocorrência do evento tende a zero ($H(x) \rightarrow 0$), isso pode ser observado na Figura 5.12. Utilizaremos o mesmo exemplo apresentado na Figura 5.11, porém aplicando a entropia de Tsallis. Considerando $q = 2$, temos que $P_1 \cong 0{,}42$ e $P_2 \cong 0{,}91$. Diferente do que ocorreu no exemplo com a aplicação da entropia de Shannon, este último exemplo mostra que as maiores probabilidades têm influência sobre o valor de entropia quando utilizamos a entropia de Tsallis.

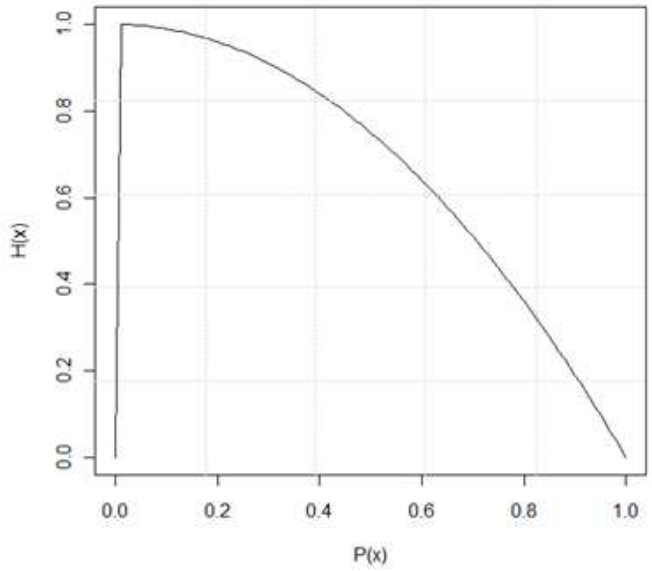

Figura 5.12 - Entropia de Tsallis x Probabilidade com $q = 2$.

5.4.4 Cálculo da probabilidade de propagação da anomalia nos enlaces

As probabilidades de propagação do tráfego anômalo nos enlaces foram estimadas a partir de dados históricos dos eventos anômalos ocorridos na rede.

Inicialmente foram separados os alarmes pertencentes ao grupo 1 (G1) e grupo 2 (G2) emitidos para o(s) dispositivo(s) afetado(s) pela anomalia, como definido na seção 5.3.1. Posteriormente, através do conhecimento das dependências entre os dispositivos, foi calculada a probabilidade da seguinte forma. Considere um dispositivo r conectado a um dispositivo w por meio de um enlace. A probabilidade de propagação da anomalia por esse enlace, denotada por $p_{r \to w}$, é dada por (5.18).

$$p_{r \to w} = \frac{\{a_1, a_2, \ldots, a_n\} \in \mathrm{G2}}{\{a_1, a_2, \ldots, a_m\} \in \mathrm{G1} + \{a_1, a_2, \ldots, a_n\} \in \mathrm{G2}} = \frac{\#\mathrm{G2}}{\#\mathrm{G1} + \#\mathrm{G2}} \tag{5.18}$$

5.4.5 Identificando a origem e destino do tráfego anômalo

Na seção 5.4.3, foi mostrado que os alarmes espúrios são prejudiciais para a correlação, pois podem trazer incertezas quanto a real fonte e destino da anomalia. Nesta seção, será apresentado um método que tem por objetivo a identificação do caminho de propagação do tráfego anômalo, sua fonte e seu destino, mesmo em cenários com alarmes espúrios.

Nós definimos $S = \{s_0, s_1, \ldots, s_m\}$, $m \in \mathbb{N}$ e $m < n_x$ o conjunto dos dispositivos candidatos para ser fonte da anomalia e o conjunto $F = \{\mathrm{f}_0, \mathrm{f}_1, \ldots, \mathrm{f}_n\}$, $n \in \mathbb{N}$ e $n < n_x$, com $F \cap S = \emptyset$, denotando os possíveis dispositivos para ser o destino da anomalia. Os elementos de S e F são dados pelo grau de entrada e saída obtidos de (5.14). Usaremos $\#S$ e $\#F$ para representar o número de elementos pertencentes aos conjuntos S e F respectivamente.

A presença de alarmes espúrios pode resultar nos possíveis cenários ilustrados na Figura 5.13. Um nó vermelho é considerado uma fonte e nó azul um destino. O caso (i) refere-se a situação onde que $\#S > 1$ e $\#F = 1$, ou seja, a ocorrência dos alarmes espúrios não gerou incerteza sobre o destino ($\boldsymbol{E}$) do tráfego anômalo; no entanto, não há certeza sobre a origem. Em (ii), temos $\#S = 1$ e $\#F > 1$. Esse cenário mostra que não há alarmes espúrios afetando a identificação da fonte ($\boldsymbol{A}$) do tráfego anômalo, mas o destino é influenciado. O último caso,

refere-se ao pior dos cenários, onde temos $\#S > 1$ e $\#F > 1$. Ele mostra que os efeitos dos alarmes espúrios culminaram em diversas hipóteses sobre a origem e o destino da anomalia e, consequentemente, possíveis caminhos de propagação.

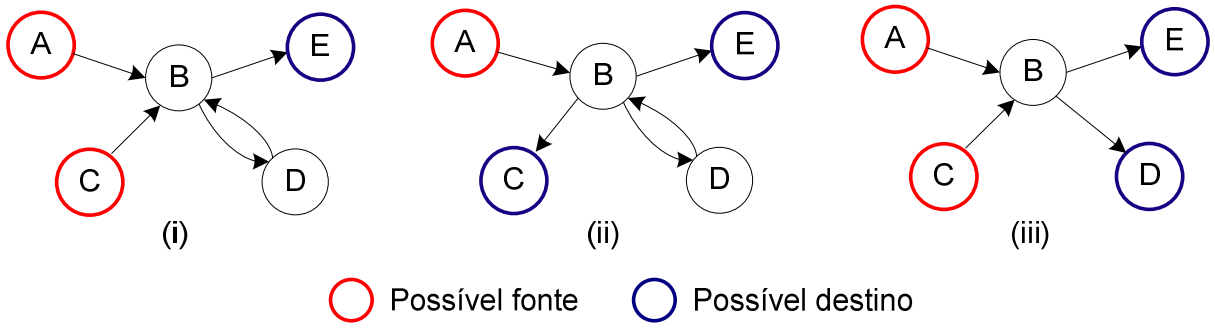

Figura 5.13 – Possíveis cenários causados por alarmes espúrios.

Nós denotamos por P_s um caminho entre um elemento de S a F, sendo P o conjunto de todos os possíveis caminhos, i.e., $P = \cup_{\forall s} P_s$. Cada P_s é descrito por uma 3-upla $< s_q, \mathrm{f}_t, \boldsymbol{\gamma} >$ onde $s_q \in S$ é um dispositivo candidato a ser a fonte, $\mathrm{f}_t \in F$ o dispositivo candidato a ser o destino do tráfego anômalo e $\boldsymbol{\gamma}$, um vetor contendo as probabilidades atribuídas aos enlaces anômalos λ_k pertencentes ao caminho que conecta s_q e f_t. Supondo que haja n enlaces anômalos em um caminho P_s, $\boldsymbol{\gamma}$ teria os valores $\gamma_1, \gamma_2, \ldots, \gamma_n$.

Para obter o caminho mais provável percorrido pela anomalia, é verificado cada caminho entre um candidato s_q e um candidato f_t. Para isso, o processo é iniciado tomando um $\mathrm{s}_q = x_i$ e verificando sua vizinhança. Se $N(x_i) > 1$, o nó sucessor de x_i será o vizinho x_{i+u} conectado por um λ_k com o maior peso, como é representado em (5.19).

$$p_{max} = \max\{p(\lambda_1), p(\lambda_2), \ldots, p(\lambda_n)\} \tag{5.19}$$

A probabilidade do enlace λ_k escolhido pelo critério p_{max} de (5.19), é adicionada ao vetor $\boldsymbol{\gamma}$. Esse processo irá continuar até que f_t seja alcançado.

Pelo critério p_{max} temos que para cada s_q até f_t há apenas um caminho. Assim, podemos dizer que número total de elementos de P, é m, onde $m = \#S$. Quando $m > 1$, significa que existem m caminhos. Nesse ponto, é necessária uma medida quantitativa que dê suporte para

julgar qual é o caminho mais provável percorrido pela anomalia. Para esse fim, usamos uma medida baseada na entropia de Tsallis, apresentada na seção 5.4.3.

Neste trabalho, o conceito de entropia é interpretado como a incerteza associada a cada $P_s \in P$ de ser o caminho percorrido pela anomalia. O menor valor de entropia calculado para todos os possíveis P_s é considerado, em termos de probabilidade, o caminho percorrido pela anomalia.

Consideramos o vetor $\boldsymbol{\gamma}$ contendo as probabilidades atribuídas a cada enlace de P_s. Assim, podemos escrever que a incerteza desse caminho ser o mais provável como

$$\boldsymbol{H}_q(P_s) = \frac{1}{q-1}\left(1 - \sum_{i=1}^{N}\left(\frac{\gamma_i}{N}\right)^q\right) \tag{5.20}$$

A busca pelo caminho da anomalia, bem com sua fonte destino, é realizada em três passos. O primeiro consiste no cálculo de $\boldsymbol{H}(P_s)$ para cada $P_s \in P$. O segundo passo é de obter o **MPP** (*Most Probable Path* - Caminho Mais Provável) pelo critério

$$\mathbf{MPP} = \min\{\boldsymbol{H}(P_0), \boldsymbol{H}(P_1), \ldots, \boldsymbol{H}(P_m)\} \tag{5.21}$$

Finalmente, no último passo, é obtida a fonte s_q e o destino f_t mais provável de **MPP**. Lembre-se de que $\mathbf{MPP} = \boldsymbol{H}(P_s)$, onde P_s é constituído por $< s_q, f_t, \boldsymbol{\gamma} >$.

O algoritmo para identificar a fonte e o destino da anomalia é apresentado na Tabela 5.1. Vale salientar que o objetivo do algoritmo é uma instância do problema geral da inferência abdutiva, onde através dos efeitos (alarmes) busca-se identificar a origem e destino do tráfego anômalo. A inferência abdutiva tem sido a forma de descobrir as causas dos incidentes observados na rede através da correlação de alarmes. Esse processo é comumente realizado em dois passos. Primeiramente é modelada a relação causa-efeito dos alarmes, em seguida, heuristicamente, busca-se identificar o incidente responsável pela geração dos alarmes observados [32].

Tabela 5.1 **-** Algoritmo para traçar o caminho percorrido pela anomalia.

Algoritmo: Identificação do caminho de propagação do tráfego anômalo	
Entrada: Grafo *G(V,E)*	
Saída: Mapa de Propagação da Anomalia	
$\boldsymbol{\gamma}$	vetor com os pesos dos enlaces
$P_s < inicio, \boldsymbol{\gamma}, fim >$	caminho entre um elemento de S e F representado por uma tripla
P	lista de caminhos
x_i	dispositivo afetado pela anomalia
x_{i+u}	dispositivo sucessor no caminho de propagação
N	lista dos sucessores de x_i

```
1:  for all s ∈ S  do
2:    x_i := s
3:    while(true)  do
4:       N := getSucessores(x_i)
5:       if N.isEmpty() then
6:          break
7:       else if N.size() == 1 then
8:          γ := N.get(0).getPesoEnlace()
9:          x_{i+u} := N.get(0).getNrDispositivo()
10:         x_i := x_{i+u}
11:      else
12:        Obter x_{i+u} baseado no λ_k = (x_i, x_{i+u}) utilizando o critério p_max
13:           p_max = max{p(x_i, x_{i+1}), p(x_i, x_{i+2}), ..., p(x_i, x_n)}
14:           γ := p_max
15:           x_i := x_{i+u}
16:      end if
17:   end while
18:   P_s := < s, x_{i+u}, γ >
19:   P.add(P_s)
20: end for
21:
22: calcularEntropiaCaminhos(P)
23:
```

5.5 Camada de apresentação

Essa camada é responsável pela apresentação do resultado do processo de correlação. Como fase final do processo de correlação, são implementados nessa camada meios para prover ao administrador de rede a visualização do cenário de rede afetado pela anomalia.

A prática comum para minimizar a carga de trabalho dos administradores de rede é reduzir a quantidade de alarmes. Porém, isso não é suficiente, pois o volume de alarmes, ainda

que reduzido, pode ser grande para ser manipulado pelo administrador a fim de identificar e aplicar ações corretivas para o problema.

Portanto, um sistema de correlação de alarmes deve facilitar a tarefa de compreensão das informações geradas pelo processo de correlação. Sob a perspectiva de um administrador humano, a visualização gráfica é amplamente aceita como uma estratégia plausível e intuitiva para entendimento do problema. Nesse caso, o administrador interagiria diretamente com a visualização de toda rede e o comportamento dos dispositivos na ocorrência de um evento anômalo, em vez de aplicar esforços para a investigação dos alarmes, que tradicionalmente são no formato de texto [65].

A ferramenta APV (*Anomaly Propagation View*) foi desenvolvida com o objetivo de automatizar o processo de correlação de alarmes e prover ao administrador uma visão global do estado da rede. A ferramenta apresenta graficamente o resultado da correlação de alarmes, possibilitando a visualização da propagação e os elementos da rede afetados pela anomalia. Um detalhamento da ferramenta e suas características serão apresentados na seção 6.3.

Capítulo 6

Implementação e resultados

6.1 Infraestrutura de monitoramento

Como infraestrutura para caracterização de tráfego (DSNS) e geração de alarmes, foi utilizada a ferramenta GBA (Gerenciamento de *Backbone* Automático). Essa ferramenta, que atualmente encontra-se na versão 6.0, foi inicialmente proposta por Proença Junior [43], e tem como principal objetivo auxiliar o gerenciamento de rede. A Figura 6.1 mostra uma visão global da ferramenta com os seus principais módulos.

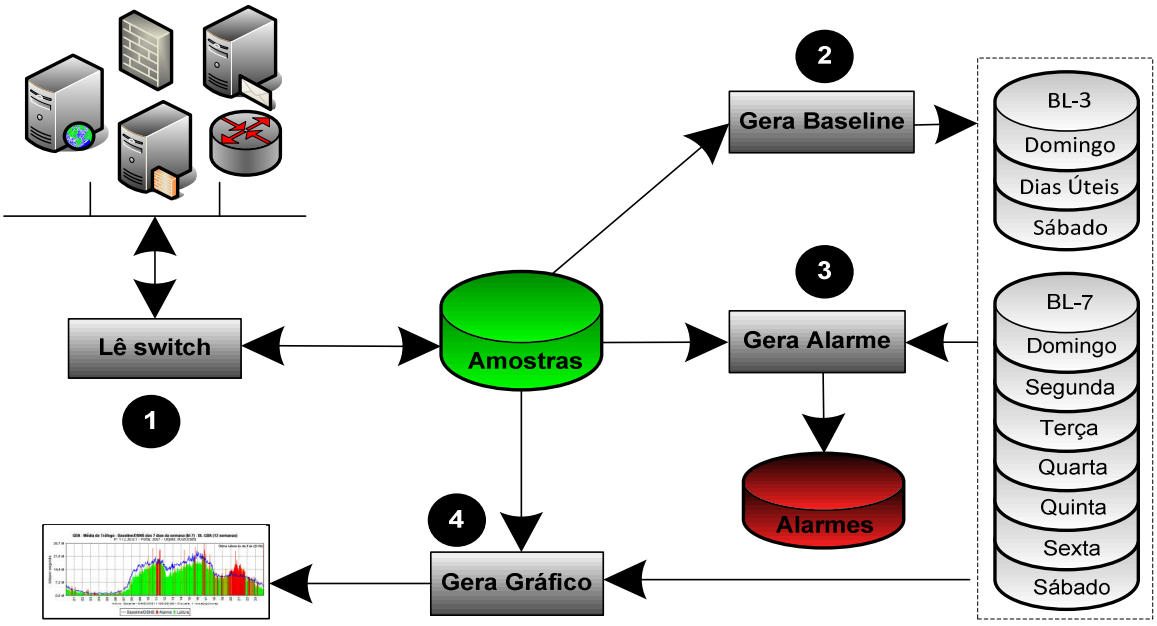

Figura 6.1 – Módulos da ferramenta GBA.

Cada módulo é descrito a seguir:

- ✓ **Módulo Lê switch**: responsável pela coleta em tempo real de dados das MIBs dos dispositivos de rede, tais como *switches* e roteadores. Dados coletados de cada objeto SNMP são armazenados em um repositório em arquivos com extensão "*.les"*. Esse módulo foi desenvolvido com a linguagem C++ e utiliza a biblioteca NET-SNMP [48];
- ✓ **Módulo Gera baseline**: responsável pela geração dos DSNS/*baseline*, utilizando dados históricos coletados pelo Le switch. Ele gera *baselines* denominados de BL-3 (um *baseline* para o domingo, um para o sábado e outro para os dias úteis) e BL-7 (um *baseline* para cada dia da semana), que são armazenados com a extensão "*.bln*". Esse módulo também é um executável escrito com a linguagem C++;
- ✓ **Módulo Gera alarme**: responsável pela emissão de alarmes. Utiliza como entrada dados dos arquivos de leitura "*.les*" coletadas pelo Lê switch e dados do arquivos "*.bln*" gerados pelo Gera *baseline*. Esse módulo foi desenvolvido com a linguagem Java e utiliza a tecnologia EJB (*Enterprise Java Beans*) versão três [49]. Tem como função analisar as amostras coletadas e gerar notificações ao administrador da rede na ocorrência de um evento anômalo. Para obter as amostras do arquivo "*.les*" é invocado outro EJB chamado EJB Leitura. Da mesma forma, o EJB Baseline é invocado para obter os dados do arquivo "*.bln*". Alarmes gerados são armazenados em arquivos com a extensão "*txt*";
- ✓ **Módulo Gera gráfico**: responsável pela geração de gráficos do tráfego coletado e o *baseline* para cada dispositivo. Através de parâmetros, é possível gerar diferentes tipos de gráficos com intervalos por segundo, minuto, hora, dia, etc. O tipo de gráfico "apenas leitura" produz um

gráfico dos dados armazenados no arquivo "*.les*". O tipo "apenas baseline" toma como entrada valores do arquivo "*.bln*" e apresenta o gráfico do *baseline*. A opção de gerar gráficos que apresentem a leitura real juntamente com os valores esperados pelo *baseline* também é possível. Nesse caso, são lidos ambos os arquivos. Similarmente ao módulo Gera alarme, o módulo Gera gráfico é escrito com a linguagem Java e utiliza a tecnologia EJB3. Os gráficos são exportados para o formato PNG (*Portable Network Graphic*).

6.2 Implementação na camada de pré-processamento

Essa seção irá descrever a implementação do módulo de pré-processamento do sistema de correlação de alarmes. A Figura 6.2 ilustra o modo de interação entre o sistema de correlação e a ferramenta GBA. A ferramenta GBA faz o monitoramento em tempo real da rede e ao detectar uma anomalia emite alarmes por meio do sistema de alarme. O sistema de correlação através de um módulo que implementa a camada de pré-processamento, obtêm os alarmes e busca inferir o caminho de propagação, a origem e o destino da anomalia, apresentando ao administrador o efeito da anomalia na rede de forma gráfica. A correlação e a apresentação gráfica ao administrador de rede são realizadas pela ferramenta APV, que será detalhada na seção 6.3.

O módulo de pré-processamento do sistema de correlação de alarmes foi desenvolvido utilizando a linguagem Java e a tecnologia EJB [49]. A Figura 6.3 apresenta através de um diagrama de componentes o módulo EJB Compressão, destacado em amarelo, e a sua forma de interação com os módulos já existentes do GBA.

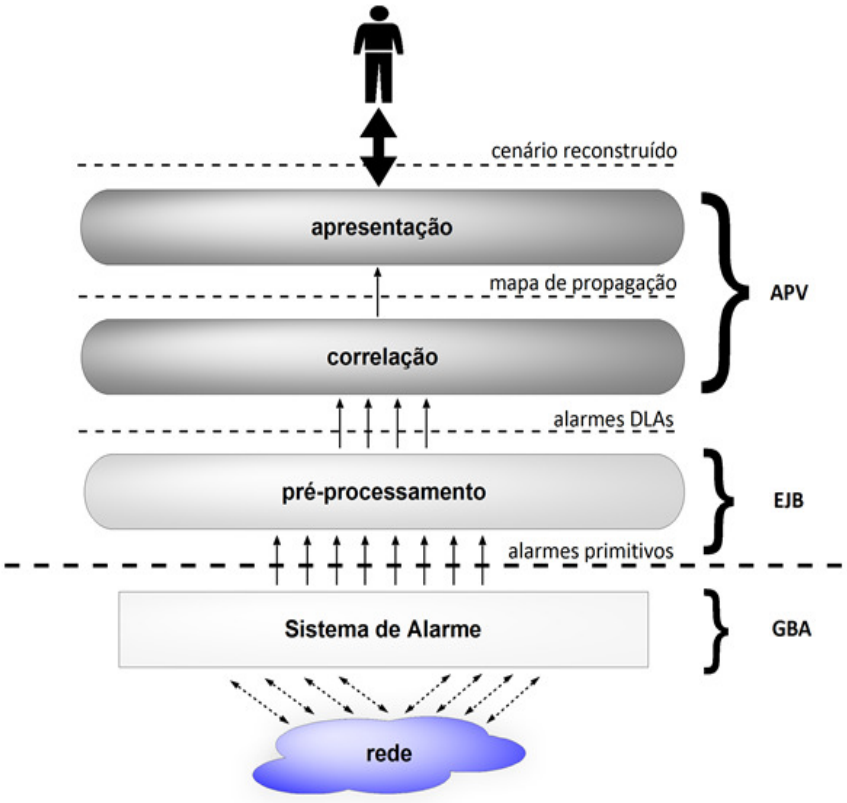

Figura 6.2 – Implementação das camadas do sistema de correlação e interação com a ferramenta GBA.

O módulo EJB Compressão é disparado por um executável Java. Depois de invocado o EJB Compressão, ele acessa o EJB Gera Alarmes. O EJB Gera Alarmes busca, através das leituras e valores DSNS, identificar as anomalias. Quando uma anomalia é encontrada, alarmes são gerados e armazenados em arquivos de alarmes. Posteriormente, o EJB Compressão acessa esse arquivo de alarmes e aplica o procedimento de redução do volume de alarmes descrito na seção 5.3.2, salvando os alarmes em arquivos chamados DLAs.

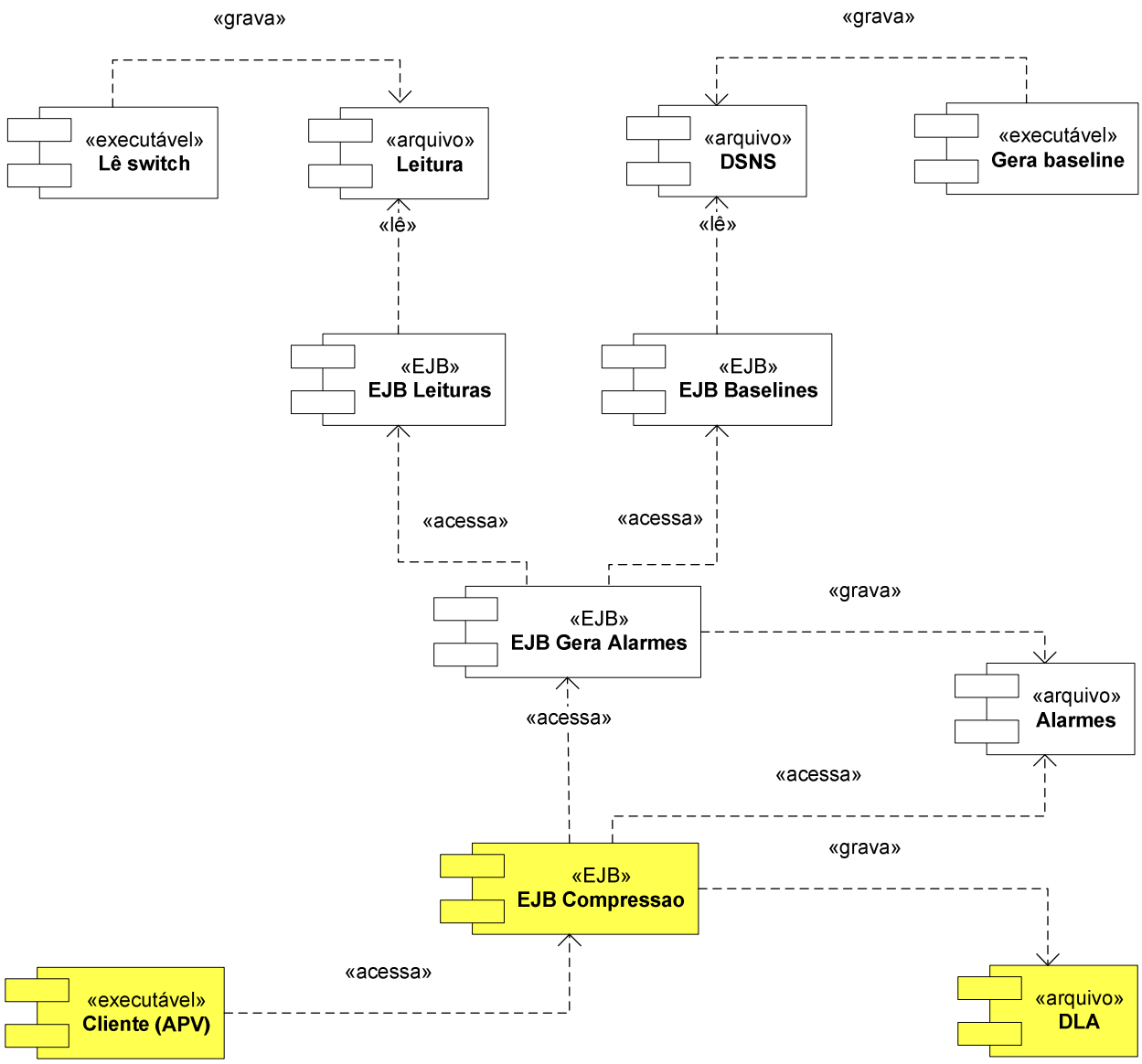

Figura 6.3 - Diagrama de componentes da ferramenta GBA com os novos módulos.

6.3 Ferramenta APV

Nesta seção, é apresentada a ferramenta APV (*Anomaly Propagation View*), a qual implementa as camadas de correlação e apresentação. A APV é uma ferramenta multiplataforma desenvolvida com linguagem Java. O objetivo principal da ferramenta é de reconstruir o cenário de rede afetado pela anomalia através de um conjunto de alarmes. Inicialmente, é feita a

correlação dos alarmes e posteriormente é apresentada a visão global do problema ocorrido na rede.

Foi utilizada a API (*Application Programming Interface*) Java Universal Network/Graph (JUNG). Essa é uma API *open-source* para linguagem Java, desenvolvida para manipular, analisar e visualizar dados que podem ser representados por um grafo. Maiores informações, *downloads* e documentação podem ser encontrados em [36].

A Figura 6.4 apresenta a ferramenta. O item (1) aponta para a barra de ferramentas. Ela é constituída de botões úteis para o tratamento de questões como, importar e abrir arquivos de alarmes, copiar e salvar o mapa de propagação, entre outros. O item (2) aponta para a área de *plotagem*, que mostra o resultado final do processo de correlação de alarmes. A rede é apresentada de maneira holística, exibindo todos os dispositivos e enlaces afetados pelo evento anômalo. Como podemos ver no item (2), é possível contrastar os dispositivos e os enlaces em vermelho com os demais não afetados pela anomalia. Os enlaces em vermelho referem-se ao caminho de propagação do tráfego anômalo. O item (3) aponta para uma tabela que contém informações mais detalhadas do resultado do processo de correlação. Essa tabela apresenta a data em que a anomalia foi detectada, primeira e última ocorrência, informações sobre a fonte e o destino mais prováveis da anomalia e do número de dispositivos e enlaces afetados pela atividade anômala.

A ferramenta possui características e facilidades relevantes. Primeiramente, ela é portável para os ambientes Windows, UNIX (Linux), Solaris e Mac OS. Além disso, a ferramenta disponibiliza recursos como o *zoom in*/*out* e a rotação do mapa de propagação, possibilitando o ajuste para uma melhor visualização por parte do administrador de rede.

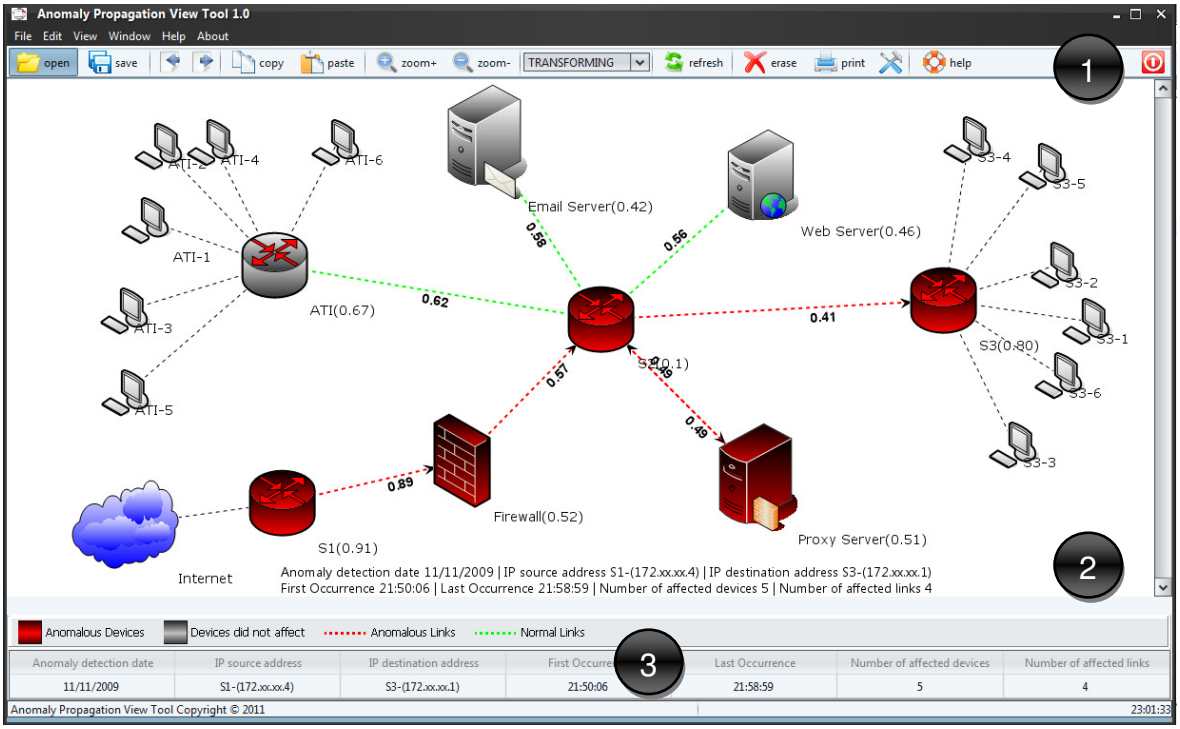

Figura 6.4 – Ferramenta APV. (1) barra de Ferramenta, (2) área de plotagem, (3) tabela com informações detalhadas sobre o evento anômalo.

6.4 Ambiente de rede monitorado

A fim de avaliar o sistema de correlação proposto, foram utilizados dados reais obtidos da rede da Universidade Estadual de Londrina. A topologia com os elementos monitorados é apresentada na Figura 6.5.

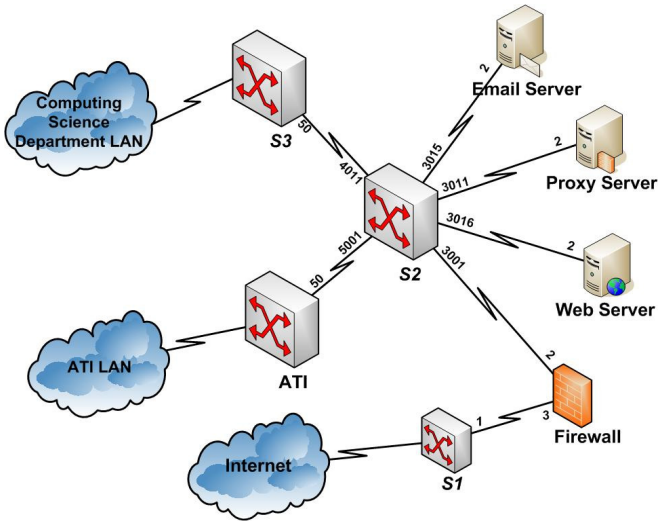

Figura 6.5 - Rede monitorada da Universidade Estadual de Londrina.

A Tabela 6.1 apresenta informações dos equipamentos monitorados, bem como uma descrição sucinta de cada um deles.

Tabela 6.1 – Informações dos equipamentos monitorados.

Rótulo	*Equipamento*	*Descrição*
x_0	*Switch* ***S1***	*Switch* intermediário entre os enlaces de Internet e o *firewall*.
x_1	*Firewall*	Responsável por filtrar todo tráfego direcionado da rede da Internet para a rede da universidade e vice-versa.
x_2	*Switch* ***S2***	Responsável por conectar os principais elementos do *backbone* da rede da UEL.
x_3	*Switch* ATI	Responsável por interconectar a sub-rede da Assessoria de Tecnologia de Informação (ATI) ao núcleo da rede.
x_4	*Switch* ***S3***	Responsável por interconectar a sub-rede do Departamento de Computação (DC) ao núcleo da rede.
x_5	Servidor de Email	Responsável por prover serviço de email para docentes, funcionários e alunos.
x_6	Servidor *Proxy*	Responsável por controlar o acesso a conteúdo da Internet.
x_7	Servidor *Web*	Principal servidor de páginas da universidade.

6.5 Resultados

Cinco estudos de casos obtidos em diferentes períodos serão apresentados nesta dissertação. Os quatro primeiros estudos de caso são referentes a eventos anômalos detectados ao longo dos anos de 2009 e 2010. Os estudos de caso um e dois apresentam um cenário ideal, onde não foram identificados alarmes espúrios influenciando na identificação do caminho de propagação da anomalia. Os estudos de caso três e quatro mostrarão um cenário com alarmes espúrios e como a medida baseada na entropia generalizada foi utilizada para inferir a origem e destino do tráfego anômalo. O estudo de caso cinco, realizado em maio de 2011, mostra um evento anômalo gerado por uma injeção artificial de tráfego na rede.

6.5.1 Estudo de caso 1

Nesse primeiro estudo de caso, é apresentado um evento anômalo detectado em 11/11/2009 e iniciado às 21h50min06seg com a última ocorrência às 21h58min59seg. O tráfego anômalo afetou o *switch* ***S1***, o *Firewall*, o servidor *Proxy*, o *switch* ***S2*** e o *switch* ***S3***. A seguir, os fragmentos contendo a primeira ocorrência de cada alarme disparado para cada objeto SNMP do dispositivo afetado pela anomalia serão apresentados.

A Figura 6.6 apresenta a primeira ocorrência de um alarme gerado às 21h50min06seg para o *switch* ***S1***. Esse equipamento é responsável por conectar a rede da universidade à Internet. Alarmes gerados para o objeto *ifOutOctets* na porta 1 de ***S1***, indicam que o tráfego anômalo está sendo gerado na Internet.

```
          Device: S1 - IP Address: 172.xx.xx.4
...
11Nov2009;21:50:06;SNMP_OBJECT:ifOutOctets;port:1
...
```

Figura 6.6 – Estudo de caso 1: Alarmes emitidos para *S1.*

A Figura 6.7 apresenta a primeira ocorrência do alarme gerado para o equipamento *Firewall* para o objeto *ipInReceives,* às 21h55min26seg.

```
Device: Firewall - IP Address: 189.xx.xx.2
...
11Nov2009;21:55:26;SNMP_OBJECT:ipInReceives;
...
```

Figura 6.7 – Estudo de caso 1: Alarmes emitidos para o *Firewall.*

Para o *switch* ***S2***, foram disparados alarmes para o objeto *ifInOctets* na porta 3001, como mostra a Figura 6.8. Isso indica que o tráfego anômalo veio do *Firewall.*

```
Device: S2- IP Address: 172.xx.xx.1
...
11Nov2009;21:54:49;SNMP_OBJECT:ifInOctets;port:3001
...
```

Figura 6.8 – Estudo de caso 1: Alarmes emitidos para *S2.*

Alarmes foram emitidos para os objetos *ipInReceives*, *tcpInSegs* e *tcpOutSegs* do servidor *Proxy* mostrado na Figura 6.9.

```
Device: Servidor Proxy - IP Address: 172.xx.xx.11
...
11Nov2009;21:53:34;SNMP_OBJECT:ipInReceives;
...
11Nov2009;21:51:34;SNMP_OBJECT:tcpInSegs;
...
11Nov2009;21:52:44;SNMP_OBJECT:tcpOutSegs;
...
```

Figura 6.9 – Estudo de caso 1: Alarmes emitidos para o servidor *Proxy.*

Na Figura 6.10 é apresentada a primeira ocorrência dos alarmes disparado para ***S2***, na porta 4011, para o objeto *ifOutOctets*, às 21h54min32seg.

```
Device: S2 - IP Address: 172.xx.xx.1
...
11Nov2009;21:54:32;SNMP_OBJECT:ifOutOctets;port:4011
...
```

Figura 6.10 – Estudo de caso 1: Alarmes emitidos para *S2.*

Os DLAs desse estudo de caso, gerados pelo sistema de correlação, são mostrados na Figura 6.11.

```
@G2
S1, Firewall
@G2
Firewall, S2
@G1-G2
S2, Proxy
@G2
S2, S3
```

Figura 6.11 – DLAs do estudo de caso 1.

A fim de construir a matriz D, os equipamentos envolvidos no evento anômalo serão rotulados como mostrado na Tabela 6.1. Assim, teremos ***S1*** como x_0, *Firewall* como x_1, ***S2*** como x_2, ***S3*** como x_4 e o servidor *Proxy* como x_6. Para esse estudo de caso temos a correspondente matriz $\mathcal{D}$.

$$\mathcal{D} = \begin{array}{c} \begin{matrix} x_0 & x_1 & x_2 & x_4 & x_6 & \end{matrix} \\ \begin{bmatrix} 0 & 1, p_{0,1} & 0 & 0 & 0 & 1 \\ 0 & 0 & 1, p_{1,2} & 0 & 0 & 1 \\ 0 & 0 & 0 & 1, p_{2,4} & 1, p_{2,6} & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1, p_{6,2} & 0 & 0 & 1 \\ 0 & 1 & 2 & 1 & 1 & \end{bmatrix} \end{array} \begin{matrix} x_0 \\ x_1 \\ x_2 \\ x_4 \\ x_6 \end{matrix} \tag{6.1}$$

Nesse cenário anômalo temos, $S = \{x_0\}$. x_0 destacado com o círculo vermelho em (6.1) é a possível fonte, pois o seu grau de entrada é 0 e o grau de saída é ≥ 1. Temos $F = \{x_4\}$, no qual x_4 é o possível destino do tráfego anômalo, destacado com o triângulo azul em (6.1), pois seu grau de saída é 0 e o grau de entrada é ≥ 1. O sistema de correlação, utilizando o algoritmo descrito na Tabela 5.1, infere que a fonte é x_0 (***S1***) e o destino x_4 (***S3***).

A visão global desse estudo de caso é apresentada na Figura 6.12, gerada pela ferramenta APV. Os equipamentos afetados pela anomalia, bem como os enlaces que compõem o caminho de propagação da anomalia são destacados em vermelho. Essa figura mostra que a informação

fornecida ao administrador de rede possibilita um rápido entendimento do impacto da anomalia na rede.

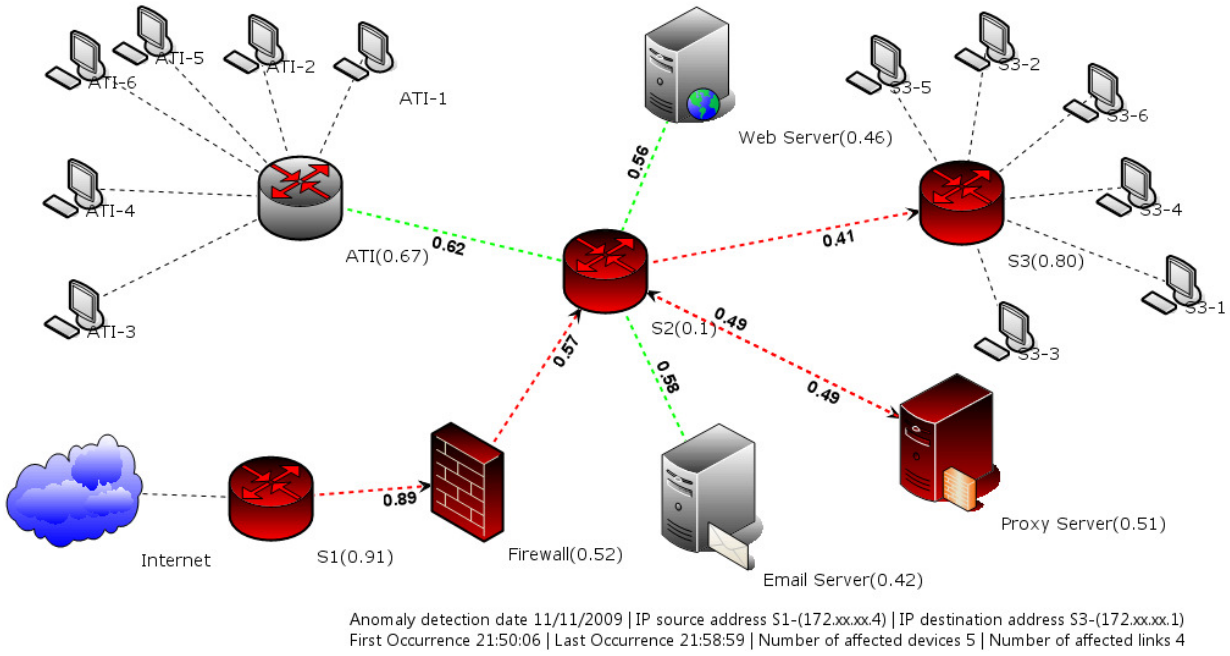

Figura 6.12 - Visão global da rede afetado pela anomalia do estudo de caso 1.

Esse estudo de caso demonstrou o processo realizado pelo sistema de correlação desde o recebimento dos alarmes primitivos gerados pelo sistema de alarmes até a apresentação do cenário global mostrando a propagação da anomalia na rede. A taxa de redução dos alarmes (TRA) obtida pela operação de compressão realizada na camada de pré-processamento foi de 71.4%. Isso implica que para esse estudo de caso, apenas 28,6% do total de alarmes gerados pela ocorrência da anomalia foram necessários para inferir a fonte e o destino do tráfego anômalo. É importante salientar que todo procedimento descrito no estudo de caso é feito de forma automatizada pelo sistema de correlação, sem a necessidade de intervenção do administrador de rede.

6.5.2 Estudo de caso 2

Esse segundo estudo de caso apresenta um evento anômalo detectado em 01/01/2010, com intervalo de duração entre 03h25min24seg e 03h38min34seg. A propagação da anomalia afetou os equipamentos ***S1***, servidor *Proxy*, ***S2*** e servidor *Web*.

Para ***S1***, o sistema de alarme disparou alarmes para o objeto *ifOutOctets*, com a primeira ocorrência em 03:25:24. O fragmento dos alarmes emitidos para ***S1*** pode ser visto na Figura 6.13, mostrando, o IP e porta do dispositivo, o nome do objeto SNMP para qual foi gerado o alarme e o *timestamp*.

```
          Device: S1 - IP Address: 172.xx.xx.4
...
01Jan2010;03:25:24;SNMP_OBJECT:ifOutOctets;port:1;
...
```

Figura 6.13 – Estudo de caso 2: Alarmes emitidos para *switch S1.*

No *Firewall* foram emitidos alarmes para os objetos *ipOutRequests* ilustrado na Figura 6.14.

```
       Device: Firewall - IP Address: 189.xx.xx.2
...
01Jan2010;03:28:07;SNMP_OBJECT:ipOutRequests;
...
```

Figura 6.14 – Estudo de caso 2: Alarmes emitidos para o *Firewall.*

De igual modo, o sistema de alarmes disparou alarmes para ***S2***, para o objeto *ifInOctets*, porta 3001, como mostrado na Figura 6.15

```
          Device: S2 - IP Address: 172.xx.xx.1
...
01Jan2010;03:28:50;SNMP_OBJECT:ifInOctets;port: 3001;
...
```

Figura 6.15 – Estudo de caso 2: Alarmes emitidos para *S2*

O sistema de alarme detectou que o tráfego anômalo chegou e saiu do servidor *Proxy* através da porta 3011 *de* ***S2***, no qual alarmes foram gerados para os objetos *ifInOctets* e *ifOutOctets*, conforme demonstrado na Figura 6.16.

```
          Device: S2 - IP Address: 172.xx.xx.1
...
01Jan2010;03:28:56;SNMP_OBJECT:ifInOctets;port:3011;
...
01Jan2010;03:29:03;SNMP_OBJECT:ifOutOctets;port:3011;
...
```

Figura 6.16 - Estudo de caso 2: Alarmes emitidos para o *S2* indicando a propagação para o servidor *Proxy*.

A Figura 6.17 apresenta alarmes gerados para ***S2*** na porta 3016 para o objeto *ifOutOctets*, indicando que a anomalia se destinou para o servidor *Web*.

```
          Device: S2 - IP Address: 172.xx.xx.1
...
01Jan2010;03:26:40;SNMP_OBJECT:ifOutOctets;port:3016;
...
```

Figura 6.17 - Estudo de caso 2: Alarmes emitidos para o *S2,* notificando a propagação para o servidor *Web*.

Os DLAs desse estudo de caso são mostrados na Figura 6.18.

```
@G2
S1, Firewall
@G2
Firewall, S2
@G1-G2
S2, Proxy
@G2
S2, Web
```

Figura 6.18 – DLAs do estudo de caso 2.

De forma similar ao estudo de caso 1, iremos rotular cada dispositivo afetado pela anomalia. Assim, temos ***S1*** como x_0, *Firewall* como x_1, ***S2*** como x_2, servidor *Proxy* como x_6 e *servidor Web* como x_7. A correspondente matriz $\mathcal{D}$ é apresentada a seguir.

$$\mathcal{D} = \begin{array}{c} \begin{matrix} x_0 & x_1 & x_2 & x_6 & x_7 & \end{matrix} \\ \begin{bmatrix} 0 & 1, p_{0,1} & 0 & 0 & 0 & 1 \\ 0 & 0 & 1, p_{1,2} & 0 & 0 & 1 \\ 0 & 0 & 0 & 1, p_{2,6} & 1, p_{2,7} & 2 \\ 0 & 0 & 1, p_{6,2} & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 1 & 1 & \end{bmatrix} \end{array} \begin{matrix} \\ x_0 \\ x_1 \\ x_2 \\ x_6 \\ x_7 \end{matrix} \quad (6.2)$$

Para esse estudo de caso temos, $S = \{x_0\}$ candidato para ser fonte, no qual é destacado com o círculo vermelho em (6.2). Temos $F = \{x_7\}$, o possível destino do tráfego anômalo destacado com o triângulo azul em (6.2). O sistema de correlação identificou que anomalia iniciou em x_0 (***S1***), tendo como destino x_7 (servidor *Web*). A Figura 6.19 apresenta o cenário anômalo descrito nesse estudo de caso, onde é possível ter uma visão global do impacto da anomalia na rede.

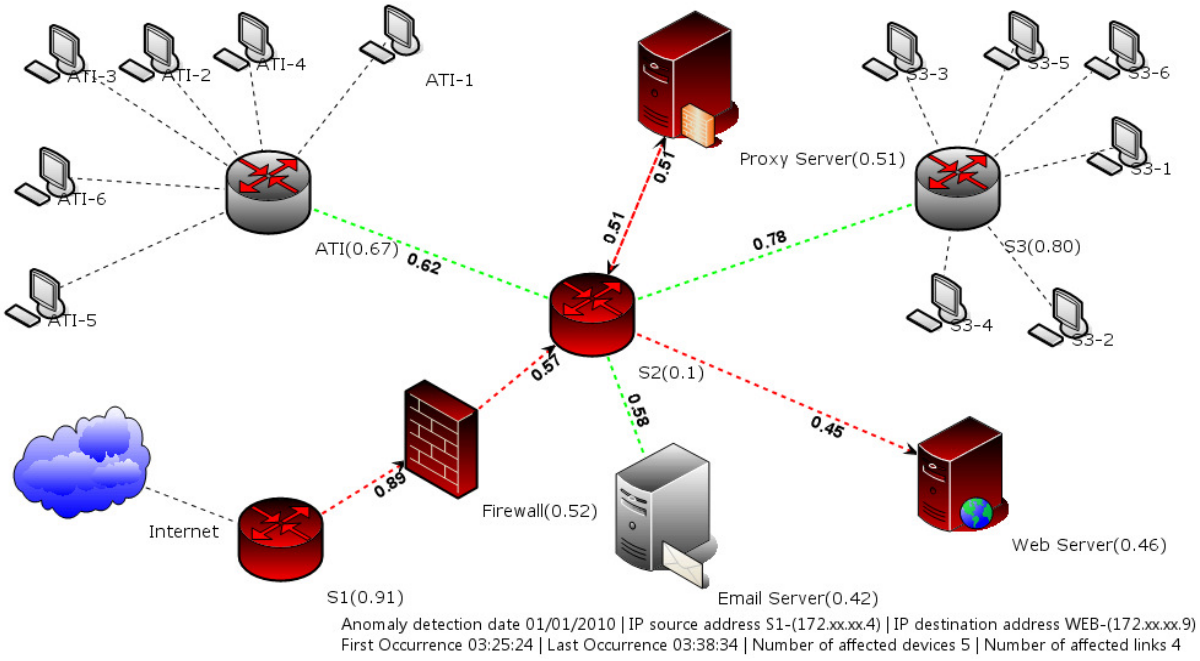

Figura 6.19 – Cenário de rede afetado pela anomalia no estudo de casos 2.

A taxa de redução dos alarmes para esse estudo de caso foi de 87,5%. Para inferir a origem e destino do tráfego anômalo, o sistema de correlação utilizou 12,5% do total de alarmes. Essa porcentagem refere-se ao volume de alarmes (DLAs) que foi provida pela camada de pré-processamento para a camada de correlação.

6.5.3 Estudo de caso 3

O estudo de caso 3 inclui um cenário com a presença de alarmes espúrios impactando na identificação do caminho de propagação anomalia, bem como sua origem e destino. Esse evento

anômalo ocorreu em 16/05/2010 e se iniciou às 17h10min23seg, com a última observação às 17h36min20seg. O tráfego anômalo afetou os equipamentos ***S2***, *Firewall*, servidor *Proxy*, ***S1*** e ***S3***.

O sistema de alarmes detectou a anomalia em ***S2*** na porta 4011, emitindo alarmes para o objeto *ifInOctets*, como mostrado a Figura 6.20. Esses alarmes indicam que o tráfego anômalo partiu de ***S3***.

```
          Device: S2 - IP Address: 172.xx.xx.1
...
16May2010;17:10:50;SNMP_OBJECT:ifInOctets;port:4011;
...
```

Figura 6.20 – Estudo de caso 3: Alarmes emitidos para *S2* indicando que a anomalia partiu de *S3*.

Foi detectado que a propagação da anomalia afetou o servidor *Proxy*, gerando alarmes para objetos *ipOutRequests* e *tcpOutSegs*, respectivamente, conforme ilustrado na Figura 6.21.

```
     Device: Servidor Proxy- IP Address: 172.xx.xx.11
...
16May2010;05:12:52;SNMP_OBJECT:ipOutRequests;
...
16May2010;05:12:52;SNMP_OBJECT:tcpOutSegs;
...
```

Figura 6.21 – Estudo de caso 3: Alarmes emitidos para o servidor *Proxy*.

A Figura 6.22 apresenta o alarme para o objeto *ifInOctets* na porta 3011 de ***S2***.

```
          Device: S2 - IP Address: 172.xx.xx.1
...
16May2010;05:12:50;SNMP_OBJECT:ifInOctets;port: 3011;
...
```

Figura 6.22 – Estudo de caso 3: Alarmes emitidos para *S2* indicando que o tráfego anômalo partiu do servidor *Proxy*.

O sistema de alarmes detectou o tráfego anômalo saindo de ***S2*** e indo para o *Firewall*. Isso é indicado pelos alarmes emitidos para o objeto *ifOutOctets* apresentado na Figura 6.23.

```
          Device: S2 - IP Address: 172.xx.xx.1
...
16May2010;05:11:40;SNMP_OBJECT:ifOutOctets;port:3001;
...
```

Figura 6.23 – Estudo de caso 3: Alarmes emitidos para *S2* indicando que o tráfego anômalo foi para o *Firewall*.

No *Firewall*, alarmes foram emitidos para o objeto *ipInReceives*, como observado na Figura 6.24.

```
          Firewall - IP Address: 189.xx.xx.2
...
16May2010;05:11:23;SNMP_OBJECT:ipInReceives;
...
```

Figura 6.24 – Estudo de caso 3: Alarmes emitidos para o equipamento *Firewall*.

Como ilustrado na Figura 6.25, alarmes foram gerados para o objeto *ifInOctets*. Isso demonstra que a propagação do tráfego anômalo afetou o ***S1***.

```
          Device: S1 - IP Address: 172.xx.xx.4
...
16May2010;05:10:23;SNMP_OBJECT:ifInOctets;port:1;
...
```

Figura 6.25 – Estudo de caso 3: Alarmes emitidos para *S1*.

Os DLAs gerados na camada de pré-processamento são mostrados na Figura 6.26. A taxa de redução dos alarmes para esse estudo de caso foi de 82,6%.

```
@G1
S1, Firewall
@G2
S2, Firewall
@G1
S2, S3
@G1
S2, Proxy
```

Figura 6.26 – DLAs do estudo de caso 3.

A matriz $\mathcal{D}$ correspondente ao estudo de caso 3 é apresentada em (6.3).

$$\mathcal{D} = \begin{array}{c} \begin{matrix} x_0 & x_1 & x_2 & x_4 & x_6 & \end{matrix} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1, p_{1,0} & 0 & 0 & 0 & 0 & 1 \\ 0 & 1, p_{2,1} & 0 & 0 & 0 & 1 \\ 0 & 0 & 1, p_{4,2} & 0 & 0 & 1 \\ 0 & 0 & 1, p_{6,2} & 0 & 0 & 1 \\ 1 & 1 & 2 & 0 & 0 & \end{bmatrix} \end{array} \begin{matrix} x_0 \\ x_1 \\ x_2 \\ x_4 \\ x_6 \end{matrix} \tag{6.3}$$

Utilizando os mesmos rótulos descritos nos estudos de casos anteriores, temos ***S1*** como x_0, *Firewall* como x_1, ***S2*** como x_2, ***S3*** como x_4 e servidor *Proxy* como x_6. Observa-se em (6.3) a existência de dois círculos vermelhos. Isso indica que temos, $S = \{x_4,\ x_6\}$. Como indicado pelo triângulo azul temos $F = \{x_0\}$. Portanto, temos $\#S = 2$ e $\#F = 1$. $\#S = 2$ é um indicativo, segundo o pressuposto de que há apenas uma fonte de anomalia em uma janela de tempo, de que há alarmes espúrios no conjunto de dados. Como consequência, temos duas prováveis fontes como apresentado na Figura 6.27.

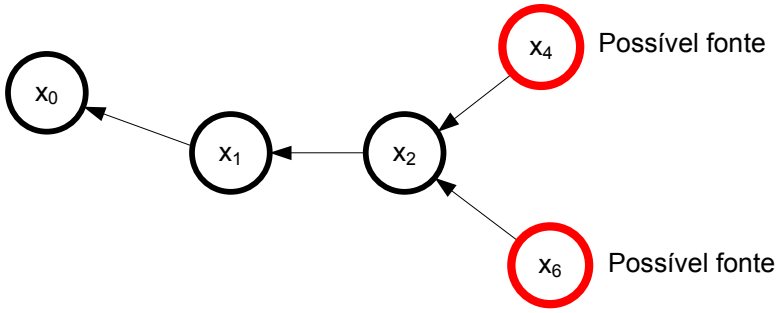

Figura 6.27 - Cenário com duas prováveis fontes de anomalia.

Dessa forma, é utilizada a medida de incerteza baseada na entropia generalizada para inferir qual é a fonte mais provável da anomalia. Há dois dispositivos candidatos para ser a fonte da anomalia, x_4 e x_6, resultando em dois caminhos, P_1 e P_2. P_1 tem início em x_4 e é constituído pelos enlaces $x_4 \rightarrow x_2 \rightarrow x_1 \rightarrow x_0$. P_2 inicia em x_6 com os enlaces $x_6 \rightarrow x_2 \rightarrow x_1 \rightarrow x_0$.

Iniciando por P_1, temos os respectivos valores de probabilidades para os enlaces: $P(x_4, x_2) = 0.78$, $P(x_2, x_1) = 0.58$, $P(x_1, x_0) = 0.81$. Então temos que $\boldsymbol{\gamma} = \{0.78,\ 0.58,\ 0.81\}$.

Essas probabilidades foram obtidas de dados históricos e calculadas através da equação (5.18), conforme descrito na seção 5.4.4. Para o caminho P_2 temos as probabilidades dos enlaces $P(x_6, x_2) = 0.51$, $P(x_2, x_1) = 0.58$, $P(x_1, x_0) = 0.81$ e seu vetor $\boldsymbol{\gamma} = \{0.51,\ 0.58,\ 0.81\}$.

O próximo passo é calcular o valor de incerteza de ambos os caminhos usando a entropia generalizada descrita pela fórmula (5.20). Usando valor $q = 1{,}1$ temos que $\boldsymbol{H}_{1.1}(P_1) \cong 3{,}7$ e $\boldsymbol{H}_{1.1}(P_2) \cong 4{,}5$, como é apresentado na Figura 6.28, gerada pelo sistema de correlação. O sistema de correlação utiliza o critério (5.21). Através desse critério é inferido que o caminho mais provável é $\mathbf{MPP} = \boldsymbol{H}(P_1)$, pois o caminho P_1 possui o menor valor de entropia. Como resultado, é inferido que a fonte do tráfego anômalo é x_4 (***S3***) e o destino é x_0 (***S1***).

O valor entrópico $q = 1{,}1$ foi escolhido, pois trouxe maior sensibilidade ao sistema de correlação. Sensibilidade no sentido de que, embora os caminhos tenham pequenas diferenças, em termos de probabilidade, de ser o caminho mais provável percorrido pela anomalia, seus valores de entropia serão bem distintos.

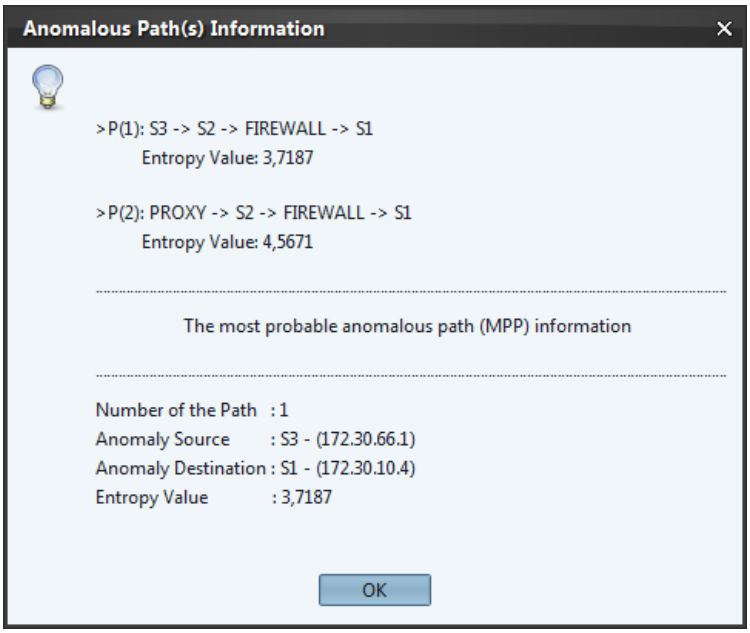

Figura 6.28 – Informações dos caminhos gerados pelo sistema de correlação para o caso de estudo 3.

A Figura 6.29 apresenta o cenário completo gerado pela ferramenta APV para este estudo de caso.

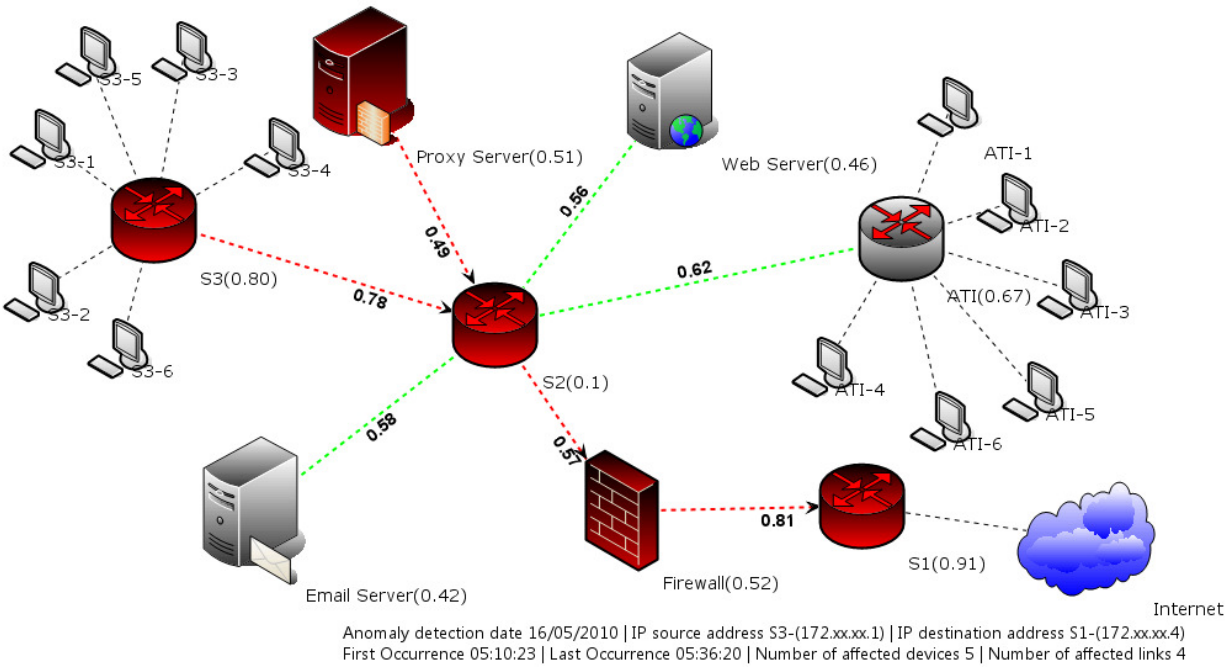

Figura 6.29 – Visão holística da rede afetada pelo evento anômalo do estudo de caso 3.

Diferente dos dois estudos de casos anteriores, esse estudo de caso apresentou um cenário com alarmes espúrios e como esses alarmes indesejados geraram incertezas no processo de correlação. Foi mostrado passo a passo como o sistema de alarmes inferiu a origem e destino do tráfego anômalo utilizando a medida de entropia generalizada.

6.5.4 Estudo de caso 4

De maneira similar ao estudo de caso 3, esse estudo de caso apresenta outro cenário com a presença de alarmes espúrios. A data da ocorrência da anomalia foi em 31/07/2010, com o primeiro alarme disparado às 17h01min45seg e o último às 17h19min20seg. O tráfego anômalo afetou os equipamentos ***S1***, *Firewall*, servidor *Proxy*, Servidor de Email, e o *switch* ATI.

O sistema de alarmes detectou a anomalia em ***S1***, na porta 1, emitindo alarmes para o objeto *ifOutOctets*, como mostrado na Figura 6.30.

```
          Device: S1 - IP Address: 172.xx.xx.4
...
31Jul2010;17:01:45;SNMP_OBJECT:ifOutOctets;port:1;
...
```

Figura 6.30 – Estudo de caso 4: Alarmes emitidos para *S1.*

O sistema de alarmes detectou o tráfego anômalo saindo do *Firewall* para ***S2***. Isso é indicado pelos alarmes emitidos para o objeto *ifInOctets***,** na porta 3001, como é apresentado na Figura 6.31.

```
          Device: S2 - IP Address: 172.xx.xx.1
...
31Jul2010;17:15:30;SNMP_OBJECT:ifInOctets;port:3001;
...
```

Figura 6.31 – Estudo de caso 4: Alarmes emitidos para *S2,* na porta 3001, indicando que o tráfego anômalo partiu do *Firewall.*

De igual modo, alarmes foram emitidos para os objetos *ifInOctets* e *ifOutOctets* na porta 3011 de ***S2,*** como mostra a Figura 6.32. Esses alarmes são indicadores que tráfego anômalo saiu de ***S2*** para o servidor *Proxy* e retornou para ***S2***.

```
          Device: S2 - IP Address: 172.xx.xx.1
...
31Jul2010;17:15:40;SNMP_OBJECT:ifInOctets;port:3011;
...
31Jul2010;17:15:40;SNMP_OBJECT:ifOutOctets;port:3011;
...
```

Figura 6.32 – Estudo de caso 4: Alarmes emitidos para *S2* na porta 3011.

Alarmes para o objeto *ifInOctets* foram emitidos para o *switch* ***S2*** na porta 3015 como ilustrado na Figura 6.33. Esses alarmes sinalizam que a anomalia partiu do servidor de email e chegou ao ***S2***.

```
          Device: S2 - IP Address: 172.xx.xx.1
...
31Jul2010;17:12:00;SNMP_OBJECT:ifInOctets;port:3015;
...
```

Figura 6.33 – Estudo de caso 4: Alarmes emitidos para *S2* na porta 3015.

Por fim, foram disparados alarmes para o objeto *ifOutOctets* na porta 5001 do *switch* ***S2*** como mostra a Figura 6.34. Isso indica que o tráfego anômalo saiu de ***S2*** para o *switch* ATI.

```
          Device: S2 - IP Address: 172.xx.xx.1
...
31Jul2010;17:19:20;SNMP_OBJECT:ifOutOctets;port:5001;
...
```

Figura 6.34 – Estudo de caso 4: Alarmes emitidos para *S2*, na porta 5001.

A operação de compressão realizada na camada de pré-processamento reduziu o volume de alarmes primitivos em 58,3%. Os DLAs desse estudo de caso são mostrados na Figura 6.35.

```
@G2
S1, Firewall
@G2
Firewall, S2
@G1
S2, Proxy
@G2
S2, ATI
@G1
S2, Email
```

Figura 6.35 – DLAs do estudo de caso 4.

A matriz $\mathcal{D}$ correspondente ao estudo de caso 4 é apresentada em (6.4).

$$\mathcal{D} = \begin{array}{c} \begin{matrix} x_0 & x_1 & x_2 & x_3 & x_5 & x_6 & \end{matrix} \\ \begin{bmatrix} 0 & 1,p_{0,1} & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1,p_{1,2} & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1,p_{2,3} & 0 & 1,p_{2,6} & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1,p_{5,2} & 0 & 0 & 0 & 1 \\ 0 & 0 & 1,p_{6,2} & 0 & 0 & 0 & 1 \\ 0 & 1 & 3 & 1 & 0 & 1 & \end{bmatrix} \end{array} \begin{matrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_5 \\ x_6 \end{matrix} \tag{6.4}$$

Nesse estudo de caso, temos ***S1*** como x_0, *Firewall* como x_1, ***S2*** como x_2, *switch* ATI como x_3, servidor de Email como x_5 e o servidor *Proxy* como x_6. De (6.4), temos $S = \{x_0, x_5\}$ e $F = \{x_3\}$.

Temos $\#S = 2$, indicando a existência de duas prováveis fontes da anomalia. A Figura 6.36 mostra o cenário anômalo com as duas possíveis fontes destacadas em vermelho.

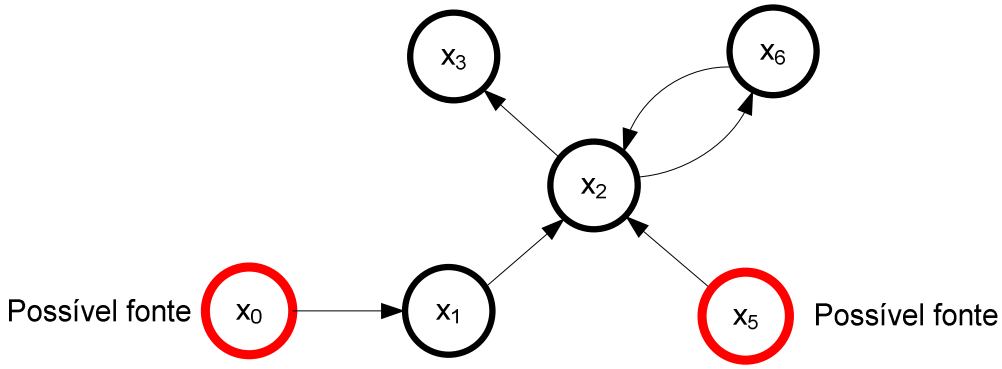

Figura 6.36 - Cenário com duas prováveis fontes de anomalia do estudo de caso 4.

Nesse estudo de caso, temos dois caminhos possíveis, representados por P_1 e P_2. O primeiro tem origem em x_0 com os respectivos enlaces $x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow x_6 \rightarrow x_2 \rightarrow x_3$. P_2 se inicia em x_5 e possui os enlaces $x_5 \rightarrow x_2 \rightarrow x_6 \rightarrow x_2 \rightarrow x_3$.

As probabilidades *a priori* dos enlaces de P_1 são: $P(x_0, x_1) = 0.89$, $P(x_1, x_2) = 0.58$, $P(x_2, x_6) = 0.51$, $P(x_6, x_2) = 0.49$, $P(x_2, x_3) = 0.32$. Então, temos que $\boldsymbol{\gamma} = \{0.89,\ 0.58,\ 0.51,\ 0.49,\ 0.32\}$.

Para o caminho P_2 temos as seguintes probabilidades atribuídas aos enlaces: $P(x_5, x_2) = 0.58$, $P(x_2, x_6) = 0.51$, $P(x_6, x_2) = 0.49$, $P(x_2, x_3) = 0.32$ resultando no vetor $\boldsymbol{\gamma} = \{0.58,\ 0.51,\ 0.49,\ 0.32\}$. Como descrito na seção 5.4.5, depois de identificar os possíveis caminhos, o sistema de alarmes calcula a incerteza/entropia de cada um deles a fim de inferir o caminho de propagação do tráfego anômalo.

Similar ao estudo de caso 3, é utilizado o valor $q = 1{,}1$. De (5.20), temos $\boldsymbol{H}_{1.1}(P_1) \cong 5{,}5$ e $\boldsymbol{H}_{1.1}(P_2) \cong 6{,}1$ como é apresentado na Figura 6.37 gerada pelo sistema de correlação. O sistema de correlação infere pelo critério (5.21) que o caminho mais provável é $\mathbf{MPP} = \boldsymbol{H}(P_1)$ tendo como fonte x_0 (***S1***) e destino x_3 (*switch* ATI).

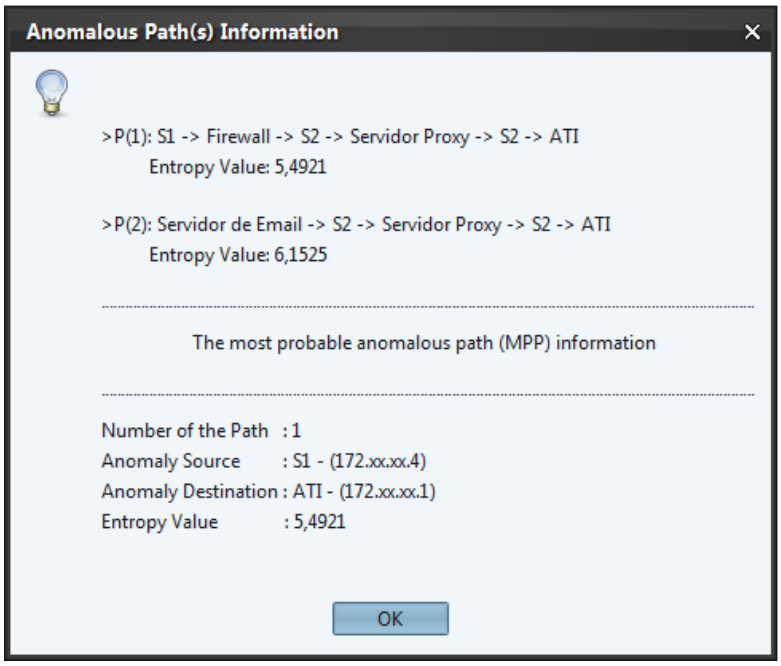

Figura 6.37 – Informações dos caminhos gerados pelo sistema de correlação para o caso de estudo 4.

A Figura 6.38 apresenta a visão global da rede gerada pela ferramenta APV.

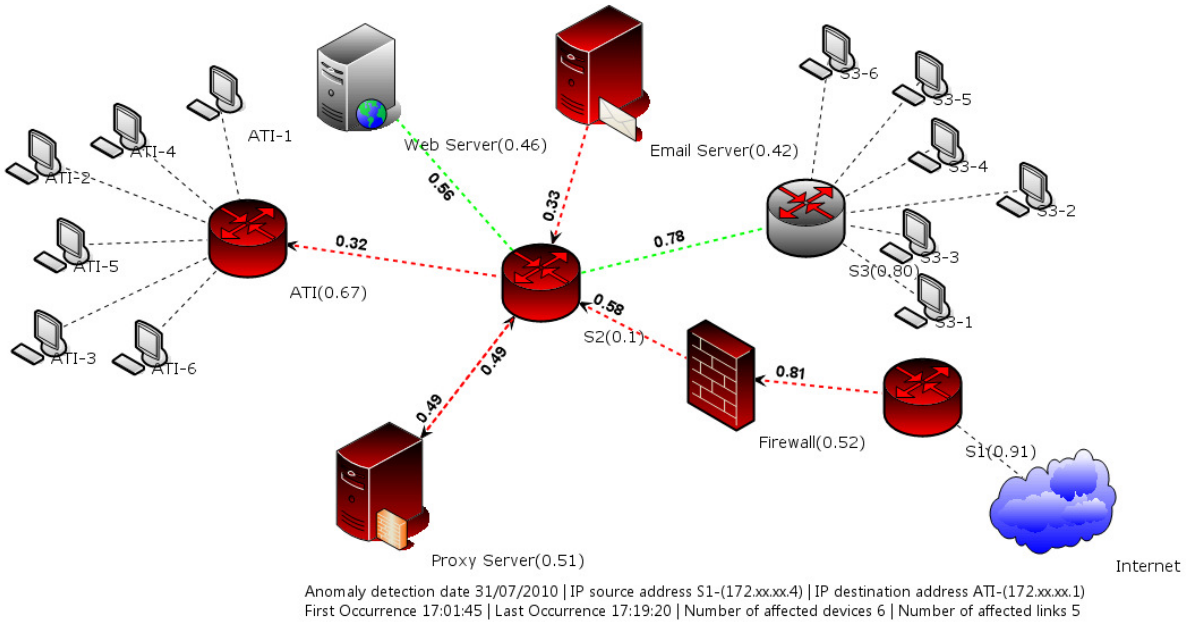

Figura 6.38 – Visão holística da rede afetada pelo evento anômalo do estudo de caso 4.

Similar ao estudo de caso três, esse estudo de caso apresentou um cenário com alarmes espúrios. Através da entropia generalizada, ***S1*** foi inferido como sendo a fonte e o *switch* que

conecta a sub-rede da ATI como sendo o destino do tráfego anômalo. De igual modo, nos estudos de caso um e dois, ***S1*** também foi identificado como fonte do tráfego anômalo, o que mostra que, na maioria das vezes a anomalia é gerada na Internet e destinada ao domínio administrativo da rede da Universidade Estadual de Londrina.

6.5.5 Estudo de caso 5

Esse estudo de caso irá apresentar um evento anômalo, no qual um tráfego anômalo foi injetado artificialmente na rede. Para tanto, foi utilizada a ferramenta geradora de tráfego *open-source* e multiplataforma denominada Ostinato [50]. Essa ferramenta possibilita a geração de um grande um volume de pacotes de diferentes protocolos (TCP, UDP, ICMP, IGMP), que pode ser configurada para que eles sejam enviados para um endereço IP com diferentes taxas. Maiores informações podem ser encontradas em [50].

Este procedimento foi realizado em 04/05/2011. Foram utilizados dois *hosts*. O primeiro localizado na sub-rede da ATI (Assessoria de Tecnologia de Informação) e o segundo na Internet como ponto de destino do tráfego gerado. Então, a ferramenta Ostinato foi configurada para gerar um tráfego ICMP com taxa de 10.000 pacotes por segundo, iniciando a transmissão às 22h50min00seg e encerrando às 22h55min24seg. A Figura 6.39 mostra a localização da fonte do tráfego e o caminho com destino a Internet.

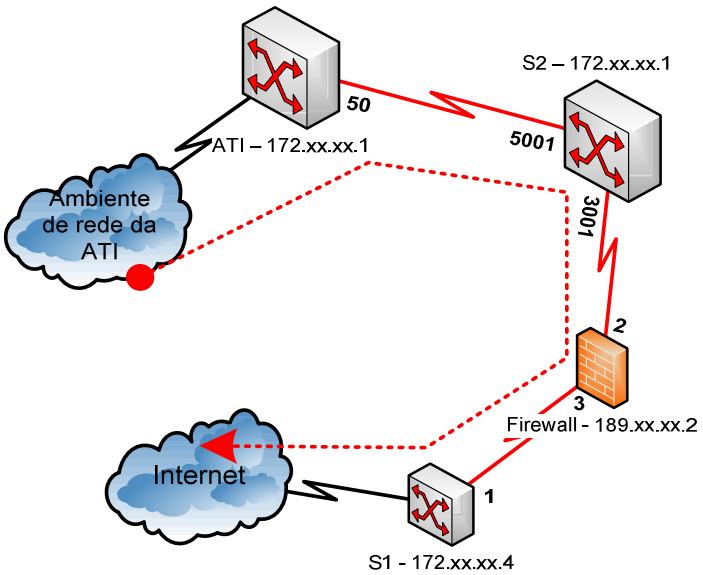

Figura 6.39 – Equipamentos e caminho de propagação do tráfego anômalo.

Para esse estudo de caso, o sistema de alarmes emitiu alarmes para o *switch* ***S2*** para a objeto *ifInOctets,* porta 5001, como mostrado a Figura 6.40.

```
          Device: S2 - IP Address: 172.xx.xx.1
...
04May2011;22:51:20;SNMP_OBJECT:ifInOctets;port:5001;
...
```

Figura 6.40 – Estudo de caso 5: Alarmes emitidos para *S2* indicando que partiu da sub-rede ATI.

De igual modo, a Figura 6.41 apresenta alarmes emitidos para ***S2***, na porta 3001, para o objeto *ifOutOctets*, indicando que o tráfego anômalo foi em direção ao *Firewall*.

```
          Device: S2 - IP Address: 172.xx.xx.1
...
04May2011;22:51:10;SNMP_OBJECT:ifOutOctets;port:3001;
...
```

Figura 6.41– Estudo de caso 5: Alarmes emitidos para *S2* sendo um indicador de que o tráfego está partindo de *S2* para o *Firewall*.

No *Firewall*, o sistema de alarmes detectou o tráfego anômalo e disparou alarmes para o objeto *ifOutOctets*, como mostra a Figura 6.42 a seguir.

```
       Device: Firewall - IP Address: 189.xx.xx.2
...
04May2011;22:51:39;SNMP_OBJECT:ifInOctets;port:3;
...
```

Figura 6.42– Estudo de caso 5: Alarmes emitidos para o *Firewall.*

A Figura 6.43 mostra alarmes gerados para S1 para o objeto *IfInOctets*.

```
          Device: S1 - IP Address: 172.xx.xx.1
...
04May2011;22:51:12;SNMP_OBJECT:ifInOctets;port:1;
...
```

Figura 6.43– Estudo de caso 5: Alarmes emitidos para *S1.*

A taxa de redução dos alarmes para esse estudo de caso foi de 40%. Os DLAs desse estudo de caso são mostrados na Figura 6.44.

```
@G1
S1, Firewall
@G1
Firewall, S2
@G1
S2, ATI
```

Figura 6.44 – DLAs do estudo de caso 5.

Os rótulos dos equipamentos envolvidos nesse evento anômalo são: ***S1*** como x_0, *Firewall* como x_1, ***S2*** como x_2 e switch ATI como x_3. A correspondente matriz $\mathcal{D}$ é apresentada a seguir.

$$\mathcal{D} = \begin{array}{c} \begin{matrix} x_0 & x_1 & x_2 & x_3 \end{matrix} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1, p_{1,0} & 0 & 0 & 0 & 1 \\ 0 & 1, p_{2,1} & 0 & 0 & 1 \\ 0 & 0 & 1, p_{3,2} & 0 & 1 \\ 1 & 1 & 1 & 0 & \end{bmatrix} \end{array} \begin{matrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{matrix} \tag{6.5}$$

Pela matriz $\mathcal{D}$ temos, $S = \{x_3\}$ e $F = \{x_0\}$. O equipamento x_3 (ATI) e x_0 (***S1***) são identificados corretamente pelo sistema de correlação como sendo a fonte e o destino do tráfego anômalo, respectivamente. A Figura 6.45 apresenta o trajeto da propagação da anomalia dentro do domínio administrativo.

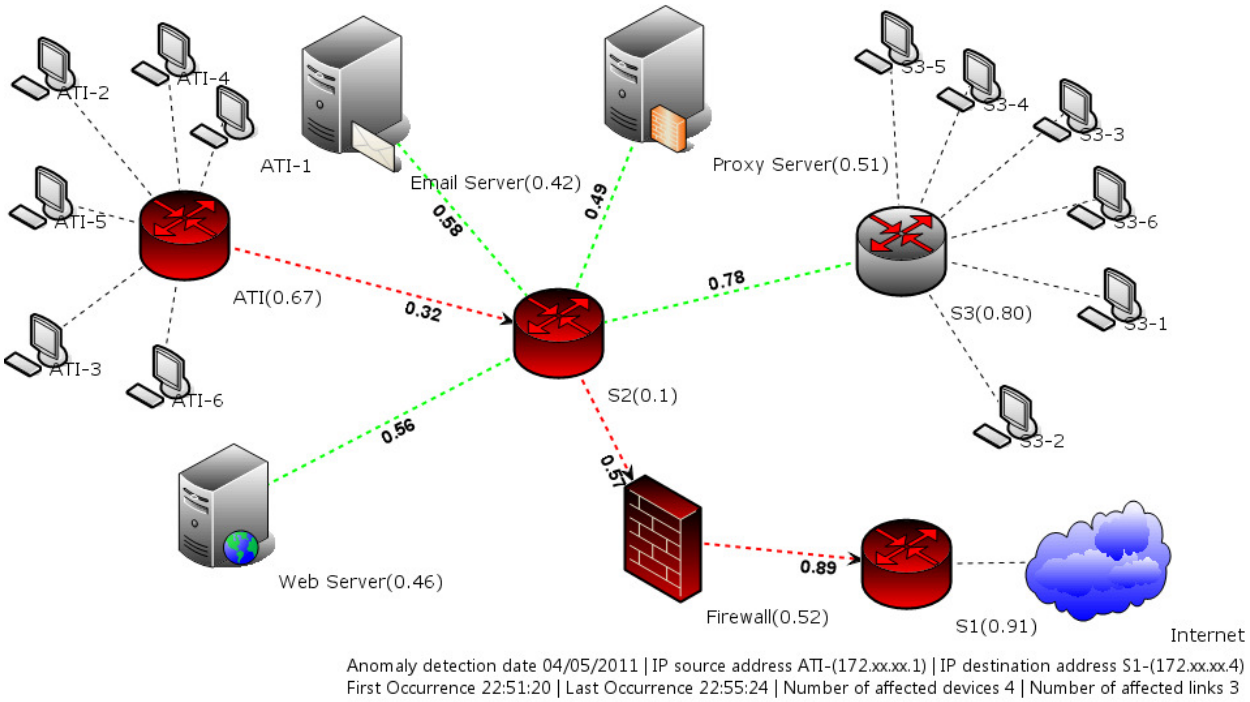

Figura 6.45 – Visão global da rede do estudo de caso 5.

O conhecimento prévio da topologia da rede e as dependências entre os seus elementos possibilitam a comparação do resultado gerado pelo sistema de correlação com o resultado esperado. Nesse estudo de caso, foi injetada uma anomalia que se iniciava na sub-rede da ATI com destino a Internet. O experimento demonstrou a capacidade do sistema de correlação de identificar corretamente o caminho de propagação da anomalia inferindo sua fonte e destino de forma automatizada.

Conclusão

Nesta dissertação foi apresentado um sistema de correlação de alarmes com o objetivo de inferir a fonte e o destino de uma anomalia. Enquanto as abordagens de correlação encontradas na literatura têm focado na redução do volume de alarmes e na identificação da origem de falhas de *hardwares* e *softwares* de rede, foi demonstrado nessa dissertação que a correlação de alarmes pode ser aplicada de uma forma mais ampla para tratar anomalias de volume.

O sistema de correlação apresentado neste trabalho foi divido em três camadas: pré-processamento, correlação e apresentação.

A camada de pré-processamento é responsável pela compressão dos alarmes primitivos utilizando seus atributos espacial e temporal. Alarmes disparados pelo mesmo dispositivo na ocorrência de uma anomalia são reduzidos a um único alarme, denominado nesse trabalho de DLA. O resultado desse processo minimiza o número de alarmes para cada dispositivo afetado por uma anomalia, consequentemente reduzindo a quantidade de alarmes no contexto global. Os DLAs possuem informações com um valor semântico maior, quando comparado com os alarmes primitivos, pois descrevem como o equipamento de rede foi afetado pelo tráfego anômalo.

A camada de correlação tem por objetivo correlacionar os alarmes previamente gerados pela camada de pré-processamento, chamados de DLAs. Através dos DLAs e de informações sobre a topologia da rede é produzida a melhor explicação sobre o evento anômalo observado. As informações sobre a topologia da rede descrevem as dependências entre os dispositivos de rede, representadas por um grafo de dependência. Essa camada implementa um método para inferir o caminho de propagação da anomalia, bem como sua origem e destino, mesmo em cenários com alarmes falsos e perdidos, na qual denominamos de alarmes espúrios. Para minimizar a incerteza gerada pelos alarmes espúrios no processo de inferência do caminho de propagação do tráfego anômalo, uma medida baseada na entropia generalizada de Tsallis foi

implementada. Como demonstrado nos estudos de caso, essa medida tornou o sistema de correlação capaz de atuar em cenários com incertezas geradas por alarmes espúrios, possibilitando sua utilização em um ambiente de rede real.

A camada de apresentação fornece uma visão global do estado da rede na ocorrência da anomalia. Ela tem como principal objetivo proporcionar ao administrador de rede um meio amigável para analisar o resultado da correlação de alarmes. Diferente de propostas que apresentam o resultado da correlação em formato texto, neste trabalho foi utilizada a representação gráfica para auxiliar a análise por parte do administrador. Para este fim, a ferramenta APV (*Anomaly Propagation View*) foi desenvolvida. Essa é uma ferramenta multiplataforma desenvolvida com linguagem Java, que apresenta os elementos de redes afetados pelo tráfego anômalo, possibilitando a visualização e um rápido entendimento do impacto da sua propagação na rede. O rápido entendimento do problema ocorrido na rede está diretamente ligado ao tempo de restauração de sua operação normal, minimização de prejuízos, perda de receita e insatisfação de clientes.

O sistema de correlação foi avaliado utilizando dados reais obtidos da rede da Universidade Estadual de Londrina. Diferentes experimentos foram realizados utilizando dados de anomalias reais, detectadas ao longo dos anos de 2009, 2010 e 2011. A fim de validar a proposta, foram apresentados nesse trabalho cinco estudos de caso.

Os estudos de caso um e dois apresentaram um cenário ideal, onde não foram identificados alarmes espúrios influenciando no processo de inferência do caminho de propagação da anomalia. A taxa de redução de alarmes do estudo de caso um foi de 71.4%, e do estudo de caso dois 87,5%.

Cenários com alarmes espúrios foram apresentados nos estudos de caso três e quatro. Neles, foi demonstrado como a medida baseada na entropia generalizada foi utilizada pelo sistema de correlação para inferir a origem e destino do tráfego anômalo. Para o estudo de caso três, o sistema de correlação obteve uma taxa de redução de 82,6%, e de 58,3% para estudo de caso quatro.

Além das anomalias diariamente detectadas na rede da Universidade Estadual de Londrina, foram geradas anomalias injetadas artificialmente na rede. A ferramenta Ostinato foi

utilizada para este propósito. Esse procedimento foi descrito no estudo de caso cinco, mostrando a capacidade do sistema de correlação de identificar o caminho de propagação e inferir a origem e o destino da anomalia. A taxa de redução de alarmes do estudo de caso cinco foi de 40%.

Destacam-se três principais contribuições desse trabalho: (i) um sistema de correlação de alarmes que tem por objetivo inferir a fonte e destino de anomalias de volume, (ii) uma medida baseada na entropia generalizada, aplicada para inferir o caminho percorrido pela anomalia, em cenários onde existem possíveis caminhos gerados por alarmes espúrios, permitindo que o sistema seja utilizado de forma eficiente em ambientes reais de redes e (iii) uma ferramenta que apresenta de forma gráfica o resultado da correlação, fornecendo de forma automatizada a visão holística do impacto da anomalia na rede.

Como sugestão de trabalho futuro, a proposta pode ser aprimorada para identificação de propagação da anomalia em cenários mais complexos, envolvendo alarmes espúrios e a existência de mais de uma fonte de anomalia em uma mesma janela de tempo. Para tanto, sugerimos que o modelo seja estendido para análises realizadas com dados de fluxos, gerados pelo IPFIX [11].

Neste trabalho, foi proposta uma solução onde o processamento e a correlação de alarmes é realizada de forma centralizada. Dado o crescimento acentuado das redes, há uma tendência de que também o volume de alarmes cresça em grande escala. Isso irá requerer grande processamento, principalmente em casos onde se busca aplicar a correlação de alarmes em tempo real. Nesse sentindo, este trabalho pode ser estendido de modo que a correlação ou parte dela seja realizada de forma distribuída.

Referências Bibliográficas

[1] A. Bouloutas, S. Calo e A. Finkel. **Alarm correlation and fault identification in communication networks**. IEEE Transactions on Communications, vol. 42, p. 523-533, 1994.

[2] A. Brown, G. Kar e A. Keller. **An active approach to characterizing dynamic dependencies for problem determination in a distributed environment**. In proceedings of IEEE/IFIP International Symposium on Integrated Network Management, 2001.

[3] A. Emad e D. Christos. **Fuzzy Temporal Reasoning Model for Event Correlation in Network Management**. In Proceedings of the 24th Annual IEEE Conference on Local Computer Networks (LCN '99), 1999.

[4] A. Hanemann. **A Hybrid Rule-Based/Case-Based Reasoning Approach for Service Fault Diagnosis**. In Proceedings of the 20th International Conference on Advanced Information Networking and Applications - Volume 02 (AINA '06), Vol. 2, p. 734-740, 2006.

[5] A. Lakhina, M. Crovella e C. Diot. **Mining anomalies using traffic feature distributions**. In Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM '05), 2005.

[6] A. Marnerides, K. Pezaros e D. Hutchison D., **Autonomic Diagnosis of Anomalous Network Traffic**. 4th IEEE WoWMoM Workshop on Autonomic and Opportunistic Communications (AOC 2010), 2010.

[7] A. Mohamed e O. Basir. **Fusion Based Approach for Distributed Alarm Correlation in Computer Networks**. In Proceedings of the 2010 Second International Conference on Communication Software and Networks (ICCSN '10), 2010.

[8] A. Nahid, F. Mahmood e D. Mehdi. **A case-based reasoning method for alarm filtering and correlation in telecommunication networks**. In Proceedings of the Canadian Conference on Electrical and Computer Engineering, p. 2182–2186, 2005.

[9] A. Patcha e J. Park. **An Overview of Anomaly Detection Techniques: Existing Solution and Latest Technological Trends**, Computer Networks, 2007.

[10] A. Ziviani, L. Monsores, S. Rodrigues e A. Gomes. **Network Anomaly Detection using Nonextensive Entropy**. In: IEEE Communications Letters, IEEE Press, ISSN: 1089-7798,vol.11, no. 12, p. 1034-1036, 2007.

[11] B. Claise. **IP Flow Information Export (IPFIX) Mediation: Framework**. RFC 6183, IETF, 2011.

[12] B. B. Zarpelão, L. S. Mendes, M. Proença Jr., e J. Rodrigues. **Parameterized Anomaly Detection System with Automatic Configuration**. IEEE Global Communications Conference (IEEE GLOBECOM 2009), Communications Software and Services Symposium (GC'09 CSS), 2009.

[13] B. B. Zarpelão. **Detecção de Anomalias em Redes de Computadores**. Tese de Doutorado, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, 2010. Disponível em: <http://www.bibliotecadigital.unicamp.br/document/?code=000776023>. Acesso em: 22 fevereiro 2011.

[14] C. Callegari, L. Gazzarrini, S. Giordano, M. Pagano e T. Pepe. **A novel multi time-scales PCA-based anomaly detection system**. Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2010 International Symposium, p.156-162, 2010.

[15] CSI. **Computer Crime and Security Survey 2009.** Disponível em: < http://gocsi.com/survey_2009>. Acesso em: 17 novembro 2010.

[16] C. Tsallis. **Possible generalization of Boltzmann-Gibbs statistics**. J. Statistical Physics, vol. 52, no. 1-2, p. 479–487, 1988.

[17] D. Gürer, I. Khan, R. Ogier e R. Keffer. **An Artificial Intelligence Approach to Network Fault Management**. SRI International, Menlo Park, California, USA, 1996.

[18] D. Jianguo, K. Bernd, B. Yingcai e C. Hansheng. **Probabilistic Inference for Network Management.** In Universal Multiservice Networks: Third European Conference (ECUMN 2004), Springer-Verlag, p. 498-507, 2004.

[19] D. Kim, H. Shinbo e H. Yokota. **An alarm correlation algorithm for network management based on root cause analysis**. Advanced Communication Technology (ICACT), 2011 13th International Conference on, 2011.

[20] E. Ekaette e B. Far. **A framework for distributed fault management using intelligent software agents**. Electrical and Computer Engineering, 2003. IEEE CCECE 2003. Canadian Conference on , vol.2, p. 797- 800, 2003.

[21] F. Hashim, K. Munasinghe e A. Jamalipour. **Biologically Inspired Anomaly Detection and Security Control Frameworks for Complex Heterogeneous Networks**. Network and Service Management, IEEE Transactions on, vol.7, no.4, p.268-281, 2010.

[22] F. Michaut e F. Lepage. **Application-oriented network metrology: metrics and active measurement tools**. Communications Surveys & Tutorials, IEEE, vol.7, no.2, p. 2- 24, 2005.

[23] F. Pouget e M. Dacier. **Alert correlation: Review of the state of the art**. Technical Report EURECOM, Institut Eurecom, France, 2003.

[24] G. Liu, A. Mok e E. Yang. **Composite Events for Network Event Correlation.** Proceedings of the 6th IFIP/IEEE International Symposium on Integrated Network Management, 1999.

[25] G. Tjhai, S. Furnell, M. Papadaki e N. Clarke. **A preliminary two-stage alarm correlation and filtering system using SOM neural network and K-means algorithm**. In Proceedings of Computers & Security, p.712-723, 2010.

[26] H. Nguyen e P. Thiran. **Failure location in transparent optical networks: the asymmetry between false and missing alarms**. In Proceedings of 19th International Teletraffic Congress, ITC19, 2005.

[27] H. Rahmani, N. Sahli e F. Kammoun. **Joint Entropy Analysis Model for DDoS Attack Detection**. Information Assurance and Security (IAS '09). Fifth International Conference on, vol.2, p. 267-271, 2009.

[28] I. Katzela e M. Schwartz. **Schemes for Fault Identification in communication Networks**. IEEE/ACM Transactions on networking, vol. 3, no. 6, 1995.

[29] J. Bellec e M. Kechadi. **Feck: A New Efficient Clustering Algorithm for the Events Correlation Problem in Telecommunication Networks**. In: Proceedings of the Future Generation Communication and Networking (FGCN 2007), pp. 469–475. IEEE Computer Society, 2007.

[30] J. Bellec e M. Kechadi. **Towards a formal model for the network alarm correlation problem**. In: The 6th WSEAS Int'l Conference on Simulation, Modelling and Optimization (SMO'06), 2006.

[31] J. Jung, B. Krishnamurthy e M. Rabinovich. **Flash crowds and denial of service attacks: characterization and implications for CDNs and web sites**. In Proceedings of the 11th international conference on World Wide Web (WWW '02), 2002.

[32] J. Klaus. **Clustering intrusion detection alarms to support root cause analysis**. ACM Transactions on Information Systems Security, vol. 6, no. 4, p. 443-471, 2003.

[33] J. Madaadhain, D. Fisher, P. Smyth, S. White e Y. Boey. **Analysis and Visualization of Network Data Using JUNG**. Journal of Statistical Software, vol. 10, p. 1-35, 2005.

[34] J. Martin-Flatin, G. Jakobson e L. Lewis. **Event Correlation in Integrated Management: Lessons Learned and Outlook**. Journal of Network and Systems Management, vol. 17, no. 4, p. 481-502, 2007.

[35] J. Moulierac e M. Molnar. **Active Monitoring of Link Delays in Case of Asymmetric Routes**. In Proceedings of the International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL '06), 2006.

[36] JUNG. **Java Universal Network/Graph Framework**. Disponível em: <http://jung.sourceforge.net/>. Acesso em: 21 Julho 2010.

[37] K. Limthong, P. Watanapongse e F. Kensuke. **A wavelet-based anomaly detection for outbound network traffic**. Information and Telecommunication Technologies (APSITT), 8th Asia-Pacific Symposium, 2010.

[38] K. Sun, S. Hiroyuki e Y. Hidetoshi. **An alarm correlation algorithm for network management based on root cause analysis.** Advanced Communication Technology (ICACT), 2011 13th International Conference on , p. 1233-1238, 2011.

[39] L. Fillatre, P. Casas, S. Vaton e I. Nikiforov. **Volume Anomaly Detection in Data Networks: an Optimal Detection Algorithm vs. the PCA Approach**. In: Valadas, R. (ed.) FITraMEn 2008. LNCS, vol. 5464, p. 96–113, 2008.

[40] L. Tong-Yan e L. Xing-Ming. **Preprocessing expert system for mining association rules in telecommunication networks**. Expert Systems with Applications, Volume 38, Issue 3, p. 1709-1715, 2011.

[41] M. Emmanuel, A. Armen, B. Stéphane, M. Oliver e D. Gérard. **Alarm correlation for complex telecommunication networks using neural networks and signal processing**. In IEEE workshop on IP operations and management, p. 3–7, 2002.

[42] M. Natu e A. Sethi. **Application of Adaptive Probing for Fault Diagnosis in Computer Networks**, In Proceedings of IEEE/ IFIP Network Operations and Management Symposium, 2008.

[43] M. Proença Jr. **Baseline Aplicado a Gerência de Redes**. Tese de Doutorado, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, 2005. Disponível em: <http://www.bibliotecadigital.unicamp.br/document/?code=vtls000365778>. Acesso em: 03 maio 2010.

[44] M. Thottan, C. Ji. **Anomaly detection in IP networks**. IEEE Transactions on Signal Processing, vol. 51, no 8, p. 2191-2204, 2003.

[45] M. Tiffany. **A Survey of Event Correlation Techniques and Related Topics**. Maio 2002. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.19.5339.

[46] M. Steinder e A. Sethi. **A survey of fault localization techniques in computer networks**. Science of Computer Programming, vol. 53, no. 2, p. 165–194, 2004.

[47] M. Steinder e A. Sethi. **Increasing robustness of fault localization through analysis of lost, spurious, and positive symptoms**. IEEE INFOCOM, 2002.

[48] NET-SNMP. **Net-SNMP**. Disponível em: <http://www.net-snmp.org>. Acesso em: 21 Maio 2011.

[49] ORACLE. **Enterprise JavaBeans Technology**. Disponível em: < http://www.oracle.com/technetwork/java/javaee/ejb>. Acesso em: 04 Agosto 2010.

[50] OSTINATO. **Packet/Traffic Generator and Analyzer.** Disponível em: < http://code.google.com/p/ostinato>. Acesso em: 23 março 2011.

[51] PANDA Security. **Annual Report: PandaLabas 2010.** Disponível em: < http://press.pandasecurity.com/wp-content/uploads/2010/05/PandaLabs-Annual-Report-2010.pdf>. Acesso em: 20 janeiro 2011.

[52] P. Baford, N. Duffield, A. Ron e J. Sommers. **Network performance anomaly detection and localization**. In Proceedings of INFOCOM, 2009.

[53] P. Casas, J. Mazel e P. Owezarski. **Steps Towards Autonomous Network Security: Unsupervised Detection of Network Attacks**. New Technologies, Mobility and Security (NTMS), 2011 4th IFIP International Conference, 2011.

[54] P. Du, S. Abe, Y. Ji, S. Sato e M. Ishiguro. **Detecting and Tracing Traffic Volume Anomalies in SINET3 Backbone Network**. In proceedings of IEEE International Conference on Communications (ICC'08), p. 5833-5837, 2008.

[55] P. Roberto, G. Giorgio e R. Roli. **Alarm clustering for intrusion detection systems in computer networks**. Engineering Applications of Artificial Intelligence, p. 429-438, 2006.

[56] R. Costa, N. Cachulo e P. Cortez. **An Intelligent Alarm Management System for Large-Scale Telecommunication Companies**. In Proceedings of the 14th Portuguese Conference on Artificial Intelligence: Progress in Artificial Intelligence (EPIA '09), Springer-Verlag, Berlin, Heidelberg, 2009.

[57] RFC 3954 – Internet Engineering Task Force (IETF). **Cisco Systems NetFlow Services Export Version 9**, RFC 3954, 2004.

[58] RFC 4765 – INTERNET ENGINEERING TASK FORCE (IETF). **The Intrusion Detection Message Exchange Format (IDMEF)**, RFC 4765, 2007.

[59] R. Kumar e J. Kaur. **Efficient beacon placement for network tomography**. In Proceedings of the 4th ACM SIGCOMM conference on Internet measurement (IMC '04). ACM, p. 181-186, 2004.

[60] R. Sadoddin e A. Ghorbani. **Alert correlation survey: framework and techniques**. In Proceedings of the 2006 International Conference on Privacy, Security and Trust: Bridge the Gap Between PST Technologies and Business Services, 2006.

[61] S. Agrawal, K. Naidu e R. Rastogi. **Diagnosing Link-Level Anomalies Using Passive Probes**. In Proceedings of INFOCOM (2007), p. 1757-1765, 2007.

[62] S. Al-Mamory e H. Zhang. **Intrusion detection alarms reduction using root cause analysis and clustering**. Computer Communications, v.32 n.2, p.419-430, 2009.

[63] S. Farraposo, P. Owezarski e E. Monteiro. **A Multi-Scale Tomographic Algorithm for Detecting and Classifying Traffic Anomalies**. In: IEEE ICC 2007, Glasgow, 2007.

[64] S. Jaiswal, G. Iannaccone, J. Kurose e D. Towsley. **Formal Analysis of Passive Measurement Inference Techniques**. In Proceedings of INFOCOM 2006. 25th IEEE International Conference on Computer Communications, 2006.

[65] S. Wallin e V. Leijon. **Rethinking Network Management Solutions**. IT Professional, vol.8, no.6, p. 19-23, 2006.

[66] S. Wallin e V. Leijon. **Telecom Network and Service Management: An Operator Survey**. In 12th IFIP/IEEE International Conference on Management of Multimedia and Mobile Networks and Services (MMNS), 2009.

[67] S. Wallin. **Chasing a Definition of "Alarm"**. Journal of Network and Systems Management, vol. 17, no 4, p. 457-481, 2009.

[68] T. Bernhard, B. Martin, S. Didier e M. Thomas. **Beyond Shannon: Characterizing Internet Traffic with Generalized Entropy Metrics**. In Proceedings of the 10th

International Conference on Passive and Active Network Measurement (PAM '09), 2009.

[69] T. Cover e J. Thomas. **Elements of Information Theory, second edition**. John Wiley and Sons, 2006.

[70] T. Chyssler, S. Nadjm-Tehrani, S. Burschka, e K. Burbeck. **Alarm Reduction and Correlation in Defence of IP Networks**. In Proceedings of the 13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE '04), 2004.

[71] T. Limmer e F. Dressler. **Survey of Event Correlation Techniques for Attack Detection in Early Warning Systems**. University of Erlangen, Dept. of Computer Science 7, Technical Report, 2008.

[72] T. Liu, H. Tseng, H. Chen, J. Hu, C. Yang e C. Yeh. **Design of Alarm Management System in Hybrid IP/Optical Networks**. International Conference on Advanced Information Networking and Applications Workshops, 2009.

[73] T. Rizzo, J. Steger, P. Pollner, I. Csabai e G. Vattay. **High quality queueing information from accelerated active network tomography**. In Proceedings of TRIDENTCOM. 2008.

[74] U. Zurutuza e R. Uribeetxeberria. **Intrusion Detection Alarm Correlation: A Survey**. In: Proceedings of the IADAT International Conference on Telecommunications and Computer Networks, 2004.

[75] W. Wang e W. Wu. **Online Detection of Network Traffic Anomalies Using Degree Distributions**. Int. J. Communications, Network And System Sciences, n. , p.177-182, 2010.

[76] Y. Bejerano e R. Rastogi. **Robust monitoring of link delays and faults in IP networks**. Networking, IEEE/ACM Transactions on, vol.14, no.5, p. 1092-1103, 2006.

[77] Y. Huang, N. Feamster e R. Teixeira. **Practical issues with using network tomography for fault diagnosis**. SIGCOMM Comput. Commun. Rev. 38, 5, 2008.

[78] Y. Tang, E. Al-Shaer e R. Boutaba. **Efficient fault diagnosis using incremental alarm correlation and active investigation for internet and overlay networks**. IEEE Transactions on Network and Service Management. v5 i1. 36-49, 2008.

[79] Y. Vardi. **Metrics useful in network tomography studies**. IEEE Signal Processing Letters 11, p. 353–355, 2004.

[80] Z. Qiuhua e Q. Yuntao. **An Event Correlation Approach Based on the Combination of IHU and Codebook**. In Computational Intelligence and Security (CIS05), vol. 3802, p. 757-763, 2005.