



Ministério da  
**Ciência e Tecnologia**



sid.inpe.br/mtc-m19/2011/02.15.17.55-TDI

# **UMA METODOLOGIA PARA CARACTERIZAÇÃO DO TRÁFEGO DE REDES DE COMPUTADORES: UMA APLICAÇÃO EM DETECÇÃO DE ANOMALIAS**

Adriana Cristina Ferrari dos Santos

Tese de Doutorado do Curso de Pós Graduação em Computação Aplicada,  
orientada pelos Drs. José Demisio Simões da Silva, e Lília de Sá Silva, aprovada em  
24 de fevereiro de 2011

URL do documento original:  
<<http://urlib.net/8JMKD3MGP7W/3973S4S>>

INPE  
São José dos Campos  
2011

## **PUBLICADO POR :**

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

Fax: (012) 3208-6919

E-mail: pubtc@sid.inpe.br

## **CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELLECTUAL DO INPE (RE/DIR-204):**

### **Presidente:**

Dr. Gerald Jean Francis Banon - Coordenação Observação da Terra (OBT)

### **Membros:**

Dr<sup>a</sup> Inez Staciari Batista - Coordenação Ciências Espaciais e Atmosféricas (CEA)

Dr<sup>a</sup> Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação

Dr<sup>a</sup> Regina Célia dos Santos Alvalá - Centro de Ciência do Sistema Terrestre (CST)

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Dr. Ralf Gielow - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Dr. Wilson Yamaguti - Coordenação Engenharia e Tecnologia Espacial (ETE)

Dr. Horácio Hideki Yanasse - Centro de Tecnologias Especiais (CTE)

### **BIBLIOTECA DIGITAL:**

Dr. Gerald Jean Francis Banon - Coordenação de Observação da Terra (OBT)

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

### **REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:**

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

### **EDITORAÇÃO ELETRÔNICA:**

Vivéca Sant'Ana Lemos - Serviço de Informação e Documentação (SID)



Ministério da  
**Ciência e Tecnologia**



sid.inpe.br/mtc-m19/2011/02.15.17.55-TDI

# **UMA METODOLOGIA PARA CARACTERIZAÇÃO DO TRÁFEGO DE REDES DE COMPUTADORES: UMA APLICAÇÃO EM DETECÇÃO DE ANOMALIAS**

Adriana Cristina Ferrari dos Santos

Tese de Doutorado do Curso de Pós Graduação em Computação Aplicada,  
orientada pelos Drs. José Demisio Simões da Silva, e Lília de Sá Silva, aprovada em  
24 de fevereiro de 2011

URL do documento original:

<<http://urlib.net/8JMKD3MGP7W/3973S4S>>

INPE  
São José dos Campos  
2011

---

Dados Internacionais de Catalogação na Publicação (CIP)

---

Santos, Adriana Cristina Ferrari dos.

Sa59m      Uma metodologia para caracterização do tráfego de redes de computadores: uma aplicação em detecção de anomalias / Adriana Cristina Ferrari dos Santos. – São José dos Campos : INPE, 2011. xxiv+169 p. ; (sid.inpe.br/mtc-m19/2011/02.15.17.55-TDI)

Tese (Doutorado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2011.

Orientadores : Drs. José Demisio Simões da Silva, e Lília de Sá Silva.

1. Tráfego de rede. 2. Clusterização. 3. Inteligência computacional. 4. Detecção de anomalias. 5. Segurança de redes. I.Título.

CDU 004.7

---

Copyright © 2011 do MCT/INPE. Nenhuma parte desta publicação pode ser reproduzida, armazenada em um sistema de recuperação, ou transmitida sob qualquer forma ou por qualquer meio, eletrônico, mecânico, fotográfico, reprográfico, de microfilmagem ou outros, sem a permissão escrita do INPE, com exceção de qualquer material fornecido especificamente com o propósito de ser entrado e executado num sistema computacional, para o uso exclusivo do leitor da obra.

Copyright © 2011 by MCT/INPE. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, microfilming, or otherwise, without written permission from INPE, with the exception of any material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use of the reader of the work.

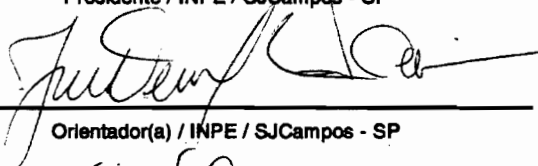
Aprovado (a) pela Banca Examinadora  
em cumprimento ao requisito exigido para  
obtenção do Título de Doutor(a) em  
Computação Aplicada

Dr. Solon Venâncio de Carvalho



Presidente / INPE / SJC Campos - SP

Dr. José Demisio Simões da Silva



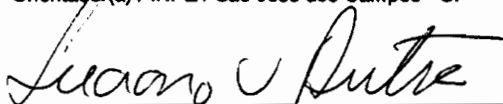
Orientador(a) / INPE / SJC Campos - SP

Dra. Lília de Sá Silva




Orientador(a) / INPE / São José dos Campos - SP

Dr. Luciano Vieira Dutra



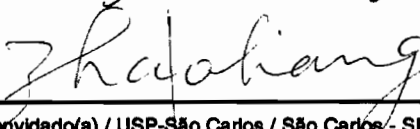
Membro da Banca / INPE / SJC Campos - SP

Dr. Edson Luiz França Senne



Membro da Banca / UNESP/GUARA / Guaratinguetá - SP

Dr. Zhao Liang



Convidado(a) / USP-São Carlos / São Carlos - SP

Dr. Adriano Mauro Cansian



Convidado(a) / UNESP/SJRP / São José do Rio Preto - SP

Aluno (a): Adriana Cristina Ferrari dos Santos

São José dos Campos, 24 de fevereiro de 2011



*“A adversidade leva alguns a serem vencidos e outros a baterem recordes.”*

*ALBERT EINSTEIN*





*A minha querida mãe Odete Cavina Ferrari por tudo: vida, fé, educação, respeito ao próximo, dedicação, carinho, apoio e amor incondicional. Te amo para sempre...*



## **AGRADECIMENTOS**

Agradeço a DEUS, por estar sempre presente e iluminar o meu caminho.

Ao Juliano Rodrigues dos Santos, meu grande amor e companheiro, pelo apoio decisivo, compreensão e incentivo na realização desse sonho. E a meu filho, Filippo Ferrari dos Santos, minha fonte de inspiração.

A meus pais Mauro e Odete, meu irmão Tomaz e cunhada/irmã de coração Larissa pelo amor, carinho em todos os momentos. Aos meus sogros Roberto e Cristina pela presença nos momentos de alegria e nos de tristeza, sempre com muito amor, alegria no coração e dedicação.

Aos meus orientadores Dr. José Demisio Simões da Silva e Dra. Lília de Sá Silva, por todas as sugestões, valiosa orientação e apoio nas pesquisas e trabalhos desenvolvidos. Sem vocês, este trabalho não seria possível. Agradeço a Deus por tê-los colocado no meu caminho.

Aos colegas do INPE, professores do Curso de Computação Aplicada (CAP) e alunos da CAP, meus agradecimentos pelas experiências compartilhadas. Ao prof. Dr. Reinaldo R. Rosa pelas explicações das técnicas de análise temporal; à Maria Cristina Peloggia de Araújo por resolver prontamente todos os nossos problemas, apoio na impressão da monografia e preparação desta apresentação e aos amigos Marinalva Dias Soares e Alexandre Soares, pela amizade e apoio técnico.

À Edenilse F. E. Orlandi, por apoiar esta pesquisa, disponibilizando o ambiente, ferramentas e dados. Também pelas orientações com a monografia e apresentação deste trabalho, exemplo de pessoa íntegra e querida por todos. Também ao Rubens Cruz Gatto, chefe da Divisão de Desenvolvimento de Sistemas de Segmento Solo (DSS) pela oportunidade e apoio. Aos colegas de trabalho da DSS, agradeço pelo incentivo e colaboração. Em especial à Milena Prado da Costa Sene, pela amizade e apoio técnico recebido nos trabalhos realizados no Laboratório de Redes da DSS.

E, naturalmente, a todas as pessoas que de alguma forma contribuíram para o cumprimento de mais esta etapa da minha vida.



## RESUMO

Os métodos de detecção de intrusos baseados em anomalias modelam o comportamento padrão do tráfego de rede e identificam anomalias como sendo os desvios do modelo de comportamento mapeado. A modelagem do comportamento do tráfego requer a análise de grandes volumes de dados para extração do conhecimento sobre as particularidades de cada ambiente de rede, considerando os serviços fornecidos, a quantidade de usuários, os acessos aos serviços efetuados ao longo do dia, entre outros. Além do tempo de processamento de grandes conjuntos, a modelagem deve ter mais cuidado e mais atenção, neste campo de pesquisa, com o consiste do número elevado de alarmes falsos gerados por este tipo de método. Para melhorar a precisão dos resultados de detecção, o comportamento da rede deve ser adequadamente mapeado e constantemente atualizado para contemplar as mudanças ocorridas no ambiente. Outro aspecto a considerar é o tamanho da base de conhecimento do modelo padrão do tráfego que, certamente, afeta o tempo de treinamento do classificador. Como contribuição nesta área, foi desenvolvida a metodologia TRAFICIN (*network Traffic Characterization on Computational Intelligence*) que descreve uma combinação de técnicas e procedimentos para caracterizar o comportamento do tráfego padrão de rede através de técnicas de inteligência computacional, que possa se tornar uma referência para atividades de detecção de anomalias em ambientes de redes operacionais. Nos testes realizados para avaliação da metodologia, a detecção de anomalias foi alcançada pela caracterização do tráfego de rede através de técnicas e abordagens adotadas para extração do conhecimento e redução da base de dados mantendo a expressividade da informação, observando taxas pequenas de alarmes falsos.



# **METHODOLOGY FOR TRAFFIC CHARACTERIZATION OF COMPUTER NETWORKS:**

## **AN APPLICATION IN ANOMALY DETECTION**

### **ABSTRACT**

The methods of intrusion detection based on anomalies model the default behavior of network traffic and identify anomalies as deviations from the behavior model mapped. The modeling of traffic behavior requires the analysis of large datasets to extract knowledge about the particularities of each network environment, considering the services provided, number of users, and access to services performed during the day, among others. Besides the processing time for large sets, the modeling must exercise greater care and concern in this search field with the high number of false alarms generated by this type of method. To improve the accuracy of the results of detection, network behavior must be properly mapped and constantly updated to include the changes in the environment. Another aspect to consider is the size of the knowledge base of the standard model of traffic which certainly affects the training time of the classifier. As a contribution in this area, we developed the methodology TRAF CIN (network Traffic Characterization on Computational INteligence) describing a combination of techniques and procedures to characterize the behavior of network traffic using techniques from computational intelligence, which can become a reference for activities to detect anomalies in network operating environments. In tests conducted to evaluate the methodology, anomaly detection was achieved by the characterization of network traffic through the clustering technique adopted for knowledge extraction and reduction of the database while retaining the expressiveness of information, observing small rates of false alarms.





## LISTA DE SIGLAS E ABREVIATURAS

TCP	Transport Control Protocol - Protocolo de Controle de Transporte
IP	Internet Protocol - Protocolo Internet
SDI	Sistemas de Detecção de Intrusos
IC	Inteligência Computacional
RNA	Redes Neurais Artificiais
MLP	Multiple-Layer Perceptron - Perceptron de Múltiplas Camadas
SVM	Support Vector Machine - Máquina de Vetor de Suporte
RBF	Neural Network Radial Basis Function - Rede neural Função de Base Radial
MaxVer	Máximo Verossimilhança
TRAFCIN	network Traffic Characterization on Computational INteligence
HTTP	Hypertext Transfer Protocol - Protocolo de Transferência de Hipertexto
DFA	Detrended Fluctuation Analysis - Análise de Flutuação Destendenciada

PDF	Probability Density Function - Função de Densidade de Probabilidade
DDoS	Distributed Denial of Service - Ataque de Negação de Serviço Distribuído
DoS	Denial of Service - Ataque de Negação de Serviço
DNS	Domain Name System - Sistema de Domínio de Nome
MKA	Mapa de Kohonen Adaptável
ICMP	Internet Control Message Protocol – Protocolo de controle de mensagem de internet
UDP	User Datagram Protocol – Protocolo de Datagrama do Usuário

## LISTA DE FIGURAS

Figura 2.1 – Gráficos das curvas em relação à curtose .....	21
Figura 3.1 – Diagrama geral do modelo ACME! reestruturado.....	43
Figura 3.2 - Arquitetura Modular do Sistema ADTRAF .....	48
Figura 3.3 – Arquitetura do Safeguard .....	49
Figura 3.4 – Falsos positivos das técnicas comparadas .....	53
Figura 4.1 - Diagrama da metodologia TRAF CIN.....	57
Figura 4.2 - Etapa de pré-processamento de dados .....	58
Figura 4.3 - Ambiente de coleta de dados reais.....	59
Figura 4.4 - Ambiente de coleta de dados anômalos .....	61
Figura 4.5 – Ciclo de atividades de coleta de tráfego anômalo.....	62
Figura 4.6 - Etapa de modelagem e mineração de dados.....	66
Figura 4.7 – Etapa de limiarização e rotulação de dados.....	72
Figura 4.8 – Etapa de detecção de anomalias .....	76
Figura 4.9 – Classificação de dados por análise espacial.....	77
Figura 4.10 – Classificação de dados por análise temporal.....	78
Figura 5.1 – Nove atributos das sessões do tráfego de rede em coordenadas paralelas .....	89
Figura 5.2 - Atributo psizeCL nas sessões do tráfego de um dia .....	91
Figura 5.3 - Atributos psizeSV nas sessões do tráfego de um dia .....	91
Figura 5.4 - Atributo pnumCL das sessões do Tráfego de um dia .....	92
Figura 5.5 - Atributo pnumSV das sessões do Tráfego de um dia .....	92
Figura 5.6 - Atributo smallPKT das sessões do tráfego de um dia.....	93
Figura 5.7 - Atributo dataDir das sessões do tráfego de um dia.....	93
Figura 5.8 - Atributo brevcCL das sessões do tráfego de um dia.....	94
Figura 5.9 – Atributo brevcSV das sessões do tráfego de um dia.....	94
Figura 5.10 – Atributo duration das sessões do tráfego de um dia .....	95
Figura 5.11 – Comportamento dos 9 atributos em um dia de tráfego .....	96
Figura 5.12 – Comportamento dos 9 atributos em um mês de tráfego .....	97
Figura 5.13 – Cálculo do PDF para os atributos psizeCL e psizeSV.....	98
Figura 5.14 – Cálculo do PDF para os atributos pnumCL e pnumSV .....	99
Figura 5.15 – Cálculo do PDF para os atributos smallPKT e dataDir.....	99
Figura 5.16 – Cálculo do PDF para os atributos brevcCL e brevcSV .....	99

Figura 5.17 – Cálculo do PDF para o atributo duration .....	100
Figura 5.18 - Representação gráfica do alfa e da curtose para o atributo "pnumCL" .....	102
Figura 5.19 - Representação gráfica do alfa e curtose para o atributo "pnumSV" .....	103
Figura 5.20 - Representação gráfica do alfa and curtose para o atributo "brecvCL" .....	103
Figura 5.21 - Representação gráfica do alfa e curtose para o atributo "brecvSV" .....	104
Figura 5.22 - Quantidade de sessões por <i>cluster</i> – Período das 0h às 07h...	107
Figura 5.23 - Quantidade de sessões por <i>cluster</i> – Período das 07h às 13h.	107
Figura 5.24 - Quantidade de sessões por <i>cluster</i> - Período das 13h às 19h..	108
Figura 5.25 - Quantidade de sessões por <i>cluster</i> - Período das 19h às 0h....	108
Figura 5.26 - Quantidade de <i>clusters</i> (Mínimos e Máximos) gerados com os parâmetros <i>ds</i> =10% e <i>sim</i> =70% .....	110
Figura 5.27 – Valores de <i>z-score</i> do atributo brecvCL - Período 00h as 07h durante um mês de captura de tráfego de rede .....	115
Figura 5.28 Valores de <i>z-score</i> do atributo brecvCL – Período das 07h às 13h durante um mês de captura de tráfego de rede .....	115
Figura 5.29 - Valores de <i>z-score</i> do atributo brecvCL – Período das 13h às 19h durante um mês de captura de tráfego de rede .....	116
Figura 5.30 - Valores de <i>z-score</i> do atributo brecvCL - Período das 19h às 0h durante um mês de captura de tráfego de rede .....	116
Figura 5.31 – Valores de <i>z-score</i> do atributo dataDir - Período 00h as 07h durante um mês de captura de tráfego de rede .....	117
Figura 5.32 Valores de <i>z-score</i> do atributo dataDir – Período das 07h às 13h durante um mês de captura de tráfego de rede .....	117
Figura 5.33 - Valores de <i>z-score</i> do atributo dataDir – Período das 13h às 19h durante um mês de captura de tráfego de rede .....	118
Figura 5.34 - Valores de <i>z-score</i> do atributo dataDir - Período das 19h às 0h durante um mês de captura de tráfego de rede .....	118
Figura 5.35 – Valores de <i>z-score</i> do atributo brecvCL - Período 00h as 07h durante dois meses de captura de tráfego de rede.....	119
Figura 5.36 Valores de <i>z-score</i> do atributo brecvCL – Período das 07h às 13h durante dois meses de captura de tráfego de rede.....	119
Figura 5.37 - Valores de <i>z-score</i> do atributo brecvCL – Período das 13h às 19h durante dois meses de captura de tráfego de rede.....	120

Figura 5.38 - Valores de <i>z-score</i> do atributo <i>brevCL</i> - Período das 19h às 0h durante dois meses de captura de tráfego de rede.....	120
Figura 5.39 – Valores de <i>z-score</i> do atributo <i>dataDir</i> - Período 00h as 07h durante dois meses de captura de tráfego de rede.....	121
Figura 5.40 Valores de <i>z-score</i> do atributo <i>dataDir</i> – Período das 07h às 13h durante dois meses de captura de tráfego de rede.....	121
Figura 5.41 - Valores de <i>z-score</i> do atributo <i>dataDir</i> – Período das 13h às 19h durante dois meses de captura de tráfego de rede.....	122
Figura 5.42 - Valores de <i>z-score</i> do atributo <i>dataDir</i> - Período das 19h às 0h durante dois meses de captura de tráfego de rede.....	122
Figura 5.43 - Arquitetura 9-10-1 da MLP .....	129
Figura 5.44 –Matriz de confusão do classificador MLP do estudo de caso 1 .	134
Figura 5.45 – Matriz de confusão do classificador MaxVer do estudo de caso 1 .....	134
Figura 5.46 – Matriz de confusão gerada para a MLP do estudo de caso 2 ..	138
Figura 5.47 – Matriz de confusão gerada para o MaxVer do estudo de caso 2 .....	138
Figura 5.48 – Matriz de Confusão do teste dos classificadores MLP e MaxVer com dados de ataque.....	140



## LISTA DE TABELAS

Tabela 2.1- Atributos típicos utilizados em detecção de anomalias. ....	10
Tabela 2.3 – Coeficientes de curtose ( $\langle k \rangle$ ).....	21
Tabela 2.4 – Classificação em função do achatamento das curvas.....	22
Tabela 2.5 – Classificação da série temporal de acordo com valores de $\alpha$ ....	25
Tabela 4.1 – “Exploits” utilizados para geração de anomalias sintéticas .....	62
Tabela 4.2 - Descrição dos atributos utilizados para representar cada sessão	63
Tabela 4.3 - Sessões aleatórias - valores reais de atributos com rótulos ( <i>class</i> ). .....	75
Tabela 5.1 – Modelos de arquivos utilizados na construção da TRAF CIN.....	84
Tabela 5.2 - Amostra de dados coletados na rede de produção.....	86
Tabela 5.3 – Amostra de tráfego anômalo gerado por um ataque do tipo DoS	88
Tabela 5.4 – Comportamento do expoente $\alpha$ nos 9 atributos do tráfego de rede .....	101
Tabela 5.5. Amostra das Séries Temporais Analisadas.....	106
Tabela 5.6. <i>Clusters</i> gerados - Parâmetros $ds=$ 10% e 70% de similaridade.	109
Tabela 5.7 – Exemplo dos dados armazenados no relatório VA.m.....	111
Tabela 5.8 – Amostra da porcentagem de sessões armazenadas em cada <i>cluster</i> .....	111
Tabela 5.9 – Os vinte maiores <i>clusters</i> do período P1 (madrugada – das 00h as 07h) .....	113
Tabela 5.10 – Resumo dos maiores <i>clusters</i> .....	114
Tabela 5.11 - Tabela com limiares <i>z-score</i> mínimos.....	124
Tabela 5.12 - Tabela com limiares <i>z-score</i> máximos .....	125
Tabela 5.13: Sessões aleatórias - valores reais de atributos classificadas pelo sistema .....	127
Tabela 5.14 – Percentual de Sessões Rotuladas.....	128
Tabela 5.15 - Subconjunto de treinamento dos classificadores com nova clusterização.....	131
Tabela 5.16 – Taxas de acerto no treinamento utilizando a base gerada no TRAF CIN.....	132
Tabela 5.17 – Índices Kappa no treinamento utilizando a base gerada no TRAF CIN.....	132
Tabela 5.18 - Subconjuntos de Testes dos classificadores.....	133

Tabela 5.19 – Taxa de Acerto dos classificadores na fase de teste do estudo de caso 1 .....	133
Tabela 5.20 – Índice Kappa dos classificadores na fase de teste do estudo de caso 1 .....	133
Tabela 5.21 - Subconjuntos de Treinamento gerado aleatoriamente.....	135
Tabela 5.22 - Taxas de acerto dos classificadores no treinamento do estudo de caso 2.....	136
Tabela 5.23 - Índices Kappa dos classificadores no treinamento do estudo de caso 2.....	136
Tabela 5.24 - Taxa de Acerto dos classificadores no teste do estudo de caso 2 .....	137
Tabela 5.25 – Índice Kappa dos classificadores no teste do estudo de caso 2 .....	137
Tabela 5.26 – Subconjunto de dados acrescido de sessões com ataques ....	139
Tabela 5.27 - Taxa de Acerto dos classificadores no teste do estudo de caso 3 .....	139
Tabela 5.28 – Índice Kappa dos classificadores no teste do estudo de caso 3 .....	139
Tabela 5.29 – Quantidade de sessões lidas em janela de 10 minutos e 1 hora .....	141
Tabela 5.30 – Sessões com ataque inseridas nos 4 períodos para análise temporal.....	142
Tabela 5.31 – Sessões com ataque detectadas pelo classificador temporal .	142



## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>1</b>
1.1.	Motivação .....	3
1.2.	Proposta de Trabalho .....	3
1.3.	Originalidade e Contribuição da Tese .....	4
1.4.	Organização do Trabalho .....	6
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>7</b>
2.1.	Tráfego de rede .....	7
2.2.	Seleção de Atributos .....	9
2.3.	Caracterização do Tráfego de Rede .....	14
2.4.	Anomalias no Tráfego de Rede .....	16
2.5.	Ataques a Redes .....	18
2.6.	Técnicas Aplicadas para Análise de Dados .....	20
2.6.1.	Curtose .....	20
2.6.2.	PDF .....	22
2.6.3.	DFA .....	23
2.6.4.	Limiarização .....	25
2.6.5.	Clusterização .....	26
2.6.6.	Rede Neural MLP .....	26
2.6.7.	SVM .....	27
2.6.8.	Rede RBF .....	29
2.6.9.	Árvore de Decisão .....	30
2.6.10.	Máxima Verossimilhança .....	31
<b>3</b>	<b>APRESENTAÇÃO DO PROBLEMA .....</b>	<b>33</b>
3.1.	Detecção de Anomalias no Tráfego de Rede .....	33
3.2.	Problemas de Detecção de Anomalias em Redes .....	36
3.3.	Caracterização do Problema .....	39
3.4.	Abordagens Existentes para Solução do Problema .....	41
<b>4</b>	<b>A METODOLOGIA TRAF CIN PROPOSTA .....</b>	<b>55</b>
4.1.	Pré-processamento dos Dados .....	58
4.1.1.	Coleta de Dados .....	59
4.1.2.	Seleção de Atributos .....	62
4.1.3.	Reconstrução de Sessões .....	64
4.2.	Modelagem e Mineração de Dados .....	65
4.2.1.	Análise Preliminar dos Dados .....	66
4.2.2.	Modelagem dos Dados .....	67
4.2.3.	Clusterização do Tráfego .....	69
4.3.	Limiarização e Rotulação de Dados .....	72
4.4.	Detecção de Anomalias .....	75
4.4.1.	Classificação por Análise Espacial do Tráfego .....	76
4.4.2.	Classificação por Análise Temporal do Tráfego .....	77

<b>5</b>	<b>RESULTADOS DE ANÁLISES .....</b>	<b>81</b>
5.1.	Cenário dos Testes.....	81
5.2.	Dados Utilizados na Análise .....	83
5.3.	Análise Estatística Preliminar.....	88
5.3.1.	Análise baseada em técnicas estatísticas .....	88
5.3.2.	Análise estatística para um dia de tráfego.....	89
5.3.3.	Análise estatística para um mês de tráfego.....	97
5.3.4.	Análise baseada em séries temporais.....	98
5.3.5.	Análise baseada em PDF .....	98
5.3.6.	Análise baseada em DFA.....	100
5.3.7.	Análise baseada em DFA e Curtose para Caracterização do Comportamento padrão do Tráfego .....	102
5.4.	Resultados da clusterização .....	105
5.5.	Caracterização dos Dados.....	112
5.5.1.	Limiarização dos Dados .....	114
5.6.	Detecção de Anomalias no tráfego de rede.....	128
5.6.1.	Classificação baseada em Análise Espacial dos dados .....	128
5.6.2.	Classificação baseada em Análise Temporal dos dados .....	141
<b>6</b>	<b>CONCLUSÃO .....</b>	<b>143</b>
	<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>151</b>
	<b>ANEXO A – PROTOCOLOS TCP/IP .....</b>	<b>161</b>

## 1 INTRODUÇÃO

As redes de computadores TCP/IP (*Transmission Control Protocol / Internet Protocol*) proporcionam muitos benefícios nas comunicações em rede, incluindo serviço de correio eletrônico, acesso a aplicações Web, exposição de produtos e serviços, transferência de arquivos, compartilhamento de dados e recursos computacionais distribuídos, dentre outros. Com o uso crescente das redes TCP/IP, as empresas agregaram mais valor aos seus produtos e serviços a partir da interação *on-line* com seus clientes e fornecedores. Para mais detalhes sobre os protocolos TCP/IP consultar anexo A.

Apesar dos esforços em sua implantação, as redes podem apresentar vulnerabilidades expondo-se às ações ilegítimas de atacantes, sendo, atualmente, um grande problema para as organizações que têm se tornado cada vez mais dependente das tecnologias de rede. Os serviços disponibilizados pelas organizações em redes TCP/IP, quando não disponíveis, podem gerar impactos negativos em seus negócios. Este fato torna evidente a necessidade de se estabelecer monitoramento e controle contínuos sobre o comportamento do tráfego de rede, identificando os problemas em tempo hábil para aplicação de contramedidas imediatas que possam evitar ou reduzir os prejuízos financeiros e manter a imagem de uma empresa usuária de serviços de redes.

Garantir disponibilidade, segurança e integridade de dados e sistemas nas redes de computadores é um desafio constante. De modo a proteger os dados, serviços e recursos computacionais das redes, diversas camadas de segurança têm sido propostas ao longo dos anos, tais como: o uso de sistemas antivírus e *anti-spyware*, *firewalls*, acesso remoto via VPN (Virtual Private Network), implantação de *honeynets*, ferramentas conhecidas como 3A (Autenticação, Autorização e Administração de usuários) e Sistemas de Detecção de Intrusos (SDI).

Os SDI mais conhecidos são:

- Snort (FOSTER et al., 2003) - sistema gratuito e leve (capaz de executar a análise de tráfego em tempo real);
- Shadow (KERBY, 2011) (desenvolvido em 1994 por Guy para fornecer uma solução de código aberto de monitoração de rede às empresas, facilmente implementados em um ambiente grande e distribuído);
- Cisco Secure SDI (CISCO, 2011) (aplicativo projetado pela Cisco para proteger eficientemente sua infra-estrutura de dados e de informações);
- Dragon (ENTERASYS, 2011) (desenvolvido pela Enterasys para verificar evidências de atividades maliciosas na rede), entre outros exemplos.

Dentre os SDI gratuitos destacam-se: o Snort, que usa a arquitetura modular por meio de *plug-ins* e o Shadow, com uma arquitetura dividida em sensor e analisador.

Os SDI podem variar quanto aos métodos de detecção, em duas principais categorias: detecção baseada em assinaturas e detecção baseada em anomalias (TSANG et al., 2007).

O método de detecção de intrusos baseado em assinaturas consiste na modelagem de padrões de ataques conhecidos e na pesquisa destes em conjuntos de dados.

Sistemas de detecção de intrusos baseados em anomalias modelam o comportamento padrão de processos de usuários, de sistemas ou do tráfego de rede e desvios deste modelo de comportamento representam anomalias, comportamento não usual ou eventos ilegítimos.

Contudo, compreender e detectar anomalias em redes de modo satisfatório ainda é um problema a ser solucionado. O processamento de SDI baseado em anomalias é extremamente complexo, pois requer análise de conjuntos de dados de grande volume e conhecimento aprofundado das características do comportamento padrão do objeto observado (APILETTI et al., 2009).

Para a construção de sistemas que abstraíam conhecimento do tráfego de rede e que proporcionem segurança efetiva às redes de computadores devem ser utilizadas técnicas que reduzam o esforço humano necessário ao controle e análise de grandes volumes de dados. Inteligência Computacional (IC) e Mineração de Dados (MD) têm sido empregadas no desenvolvimento de ferramentas de análise de tráfego de rede (FORTUNA et al., 2007; SINGH, RAJAMENAKSHI, 2009).

Neste trabalho, utilizou-se o método de detecção baseado em anomalias para análise do comportamento padrão do tráfego de rede, o qual requer a inspeção do tráfego padrão para o entendimento dos tipos de anomalias existentes.

### **1.1. Motivação**

Desenvolver sistemas eficientes de detecção de intrusos na rede é um grande desafio, devido, principalmente, à complexidade do ambiente de rede, em termos de:

- Quantidade e natureza dos serviços oferecidos;
- Quantidade de usuários e sistemas e da mudança no comportamento destes afetando a configuração da rede;
- Atualização constante da base de conhecimento do SDI ;
- O grande volume de dados de tráfego a analisar.

A principal motivação para este trabalho é desenvolver uma metodologia para caracterização do tráfego de redes de computadores que proporcione melhorias no processo de detecção de anomalias no tráfego, buscando resolver os problemas de: análise de grande volume de dados e geração de altas taxas de falsos positivos.

### **1.2. Proposta de Trabalho**

A metodologia TRAFICIN (*network Traffic Characterization on Computational Intelligence*) proposta neste trabalho, caracteriza o tráfego padrão de rede TCP/IP gerado a partir de serviços Web (HTTP), onde cada registro de sessão

é descrito por nove atributos, para treinamento satisfatório de classificadores para detecção de anomalias de modo rápido e preciso.

Trata-se de uma metodologia de análise, processamento e extração da informação de tráfego de redes de computadores, utilizando técnicas de Inteligência Computacional, em uma abordagem de mineração de dados, visando identificar padrões de anomalias nos dados espaço-temporais do fluxo de rede observado, adquirido em grandes volumes de dados. A característica espacial da abordagem leva em consideração o espaço das variáveis observadas e a característica temporal está relacionada com o comportamento diferenciado do tráfego de rede em instantes de tempo diferentes do tráfego da rede.

A metodologia TRAF CIN caracteriza o comportamento padrão do tráfego e visa se tornar uma referência para atividades de detecção de anomalias em ambientes de redes operacionais. A idéia da TRAF CIN é:

- Criar modelos do comportamento padrão do tráfego de rede compactos e significativos (precisos) e adaptáveis dos dados espaços-temporais do fluxo de rede observado (eliminando redundâncias);
- A partir do modelo do tráfego caracterizado treinar adequadamente os SDI para detectar anomalias com baixa taxa de alarmes falsos.

As amostras de dados estudadas foram coletadas em dias e instantes de tempo diferentes e correspondem a um grande volume de dados de sessões do tráfego Web. Mineração dos dados por clusterização foi aplicada nas sessões do tráfego, no intuito de caracterizar o comportamento padrão da rede, levando em consideração a sua variação no tempo.

A metodologia desenvolvida para detecção de anomalias na rede consiste em quatro etapas consecutivas, sendo: pré-processamento, modelagem e mineração de dados, limiarização e rotulação dos dados e detecção de anomalias, que serão descritas no Capítulo 4

### **1.3. Originalidade e Contribuição da Tese**

A originalidade deste trabalho na área de caracterização do tráfego de rede de computadores para detecção de anomalias é o desenvolvimento da metodologia TRAF CIN baseada na combinação de técnicas de Inteligência Computacional e estatísticas para caracterizar o comportamento padrão do tráfego de rede ao longo do tempo, gerando um modelo compacto, preciso e adaptável do tráfego de rede. As técnicas aplicadas incluem:

- Análise estatística convencional dos atributos das sessões, por meio dos cálculos de média, desvio padrão, variância e curtose para análise de padronização do tráfego;
- Análise temporal dos dados, utilizando as técnicas de DFA (Detrended Fluctuation Analysis) e PDF (Probability Density Function) para investigar se os dados analisados como séries temporais eram passíveis de previsão;
- Heurística baseada em clusterização para redução de dados do tráfego histórico sem perda de informações relevantes para a caracterização, incluindo:
  - Técnica de IC “Mapa de Kohonen Adaptável” para clusterização de dados, baseada em vetores representantes do tráfego;
  - Seleção dos *clusters* mais populosos de sessões para representar o comportamento padrão do tráfego;
- Heurística para reorganização dos dados do tráfego histórico por dia da semana e período do dia para melhor extração do conhecimento implícito;
- Limiarização temporal baseada em *z-score* para definir limites de comportamento padrão do tráfego;
- Sistema baseado em regras de produção para rotular o tráfego histórico como padrão ou anômalo;
- Heurística baseada em clusterização de dados do tráfego rotulado como padrão (0) para equilibrar a porcentagem de sessões rotuladas das três classes, proporcionando:
  - Menor tempo de treinamento do módulo de detecção do SDI, pela redução da base de dados do treinamento do classificador;

- Melhor precisão dos resultados, evitando saturação dos classificadores;
- Técnicas de Inteligência Computacional, tais como, os classificadores MaxVer, rede neural MLP, rede neural RBF, SVM e Árvore de Decisão para classificação de sessões do tráfego como padrão ou anômala, incluindo:
  - Classificação baseada em análise espacial de cada valor de atributo das sessões correntes;
  - Classificação baseada em análise temporal dos dados pela frequência de ocorrência de sessões com a mesma característica, através de uma janela de captura de 10 minutos de dados, para detecção de ataques de Probing ou DoS.

A principal contribuição deste trabalho é a metodologia para a criação de modelos do comportamento padrão do tráfego de rede compactos e significativos, a partir de análise espaço-temporal de grandes volumes de dados do tráfego, pois pelas pesquisas realizadas ainda não havia sido aplicada.

#### **1.4. Organização do Trabalho**

O Capítulo 2 apresenta o referencial teórico deste trabalho, incluindo tráfego de rede, seleção de atributos, caracterização do tráfego, anomalias de rede e algumas das técnicas Estáticas, Análise Temporal e Inteligência Computacional aplicados na metodologia TRAFICIN. O Capítulo 3 fornece as premissas das dificuldades encontradas na área de detecção de anomalia, apresenta a caracterização do problema a ser solucionado pela metodologia TRAFICIN, bem como aborda diversas aplicações propostas por diferentes autores na área de detecção de anomalias em rede. O Capítulo 4 descreve as quatro etapas da metodologia TRAFICIN para caracterização do tráfego padrão de rede. O Capítulo 5 apresenta os resultados obtidos com a metodologia TRAFICIN na análise espacial e temporal dos dados de rede, especificando os cenários dos testes (*hardwares* e *softwares* utilizados). Finalmente, o Capítulo 5 é destinado às conclusões do trabalho e os próximos passos a serem desenvolvidos.



## 2 FUNDAMENTAÇÃO TEÓRICA

Os principais aspectos teóricos e técnicos relativos à metodologia TRAFKIN desenvolvida para caracterização do comportamento padrão do tráfego e posterior detecção de anomalias em redes de computadores são descritos a seguir.

### 2.1. Tráfego de rede

O tráfego de uma Rede TCP/IP consiste de um amplo e multivariado conjunto de dados de pacotes de rede gerados durante as comunicações entre os *hosts*. Para analisá-los, primeiro é preciso coletá-los.

O processo de coleta do tráfego de rede consiste na captura contínua destes pacotes de rede contendo os dados brutos (em formato hexadecimal), através de uma estação servidora, denominada “sensor do SDI”, com capacidade de disco para armazenamento de um grande volume de dados.

Dependendo do objetivo da análise a ser realizada, pode-se ter um sensor do SDI ou posicionado fora ou dentro dos limites de proteção do *firewall* ou dois sensores, um dentro e outro fora dos limites de proteção do *firewall* (BOUZIDA; MARGIN, 2008).

Ao posicionar o sensor fora dos limites protegidos pelo *firewall*, todo o tráfego direcionado à rede monitorada vindo da Internet é coletado pelo sensor (CHAVES, 2002; SILVA, 2007). Dentro dos limites de proteção do *firewall*, o sensor é capaz de observar o tráfego externo permitido pelo *firewall* em direção à rede monitorada e o tráfego desta rede para as redes externas passando pelo *firewall*.

Um sensor de SDI pode ser posicionado dentro e outro fora dos limites de proteção do *firewall*. Deste modo, de acordo com Northcutt et al. (2007) os benefícios de ambas as técnicas podem ser obtidos, inclusive as regras do *firewall* podem ser validadas.

Os dados dos pacotes de rede coletados à partir do sensor de SDI devem ser remontados em sessões do tráfego, de modo que possam fornecer informação significativa sobre as comunicações ocorridas (BURBECK; NADJIM, 2005).

Uma sessão de rede TCP/IP corresponde a qualquer seqüência de pacotes, que caracterize a troca de informações entre dois endereços IP, durante um determinado intervalo de tempo, relacionada a um determinado serviço de rede, que tenha informação de início, meio e fim, mesmo que toda comunicação esteja contida em um único pacote (CHAVES, 2002). Sessões de diferentes aplicações possuem comportamentos próprios.

Uma sessão de rede pode ser unicamente caracterizada pela combinação de diferentes características ou atributos. Os atributos de sessão contribuem para analisar o comportamento do tráfego de rede de diferentes formas. Alguns tipos de análises do tráfego são realizadas através de informações localizadas na parte de conteúdo (*payload*) dos pacotes, enquanto outras podem ser realizadas a partir da observação dos campos no cabeçalho dos pacotes.

Um atributo de sessão pode ser primitivo, ou seja, um valor extraído diretamente de um campo do cabeçalho do pacote ou um atributo derivado, construído a partir do processamento de atributos primitivos (MUKKAMALA; SUNG 2003). Compreendem os atributos primitivos aqueles extraídos do cabeçalho IP: endereço IP de origem e de destino, protocolo de transporte utilizado, tamanho total do pacote em bytes; e aqueles extraídos do cabeçalho TCP: portas de origem e de destino e *flags*. Os atributos derivados são obtidos do processamento dos atributos primitivos e correspondem à informação semanticamente mais forte para a representação do tráfego (FURLONG, 2007). Como exemplo de atributos derivados tem-se a duração da sessão, que corresponde à diferença entre o tempo de captura do último e do primeiro pacote da sessão.

Do grande número de atributos que pode ser monitorado para o propósito de mapear o comportamento do tráfego de rede, é importante definir quais são realmente úteis para análise. Este assunto é abordado na subseção seguinte.

## 2.2. Seleção de Atributos

O tráfego de uma rede pode ser analisado a partir de diferentes níveis de abstração, envolvendo dados de sessão, fluxo, pacotes e *bytes* (DAINOTTI et al., 2006; LIU, HUANG, 2010). O que define o nível de abstração da análise do tráfego são as características ou atributos selecionados para análise.

A seleção de atributos do tráfego de uma rede é uma questão relevante para a caracterização satisfatória do comportamento padrão do tráfego de rede, porque a qualidade e a quantidade de atributos afeta a precisão do modelo de caracterização do comportamento do tráfego e desempenho do processo de detecção de anomalias (LIU e LI, 2009).

Bons atributos deveriam prover informação útil para classificar o tráfego como padrão ou não, gerando alta taxa de acertos com baixa taxa de alarme falso. Recomenda-se (MUKKAMALA; SUNG, 2003) não selecionar muitos atributos que servirão de entrada para os classificadores de dados de tráfego, pois podem invalidar os resultados de detecção.

O site do KDD-CUP 1999 (STOLFO et al., 1999) disponibiliza uma listagem completa de 41 atributos para uso público e encontram-se divididos em três classes: (1) atributos básicos de conexões TCP individuais, (2) atributos de conteúdo dentro de uma conexão sugerida através de conhecimento do domínio e (3) atributos do tráfego calculados em uma janela de tempo de dois segundos.

Uma amostra destes atributos é apresentada na Tabela 2.1.

Tabela 2.1- Atributos típicos utilizados em detecção de anomalias.

Nome do Atributo	Descrição	Classe
duration	Duração da conexão	1
Src_ip	Endereço IP do iniciador da conexão	1
Src_bytes	Número de bytes enviados pelo iniciador	1
num_root	Número de acessos do “root”	2
Num_access_files	Número de operações de controle de acesso à arquivos	2
Num_shells	Número de prompt de shell	2
service	Serviço acessado (por porta): http, ftp, telnet	1
count	Número de conexões para o mesmo <i>host</i> na conexão corrente nos últimos dois segundos	3
Serror_rate	% de conexões com erros “SYN”	3
Same_srv_rate	% de conexões de mesmo serviço	3

Fonte: Stolfo et al. (1999).

Na maioria dos sistemas que analisam o comportamento do tráfego de rede, os atributos são escolhidos manualmente, com base na análise de tipos de ataques, análise estatística e visualização de dados. De acordo com Mukkamala e Sung (2003), devido à falta de um modelo analítico, pode-se apenas pesquisar e determinar a importância relativa das variáveis de entrada (atributos) dos SDI por meio de métodos empíricos. Uma análise completa requisitaria a verificação de todas as possibilidades, ou seja, tomando dois atributos por vez para analisar sua dependência ou correlação, em seguida, tomando 3 atributos por vez e assim por diante. Isto, entretanto, é inviável (requer  $2n$  experimentos).

No trabalho de Mukkamala e Sung (2003) é utilizada a técnica de aprendizagem de máquina para a seleção dos atributos. Os atributos são removidos um por vez e, para cada conjunto restante de atributos, é avaliado o desempenho do modelo de detecção de intrusão utilizando SVMs (*Support Vector Machines*).

Wei e Sylvain (2010) desenvolveram um sistema que extrai atributos importantes do tráfego de rede para detecção de ataques DDoS em redes de

computadores. A partir da coleta de um conjunto grande de tráfego de ataque DDoS, estes autores utilizam métodos estatísticos (PINA, 2011) para classificar os 41 atributos do tráfego de rede disponibilizados publicamente no KDD-CUP 1999. Redes Bayesianas e árvore de decisão (algoritmo C4.5) são utilizadas para detectar ataques DDoS e também decidir quantos atributos são apropriados para a detecção. Os resultados empíricos mostram que com apenas 9 atributos (count – número de ligações para o mesmo *host* nos últimos dois segundos, *dst\_host\_count*, *dst\_host\_srv\_diff\_host\_rate*, *dst\_host\_srv\_count*, *src\_bytes*, *srv\_count*, *srv\_diff\_host\_rate*, *dst\_host\_srv\_error\_rate*, *service*), a capacidade de detecção se compara com a detecção usando os 41 atributos, nos dois classificadores testados.

Uma nova abordagem para seleção de atributos é apresentada por Liu e Li (2009). Para a classificação do fluxo de rede automaticamente é empregado o método AutoClass baseado na classificação bayesiana, explorando os diferentes atributos estatísticos. Após o processo de aprendizagem das classes “anômala” ou “normal”, o algoritmo pode encontrar o conjunto mais apropriado de atributos, em um processo repetido, como segue: primeiro um conjunto de atributos são selecionados. Aplica-se uma métrica para estimar os atributos e aquele que obtiver o melhor resultado (estiver mais próximo das classes de aprendizagem definidas pela métrica) é colocado em uma lista de atributos. Em seguida, um novo atributo é testado e se tiver melhor resultado substitui o anterior. Este processo é repetido até que nenhuma melhoria seja alcançada. Então, os atributos finais são considerados como o melhor conjunto de atributos.

Os resultados do método AutoClass representam um esforço de aplicação de técnicas de aprendizagem de máquina automatizáveis para melhorar o processo de seleção de atributos e conseqüentemente o processo de detecção de intrusos.

No trabalho de Silva (2007), por exemplo, foram selecionados 9 atributos de sessões: tamanho médio dos pacotes recebidos pelo cliente, tamanho médio dos pacotes recebidos pelo servidor, número de pacotes recebidos pelo,

número de pacotes recebidos pelo servidor, porcentagem de pacotes pequenos, direção do tráfego, total de dados recebidos pelo cliente, total de dados recebidos pelo servidor; duração da sessão.

A aplicação para detecção de ataques à rede desenvolvida por Silva et al. (2004), utiliza nos testes 41 atributos fornecidos pelo KDDCup1999 (LEE; STOLFO, 1998) e quatro redes neurais para os testes. Os atributos são os mesmos considerados por Mukkamala e Sung (2003) incluindo: duração da conexão (em segundos), tipo de protocolo de transporte usado na comunicação (TCP, UDP...), tipo de serviço (HTTP, FTP, TELNET...), número de *bytes* enviados da origem para o destino, número de bytes enviados do destino para a origem, status da conexão (normal ou erro), *land* (1 se a conexão é de/para o mesmo *host*, 0 caso contrário), número de fragmentos errados e número de pacotes com *flag* URG ativada. Os principais tipos de ataques considerados nos testes foram DoS e Probing.

O sistema denominado MINDS (*Minnesota Intrusion Detection System*) (MINDS, 2011), desenvolvido pela Universidade de Minnesota, utiliza a técnica de detecção de outliers para identificar intrusões em rede, por meio de atributos, tais como: endereços IP de origem e destino, portas de origem e destino, protocolo, *flags*, número de *bytes* e número de pacotes, número de fluxos para endereços IP com destino único e proveniente da mesma origem dentro da rede nos últimos *t* segundos.

No trabalho de Muraleedharan (2008), é apresentado uma análise do comportamento do tráfego de rede utilizando 10 atributos de fluxo para identificar anomalias nas atividades de *host* e rede, tais como: tempo de início do fluxo, tempo final do fluxo, duração do fluxo, protocolo da camada de transporte, IP de origem, IP de destino, porta de origem, porta de destino, número de pacotes em um fluxo, número de *bytes* no fluxo e *flags*. Estes atributos foram selecionados através de estudo estatístico do tráfego de forma empírica.

O SDI NETMINE desenvolvido por Apiletti et al. (2009) realiza análise de tráfego de rede por meio de técnicas de mineração de dados (*K-means*) para

caracterização dos dados de tráfego e detecção de anomalias através de 5 atributos: IP de origem, IP de destino, porta de origem, porta de destino, número de pacotes do fluxo. Estes atributos são seleccionados empiricamente. NETMINE executa (i) análise do fluxo on-line para reconstruir as sessões do tráfego e filtrá-las, análise de refinamento (ii) para descobrir as relações entre os dados capturados, e (iii) a regra de classificação em diferentes grupos semânticos.

O SDI desenvolvido por Liu e Huang (2010), baseia-se na análise do tráfego de dados de rede, tais como pacote, volume e sessões. No modelo de análise de volume do tráfego, os autores referem-se a quantidade de bytes de dados que foram transmitidos na rede em um determinado período de tempo. Os valores podem indicar o status da rede ocupada e revelar a necessidade de largura de banda da rede. Depois de isolar o tráfego de dados pelo seu tipo de serviço (HTTP, P2P, etc.), aplica-se a análise de volume no tráfego de dados. A partir do resultado da análise, sabe-se a variação do tráfego de cada tipo de aplicações. Ao traçar as curvas da análise em conjunto, é possível descobrir a diferença de variação de volume para cada tipo de serviço. Na análise de fluxo de rede, os autores utilizam 5 atributos seleccionados de forma empírica: IP de origem, IP de destino, porta de origem, porta de destino e tipo de protocolo.

De acordo com a pesquisa realizada recentemente, observou-se que os autores seleccionam diferentes atributos para o desenvolvimento de aplicações e métodos de detecção de anomalias. Alguns dos atributos encontrados com maior frequência na literatura são listados na Tabela 2.2 a seguir.

TABELA 2.2 - Atributos típicos utilizados em detecção de anomalias

Atributos	Descrição
serviço	Serviços acessados (por porta): HTTP, FTP, Telnet
duração	Duração da conexão
src_ip	Endereço IP do iniciador da conexão
dst_ip	Endereço IP do <i>host</i>
src_bytes	Número de bytes enviados pelo iniciador
dst_bytes	Número de bytes enviados pelo <i>host</i>
protocolo	TCP, UDP, ICMP
num_conn	Número de conexões abertas
tcp_flags	Flags TCP (SYN, ACK, RST, ...)

### 2.3. Caracterização do Tráfego de Rede

Neste trabalho, o processo de caracterização do comportamento do tráfego de rede envolve a análise das sessões do tráfego por meio de seus atributos e visa encontrar e mapear padrões nos conjuntos de dados analisados. Através da comparação do tráfego corrente (perfil atual) com o modelo do comportamento do tráfego histórico de rede caracterizado (perfil armazenado), utilizando algum método de pontuação ou grau de normalidade, é possível detectar desvios no perfil atual indicando a presença de anomalias no tráfego.

A caracterização do comportamento do tráfego de rede requer a investigação dos dados coletados ao longo do tempo, observando o perfil de acesso e uso dos serviços e as mudanças de comportamento no fluxo de dados analisado.

Dois tipos de análise podem ser conduzidos para acompanhamento do comportamento do tráfego de rede em busca de definir seu comportamento padrão para detectar anomalias (SILVA, 2007):

- Observação dos conjuntos de amostras de dados de diferentes tamanhos, sem considerar a correlação dos dados ao longo do tempo;



- Observação dos conjuntos de amostras de dados de diferentes tamanhos em diferentes períodos, por meio de análise temporal dos dados.

Dos vários trabalhos encontrados na literatura sobre caracterização de tráfego de rede, a maioria (DAINOTTI et al., 2006; LIU, CHEN, 2010; MUELDER et al. 2005; TERDIK, GYIRES, 2009; dentre outros); aplica técnicas estatísticas para criar seu modelo de comportamento do tráfego.

No trabalho de Dainotti et al. (2006), por exemplo, foram criados uma metodologia e um *software* para caracterizar estatisticamente o tráfego de rede no nível de pacotes, com o objetivo de encontrar invariantes temporal e espacial em aplicações baseadas em TCP, tais como, HTTP e SMTP. Técnicas como distribuição de estimativa, ajustes estatísticos, medidas de discrepância e Maximização de Expectativa (EM) foram aplicadas para análise de distribuição dos dados. Nos resultados, os autores mostraram a aplicabilidade da metodologia em nível de pacotes para caracterizar o tráfego e identificar as propriedades invariantes temporais e espaciais, quando um conjunto de amostras estatísticas suficientemente relevantes é escolhido.

O uso da técnica de visualização para caracterizar ataques, em particular, varreduras, também tem sido explorada. Um trabalho nesta linha é apresentado por Muelder et al. (2005), no qual uma metodologia para caracterização e visualização de tráfego de rede, objetivando encontrar traços de varredura, é desenvolvida. Para análise das amostras de tráfego coletadas foi criado um conjunto de métricas derivadas dos dados coletados para extração do conhecimento destes, as quais fornecem uma visão gráfica relativamente intuitiva dos dados. A técnica de Wavelet foi utilizada para gerar uma visão global das relações dos dados coletados, por serem relativamente resistentes a deslocamentos de fase e ruído, ou seja, padrões semelhantes resultarão em escalogramas wavelet semelhantes, mesmo que os padrões estejam ligeiramente deslocados, ou partes do tráfego estejam faltando.

Uma pesquisa recente no campo de análise do tráfego de rede é proposta por Kundu et al., (2009). Os autores utilizam a técnica de Poisson e de Parzen

baseada na função de Kernel Gaussiana para definir a natureza dos fluxos de tráfego real para uma estimativa precisa do tráfego.

## **2.4. Anomalias no Tráfego de Rede**

Anomalias no tráfego de rede são caracterizadas por alterações anormais e significativas nos padrões de atividade da rede, interrompendo o comportamento característico do tráfego (ALVARADO et al., 2009)

É importante analisar as anomalias, mesmo que estas sejam não-intencionais ou não-maliciosas, pois podem provocar congestão na rede e esgotar recursos de memória e processamento de roteadores e outros elementos ativos, não produzindo grande impacto na rede, mas afetando o uso dos serviços pelo cliente.

As causas e formas das anomalias na rede podem variar significativamente, desde abusos à rede, tais como ataques de negação de serviços, a falhas nos equipamentos, por exemplo, configurações incorretas do roteador.

Inclusive, uma anomalia pode ser um ataque antigo que foi alterado de forma a evitar a detecção, ou pode ser uma forma completamente nova de ataque (CELENK et al., 2010). A diversidade de causas e formas de anomalias na rede torna a tarefa de detectá-las de forma rápida e precisa um grande desafio para os profissionais de segurança.

Conforme mencionado por Caswell et al. (2003), um diagnóstico completo de anomalias deveria ser capaz de detectar uma variedade de anomalias de diferentes estruturas, distinguindo-se entre diferentes tipos de anomalias e anomalias de mesmo grupo, com características similares.

De acordo com Patcha e Park (2007), o processo de detecção de anomalias é complexo por vários motivos:

- Identificar anomalias requer uma sofisticada infra-estrutura de monitoração;

- Muitos ISPs (Internet Service Provider) coletam somente dados de medição simples do tráfego, por exemplo, coleta de volume médio de tráfego através do protocolo SNMP. Existem ISPs que coletam dados de contagem de fluxos, o que requer muito processamento de dados;
- Não existem ferramentas rápidas o suficiente para processar os dados de medições de modo a detectar anomalias em tempo real;
- A natureza do tráfego de rede é multidimensional e ruidosa, o que torna difícil extrair informações significativas sobre anomalias seja qual for o tipo de estatística do tráfego.

Conforme definido por Patcha e Park (2007), as anomalias no tráfego de rede podem ser classificadas em quatro principais tipos: anomalias de operação da rede, anomalias “*flash crowd*” (tempestades), anomalias de medição e ataques à rede.

As anomalias de operação da rede incluem eventos de falhas, tais como interrupção de funcionamento de elementos ativos ou eventos que geram significativas mudanças no comportamento da rede, tais como a implantação de novos equipamentos, configuração inadequada temporária de dispositivos.

As anomalias do tipo “*Flash Crowd*” são, em geral, decorrentes de disponibilização de uma nova versão de *software* ou acesso a um *site* com novidades. Um comportamento *flash crowd* é observado através de um rápido crescimento nos fluxos de tráfego de um determinado tipo (por exemplo, fluxos FTP ou HTTP) ou para um destino bem conhecido com uma redução gradual com o tempo.

As anomalias de medição não ocorrem devido ao uso abusivo da rede, mas por problemas com a infra-estrutura de coleta de dados de medição, que incluem perda de dados do fluxo devido à sobrecarga do roteador.

Ataques a redes de computadores compreendem um “conjunto de ações ilícitas que tentam comprometer a integridade, confidencialidade, ou disponibilidade de recursos na rede”, independente do sucesso ou não destas ações

(PATCHA; PARK, 2007). Regras de privacidade podem ser quebradas devido a um ataque, comprometendo a confidencialidade da informação. Informações podem ser alteradas, modificando a integridade dos dados. E a infra-estrutura de rede pode tornar-se indisponível e não confiável, afetando a disponibilidade do recurso.

Muitos trabalhos têm sido desenvolvidos na linha de caracterização e detecção de anomalias em redes de computadores com diferentes abordagens. O trabalho de Alvarado et al. (2009), aplica o método de elementos restantes (MRE) para detectar anomalias no tráfego de rede baseadas na caracterização dos atributos do tráfego através de uma medida de incerteza proporcional. De acordo com os autores, o MRE tem a funcionalidade e desempenho para detecção de comportamentos anormais e pode servir como base para novas gerações de SDI.

O trabalho de Shon e Moon (2007) aplica as seguintes técnicas em sua abordagem de caracterização e classificação de tráfego de rede: SOFM (*Self-Organized Feature Map*) para caracterizar o perfil normal dos pacotes de rede e aprendizagem SVM para classificar os dados.

## **2.5. Ataques a Redes**

Ataques a redes são considerados uma anomalia ou utilização abusiva na qual o comportamento anormal é aplicado no tráfego de rede. Identificar um ataque é detectar desvios no tráfego padrão de referência da rede, pré-definido (LI; LEE, 2005).

Ataques são distintos de outros tipos de anomalias por não serem detectados através de medições de taxas de bits ou de pacotes, mas por medições da contagem de fluxos (sessões) de rede, a partir de informações de pares de portas e endereços lógicos de origem e destino das conexões.

Quanto a seus objetivos, os ataques mais freqüentes a redes de computadores podem ser classificados como DoS, Probing, U2R (User to Root) e R2L (Remote to Local) (SILVA, 2007).

Estes ataques podem ser aplicados no local do alvo, com conta de usuário autorizado, ou lançados remotamente, através de uma conexão de rede, sem qualquer conta de usuário ou acesso privilegiado ao sistema alvo, mas utilizando apenas o acesso público concedido pelo sistema. Outra estratégia utilizada com frequência pelos atacantes consiste em disparar um ataque utilizando uma conta de usuário sem privilégio para ganhar acesso inicial ao sistema. Em seguida, o atacante utiliza uma conta de usuário autorizado para tentar elevar seus privilégios e obter controle completo do alvo.

Os ataques DoS tentam reduzir o desempenho ou interromper o funcionamento de sistemas e serviços de rede. O principal objetivo deste tipo de ataque é tornar inoperante um serviço ou interromper a atividade de uma estação servidora ou equipamento ligado em rede. Exemplos de ataques desta categoria são: Apache2, Back, Land, Mail bomb, SYN Flood, Ping of death, Process table, Smurf, Syslogd, Teardrop e Udpstorm (LI; LEE, 2005).

Ataques de negação de serviço distribuído (*Distributed DoS - DDoS*): são inundações de ataques DoS onde os atacantes utilizam vários computadores para lançar o ataque e mais rapidamente sobrecarregar um determinado sistema alvo. Neste tipo de ataque é realizada uma sobrecarga ou inundação de pacotes contra um determinado serviço, *host* ou rede, gerando muitas vezes uma quantidade de dados global maior que a rede ou *host* pode suportar, tornando a rede ou serviços instáveis e conseqüentemente prejudicando o seu desempenho. Estes ataques possuem uma estrutura previamente montada, onde diversas máquinas lançam um ataque sobre uma vítima. Esses ataques fazem uso de várias “máquinas zumbis” para que o número de requisições de conexão ao servidor seja muito maior. São ataques mais eficientes e complexos, e mais difíceis de se detectar.

Ataques Probing realizam varredura de uma rede ou de sistemas-alvo, enviando diferentes tipos de pacotes de rede, em busca de vulnerabilidades e informações de interesse para planejamento de ataques. As respostas recebidas destes sistemas são utilizadas pelo atacante para aprender sobre as características da rede e dos sistemas, incluindo a topologia da rede, os *hosts*

ativos e suas respectivas configurações de *software*, incluindo sistema operacional, software do servidor e versões de aplicativos. Ataques desta natureza não penetram ou comprometem os sistemas. Alguns exemplos de ataques desta categoria são: *Ipsweep*, *Mscan*, *Nmap*, *Saint*, *Satan*.

Os ataques de penetração conhecidos como ataques R2L e U2R realizam aquisição ou alteração não autorizada dos privilégios, recursos ou dados do sistema, violando as propriedades de integridade e controle dos recursos e dados. Com estes ataques, pode-se ganhar controle de um sistema ao explorar uma variedade de falhas de *software*.

Um ataque R2L (*Remote to Local*) ocorre quando um usuário realiza um acesso remoto não autorizado a uma máquina e consegue privilégios de usuário local. Neste tipo de ataque, são enviados pacotes para uma máquina na rede na qual o atacante não tem conta e, em seguida, são exploradas algumas vulnerabilidades desta máquina que permitem a obtenção de acesso local como se fosse um usuário daquela máquina. Exemplos deste tipo de ataque incluem: ataques de dicionários, *Ftp\_write*, *Guest*, *Imap*, *Named*, *Phf*, *Sendmail*, *Xlock*, *Xsnoop* e *Named*.

Um ataque U2R (*User to Root*) ocorre quando um atacante inicia a exploração do *host* com uma conta de usuário normal do sistema e consegue explorar vulnerabilidades deste para ganhar acesso de root ao sistema. Exemplos destes ataques incluem: *buffer overflows*, *Eject*, *Ffbconfig*, *Fdformat*, *LoadModule*, *Perfl*, *Ps* e *Xterm*.

## **2.6. Técnicas Aplicadas para Análise de Dados**

Para a análise do grande volume de dados do tráfego de rede em busca de conhecimento relevante, caracterização e classificação do comportamento tráfego de rede como padrão ou anômalo, foram adotadas na TRAFICIN, várias técnicas, das quais algumas são sucintamente descritas a seguir.

### **2.6.1. Curtose**

A curtose é uma medida do quanto uma curva de frequência é achatada em relação a uma curva normal de referência (MILONE, 2004). É uma medida de

dispersão que caracteriza a diminuição ou grau de achatamento da curva característica de distribuição normal de uma população (também se diz que ela é uma medida da concentração dos dados em torno do seu centro).

Conforme apresenta a Figura 2.1, à curva normalmente achatada dá-se o nome de mesocúrtica; à mais achatada que ela, platocúrtica; à menos achatada (ou mais afilada), leptocúrtica.

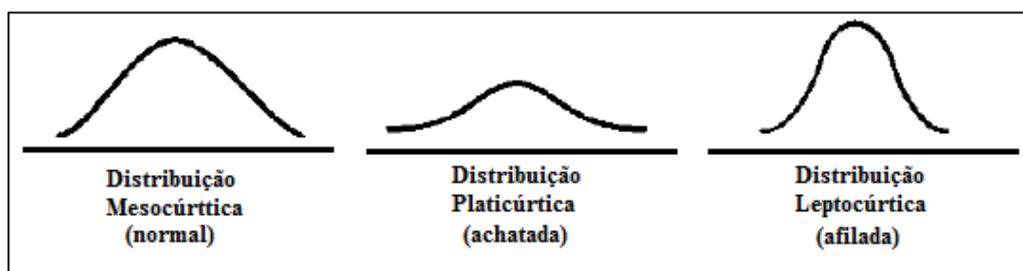


Figura 2.1 – Gráficos das curvas em relação à curtose  
Fonte: Adaptado de Milone (2004).

O grau de curtose das curvas pode ser dado tanto em relação aos momentos centrados quanto em relação às separatrizes. Os coeficientes correspondentes, para cada tipo de medida são apresentados na Tabela 2.3:

Tabela 2.3 – Coeficientes de curtose (<k>)

Momentos		Separatrizes		
		Quartidecil	Decicentil	Quarticentil
$a_2 = \frac{\bar{\mu}_4}{\bar{\mu}_2^2}$	$\gamma = a_2 - 3$	$\gamma_{QD} = \frac{A_Q}{2A_D}$	$\gamma_{DP} = \frac{A_D}{2A_P}$	$\gamma_{QP} = \frac{A_Q}{A_P}$

O  $\gamma$  é uma variável auxiliar que torna a avaliação da curtose semelhante à da assimetria, onde coeficiente negativo indica curva achatada; coeficiente nulo aponta curva normal e coeficiente positivo acusa curva alongada. O *coeficiente momento* relaciona os dois primeiros momentos de ordem par. Nele, o momento de segunda ordem está elevado ao quadrado para ter, como índice, um número puro.

As curvas são classificadas em termos de achatamento e em função dos coeficientes de curtose, como apresenta a Tabela 2.4.

Tabela 2.4 – Classificação em função do achatamento das curvas

Coeficiente	Tipo de Curva		
	Platicúrtica	Mesocúrtica	Leptocúrtica
Quartidecil	$\gamma_{OD} > 0,263$	$\gamma_{OD} = 0,263$	$\gamma_{OD} < 0,263$
Decicentil	$\gamma_{DP} > 0,275$	$\gamma_{DP} = 0,275$	$\gamma_{DP} < 0,275$
Quartacentil	$\gamma_{QP} > 0,290$	$\gamma_{QP} = 0,290$	$\gamma_{QP} < 0,290$
Momento	$\alpha_2 < 3$	$\alpha_2 = 3$	$\alpha_2 > 3$
	$\gamma < 0$	$\gamma = 0$	$\gamma > 0$

### 2.6.2. PDF

A Probability Density Function (PDF - Função de Densidade de Probabilidade) é uma técnica aplicada a séries temporais para diferentes escalas de auto-correlação, com o objetivo de definir se as flutuações são ou não Gaussianas (LUNA; BALLINI, 2005). PDF em estatística advém da função de distribuição normal. A distribuição normal ou Gaussiana foi desenvolvida pelo matemático Frances Abraham de Moivre, sendo caracterizada pela média e pelo desvio padrão. A PDF da distribuição normal é definida pela equação 2.1 abaixo:

$$f(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad (2.1)$$

onde:  $x$  = variável aleatória;  $\mu$  = média;  $\sigma$  = desvio padrão;  $\sigma^2$  = variância.

A curva normal abrange uma área que varia de infinito positivo a infinito negativo. As áreas sob a curva são divididas pelo desvio padrão em torno da média.

Se  $\mu=0$  e  $\sigma=1$ , a distribuição de frequência da população de eventos é chamada de distribuição normal padrão, a qual é definida pela simetria e curtose, e a PDF desta é reduzida à equação 2.2 abaixo:

$$f(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) \quad (2.2)$$

Ao trabalhar com a curva normal padronizada não é necessário calcular a média e o desvio padrão, sendo o z-score o parâmetro que define as áreas sob



a curva. Cada  $z$  em torno da média corresponde uma proporção bem definida de casos da população que se enquadram dentro deste.

Entretanto, duas informações sempre estão presentes e são as mais importantes, a saber, o  $z$  e a proporção de casos que se enquadram na faixa entre a média (0) e este  $z$ . Se  $z$  é conhecido, pode-se descobrir a proporção de casos que corresponde a este, ou se a proporção de casos é sabida, pode-se descobrir o  $z$  correspondente.

### 2.6.3. DFA

A Detrended Fluctuation Analysis (DFA – Análise de Flutuação Destendenciada) é uma ferramenta que tem sido utilizada para detecção de correlações de longo alcance em séries temporais. Sua abordagem permite eliminar a tendência de uma série temporal em diferentes escalas, analisando as flutuações intrínsecas do dado. As flutuações são entendidas como a medida de variabilidade do sinal associada à variância de cada segmento da série em diferentes escalas (VERONESE et al., 2010).

O algoritmo DFA considerado no TRAFICIN, foi introduzido por Peng (1994), sendo composto por seis operações computacionais a partir de uma série temporal de amplitudes discretas  $\{A_i\}$ :

- **Integração Discreta:** Calcula a representação cumulativa de  $\{A_i\}$  como demonstrado na equação 2.3:

$$C_{(k)} = \sum_{i=1}^k (A_i - \langle A \rangle), \quad (k = 1, 2, \dots, N) \quad (2.3)$$

onde  $\langle A \rangle = \frac{1}{N} \sum_{i=1}^N A_i$  é a média de  $\{A_i\}$ .

- **Janela:** Usando uma janela local de tamanho  $n$ , divide  $C_{(k)}$  de forma a não sobrepor  $N_n = \text{int}(N/n)$  sub-intervalo  $c_j$  ( $j = 1, 2, \dots, N_n$ ). Note-se que cada sub-intervalo tem tamanho  $n$  e  $N$  pode não ser um inteiro múltiplo de  $n$ . Então, a série  $C_{(k)}$  é dividida mais uma vez do lado oposto para se certificar que todos os pontos sejam abordados, apresentando no final desta operação  $2N_n$  sub-intervalos.

- **Montagem:** em cada sub-intervalo, calcula-se os mínimos quadrados ajustados da seguinte forma:

$$P_j^m(k) = b_{j0} + b_{j1}k + \dots + b_{jm-1}k^{m-1} + b_{jm}k^m, \quad m = 1, 2, \dots \quad (2.4)$$

Onde  $m$  é interpretado como a ordem da tendência retificada, denotada aqui como  $DFA^m$ .

- **Variância:** Calcula o desvio cumulativo da série em cada sub-intervalo, onde a tendência tem sido subtraída:  $C_j(k) = C(k) - p_j^m(k)$ . Em seguida, calcula a variação de  $2N_n$  sub-intervalos:

$$F^2(j, n) = \langle C_j^2(i) \rangle = \frac{1}{n} \sum_{i=1}^n [C((j-1)n + i) - p_j^m(i)]^2 \quad (2.5)$$

para  $j = 1, 2, \dots, N_n$  e:

$$F^2(j, n) \langle C_j^2(i) \rangle = \frac{1}{n} \sum_{i=1}^n [C(N - (j - N_n)n + i) - p_j^m(i)]^2 \quad (2.6)$$

para  $j = N_n + 1, N_n + 2, \dots, 2N_n$ .

- **Flutuação:** Calcula a média de todas as variâncias e a raiz quadrada para obter a função de flutuação do DFA, representada por  $F(n)$ , onde:

$$F(n) = \left[ \frac{1}{2N_n} \sum_{j=1}^{2N_n} F^2(j, n) \right]^{1/2} \quad (2.7)$$

Expoente escalar: novamente executa, recursivamente, o cálculo da janela para calcular o  $F(n)$  correspondente com diferentes  $n$  ( $[N/4] > n \geq 2m + 2$ ) de comprimentos de caixa. Em geral, na presença de flutuações na forma de lei de potência:  $F(n) = Kn^\alpha$ ,  $F(n)$  aumenta linearmente com o aumento do  $n$ . Então, usando a regressão linear pela regressão do quadrado-mínimo do duplo  $\log$  plota-se  $F(n) = \log \alpha \log n$  e pode-se obter a inclinação  $\alpha$ , que é o expoente escalar do método DFA.

Através do valor dos expoentes das flutuações alfa calculados pelo DFA, pode-se classificar uma série temporal, de acordo com a Tabela 2.5:

Tabela 2.5 – Classificação da série temporal de acordo com valores de  $\alpha$

Classificação	$\alpha$ (DFA)
Anti-persistente	$\alpha < 1/2$
Persistente	$\alpha > 1/2$
Não-persistente	$\alpha = 1/2$

O expoente de flutuação DFA ( $\alpha$ ) foi utilizado para verificar o grau de correlação das amostras, no caso, dos nove atributos das sessões do tráfego analisadas. Para informações mais detalhada do algoritmo DFA consultar Veronese et al. (2010).

#### 2.6.4. Limiarização

A técnica de limiarização utilizada neste trabalho, objetiva definir os valores de limiares que permitem rotular o tráfego de rede como padrão ou anômalo.

Na literatura, várias heurísticas são utilizadas na limiarização do tráfego de rede. No trabalho de Lakhina et al. (2004), um limiar estatístico simples foi aplicado para rotular o fluxo de tráfego anômalo. Se o intervalo de endereço ou porta representou mais do que a fração  $p$  do tráfego todo na variável *timebin* (definido sobre um os três atributos utilizados: *bytes*, *pacotes*, *IP-level flows*) então considera-se que o fluxo possui características dominantes (anomalias). Os autores descobriram que um valor de  $p=0,2$  funcionou bem nesta heurística de limiarização.

Uma outra heurística de limiarização criada para rotular o tráfego como anômalo ou padrão é apresentada por Soysal e Schmidt (2010). Os autores apresentam uma abordagem sistemática para investigar e avaliar o desempenho de três algoritmos de máquina de aprendizagem (Redes Bayesianas, Árvore de Decisão e rede MLP) para classificar o fluxo de rede como anômalo ou padrão. Os dados para treinamento foram rotulados a partir de: um valor de limiar abstraído de um estudo rigoroso de um grande volume de dados e o tráfego de tipos diferentes de serviços de rede, levando-se em consideração diferentes frações de dados de cada tipo de tráfego, bem como a quantidade de fluxo gerada por cada um.

### **2.6.5. Clusterização**

A técnica de clusterização visa segmentar um conjunto de dados num número de subgrupos homogêneos ou *clustering*. Tem por objetivo formar grupos o mais homogêneos em si e mais heterogêneos entre si. A diferença fundamental entre a formação de *clustering* e a classificação é que na clusterização não existem classes predefinidas para agrupar os dados em estudo. Os dados são agrupados em função de suas similaridades, ou seja, quando se deseja utilizar a técnica de clusterização, seleciona-se um conjunto de atributos e em função da similaridade desses atributos são formados os *clustering* (SINGH; RAJAMENAKSHI, 2009).

De acordo com a literatura (HUBBALLI et al., 2009; MAHMOOD et al., 2008; SINGH, RAJAMENAKSHI, 2009; YU et al., 2005; XU, WUNSCH, 2005), existem diferentes algoritmos de clusterização implementados para caracterização do comportamento do tráfego de redes de computadores.

Uma abordagem utilizando uma combinação de métodos foi desenvolvida por Shon e Moon (2007) com o objetivo de proporcionar aprendizagem não supervisionada e capacidade reduzida de alarmes falsos na detecção de anomalias em redes de computadores. Os autores utilizaram as seguintes técnicas: primeiro o clusterizador SOFM (*Self-Organized Feature Map*) para criar um perfil de pacotes normais, sem conhecimento pré-existente; um sistema de filtragem PTF (*Fingerprinting*), a fim de rejeitar o tráfego incompleto; uma técnica de seleção de atributos utilizando um algoritmo genético (GA) para extrair informações dos pacotes HTTP otimizados e SVM para detectar anomalias no tráfego.

No trabalho de Singh e Rajamenakshi (2009), uma abordagem para análise e visualização de dados de fluxo de rede através de clusterização é apresentada. A técnica de clusterização é baseada em K-Means para analisar três atributos (IP, portas, protocolos) e prever o tipo de fluxo, ou seja, se é anômalo ou normal.

### **2.6.6. Rede Neural MLP**

O modelo de rede neural MLP (Multi-Layer Perceptron) tem treinamento baseado no processo de aprendizagem supervisionada. Trata-se de uma rede alimentada adiante (*feedforward*), pois o sinal entre os neurônios se propaga apenas para frente, camada a camada. Possui uma ou mais camadas de neurônios ocultos, responsáveis pela extração gradual das características mais significativas presentes nos padrões de entrada durante o processo de treinamento (SWIFT; DAGLI, 2008). Cada neurônio das camadas da rede inclui uma função de ativação não-linear. Uma das funções mais utilizadas é a sigmóide, caracterizada pela função logística:

$$y_j = \frac{1}{1 + e^{-v_j}} \quad (2.8)$$

Onde  $v_j$  é o campo local induzido do neurônio  $j$ , calculado através da soma de todas as entradas sinápticas ponderadas pelos respectivos pesos, e  $y_j$  é a saída do mesmo neurônio.

O algoritmo de treinamento da MLP baseado na regra de aprendizagem por correção do erro é chamado de back-propagation (retro-propagação de erro) e consiste em dois passos: propagação (para frente) e retropropagação (para trás).

No primeiro passo, um vetor padrão é aplicado aos nós de entrada da rede e seu efeito se propaga através do modelo, camada por camada, até gerar um sinal de saída como resposta da rede. Durante a segunda parte do algoritmo, os pesos sinápticos são ajustados, de acordo com uma regra de correção de erro. Os pesos são então ajustados de maneira a fazer com que a resposta real da rede se aproxime cada vez mais da resposta desejada.

Critérios de parada são utilizados para encerrar a fase de ajuste dos pesos. Informações adicionais sobre a rede MLP podem ser encontradas em (FAUSETT, 1994).

#### **2.6.7. SVM**

As máquinas de vetor de suporte (SVM, do inglês *Support Vector Machine*) são uma classe de máquinas de aprendizagem que se utiliza de aprendizagem

supervisionada e pode ser utilizada tanto para classificação de padrões quanto para problemas de regressão linear. A idéia principal deste tipo de rede é construir um hiperplano como superfície de decisão de tal forma que a margem de separação entre os exemplos positivos e negativos seja máxima, para o caso de um problema de classificação (VAPNIK, 1998).

Estabelecida sobre a teoria do método de minimização estrutural de risco, a SVM se mostra especialmente resistente também a problema de *over-fitting*, atingindo freqüentemente um alto desempenho de generalização na solução de problemas de previsão em séries temporais (CAO, 2002) e, também, na classificação de padrões.

O projeto da máquina depende principalmente da extração de um subconjunto dos dados de treinamento que representem características estáveis dos dados, chamados de vetores de suporte, através de um algoritmo. Dependendo de como este núcleo interno é gerado, pode-se construir diferentes máquinas de aprendizagem (polinomial, função de base radial), que se caracterizam por superfícies de decisão não-lineares próprias.

SVM pode ser entendida como uma máquina de aprendizado universal cuja superfície de decisão é parametrizada por um conjunto de vetores de treinamento e seus correspondentes pesos. A separação ótima entre os vetores de treinamento dentre as diferentes classes corresponde à separação ótima entre as classes no conjunto de treinamento (HAYKIN, 2001).

Dado um conjunto de treinamento rotulado  $D_1 = \{(x_i, y_i) \in X \times Y; i = 1, \dots, l\}$  com  $Y = \{-1, +1\}$ , o objetivo consiste em encontrar o hiperplano de separação que classifica corretamente as  $l$  amostras de treinamento. O hiperplano ótimo corresponde ao hiperplano de maior margem de separação entre duas classes. A margem de separação corresponde ao intervalo definido ao longo do hiperplano, cuja largura é determinada em função dos vetores mais próximos ao hiperplano, denominados vetores de suporte.

A função de base radial (RBF), apresentada na equação 2.9, é empregada freqüentemente na literatura como função kernel nas SVMs:

$$K_{RBF}(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{\sigma^2}} \quad (2.9)$$

Onde  $\sigma$  é um parâmetro responsável pela flexibilização do hiperplano de separação. A formulação do método SVM permite apenas a separação entre duas classes, o que não atende a maioria dos problemas reais. Para contornar esta limitação são empregadas Estratégias Multiclasse, as quais basicamente consistem na combinação entre classificadores binários. Alguns exemplos de Estratégia Multiclasse são abordados em Webb (2002).

#### 2.6.8. Rede RBF

Outro tipo de rede neural que tem se tornado uma alternativa de modelo não linear para problemas de regressão e classificação são as redes neurais de função de base radial. A determinação de uma função de base radial usada como função de transferência, aplicada aos neurônios ocultos de uma RBF, é o fator primordial desta arquitetura de rede neural (HAYKIN, 2001).

De acordo com Haykin (2001), a arquitetura das RBF consiste em uma rede com três camadas, sendo uma de entrada, uma oculta e a outra de saída. A primeira camada apenas propaga as entradas, isto é, não há aplicação de parâmetros de ponderação, ou ainda, as saídas são iguais às entradas.

Na camada oculta ocorre a mais importante etapa do processamento. Nesta, as unidades devem satisfazer uma propriedade de serem radialmente simétricas às quais devem possuir:

- Um vetor centro ( $c_k$ ) gerado a partir dos dados de entrada considerando os centros de aglomerados de dados;
- Uma medida de distância (raio) ( $D_k$ ) que representa o quão distante está um vetor de entrada ( $x_i$ ) do vetor centro, a exemplo da equação 2.10, onde N é o número de padrões:

$$D_k = \sqrt{\sum_{i=1}^N (x_i - c_{ik})^2} \quad (2.10)$$

Uma função de transferência de bases radiais que transforma a distância  $D_k$  na saída de cada unidade de processamento da camada, sendo a mais comum a

Gaussiana, apresentada a seguir, onde  $\sigma_k$  é uma constante denominada fator de escala ou desvio-padrão do espaço delimitado pela unidade  $k$ :

$$v_k = \exp\left(\frac{-D_k^2}{\sigma_k^2}\right) \quad (2.11)$$

Resumindo, a camada oculta processa a informação em duas etapas: o cálculo da distância  $D_k$  e a aplicação da função de transferência. As saídas das unidades da camada oculta são totalmente conectadas às unidades da camada de saída final através de suas respectivas ponderações. A última camada possui unidades lineares, isto é, cada saída é resultado da soma ponderada das entradas, sendo os pesos ajustados através de um método como o backpropagation citado anteriormente. Informações adicionais sobre a rede RBF podem ser encontradas em (HAYKIN, 2001).

#### **2.6.9. Árvore de Decisão**

Uma árvore de decisão é um modelo de aprendizagem de máquina preditivo que decide o valor-alvo (variável dependente) de uma nova amostra com base em valores de atributos diferentes dos dados disponíveis. Os classificadores de árvores de decisão constroem uma árvore, onde os nós internos indicam os atributos diferentes, os ramos entre os nós dizem os possíveis valores que estes atributos podem ter observado nas amostras, enquanto os nós terminais nos dizem o valor final (classificação) de variável dependente (BREIMAN et al., 1984).

O atributo que deve ser previsto é conhecido como variável dependente, pois seu valor depende, ou é decidida pelos valores de todos os outros atributos. Os outros atributos, que ajudam na previsão no valor da variável dependente, são conhecidos como variáveis independentes no conjunto de dados. Segundo Breiman et al. (1984), a estrutura da árvore é organizada de tal forma que:

- Cada nó interno (não-folha) é rotulado com o nome de um dos atributos previsores; os ramos (ou arestas) saindo de um nó interno são rotulados com valores do atributo naquele nó; cada folha é rotulada com uma classe, a qual é a classe prevista para exemplos que pertençam àquele nó folha.



O processo de classificação de um vetor ocorre fazendo-o “caminhar” pela árvore, a partir do nó raiz, procurando percorrer os arcos que unem os nós, de acordo com as condições que estes mesmos arcos representam. Ao atingir um nó folha, a classe que rotula aquele nó folha é atribuída àquele vetor. Mais detalhes da Árvore de Decisão podem ser encontradas em Breiman et al. (1984).

#### 2.6.10. Máxima Verossimilhança

Máxima Verossimilhança (MaxVer) é um método de classificação, que considera a ponderação das distâncias entre médias das classes, utilizando parâmetros estatísticos. O método é baseado em aprendizagem supervisionada, ou seja, semelhantemente ao funcionamento das redes neurais, deve ser fornecido ao sistema um conjunto de dados ou amostras que representam as classes de interesse para que o classificador tenha condições de definir um diagrama de dispersão em torno da média, bem como, a distribuição das probabilidades das amostras pertencerem ou não à determinada classe (VIEIRA, 1996).

Dado um conjunto de dados com N vetores, cada vetor descrito pelo seu vetor de atributos  $x$  será classificado como sendo da classe  $\omega_j$  se  $g_j(x) > g_k(x)$  para *todo*  $j \neq k, j, k = 1, \dots, N$ .

Onde  $g_j$  é denominada função discriminante para a classe  $\omega_j$ . Para o classificador MaxVer, onde os dados seguem uma distribuição gaussiana multivariada, a função discriminante é dada por Duda e Hart (1973):

$$g_j(x) = -\frac{1}{2} (x - m_j)^t \Sigma_j^{-1} (x - m_j) - \frac{1}{2} \ln |\Sigma_j| + \ln P(\omega_j) \quad (2.12)$$

Onde  $m_j$  e  $\Sigma_j$  são o vetor média e a matriz de covariância e  $P(\omega_j)$  é a probabilidade a priori da classe  $j$ . As médias e matrizes de covariância das classes de interesse são estimadas a partir de amostras conhecidas. As probabilidades a priori são normalmente consideradas iguais e desprezadas no cálculo da função de comparação. Outras informações sobre o MaxVer podem ser encontradas em Duda e Hart (1973).



### **3 APRESENTAÇÃO DO PROBLEMA**

A natureza do tráfego de redes TCP/IP é multivariada e dinâmica, o que torna um desafio a tarefa de construção de uma base de conhecimento adequada que represente o comportamento padrão do tráfego ao longo do tempo. Além disto, o tráfego de rede produz grandes volumes de dados. A análise destes conjuntos requer um tempo de processamento considerável, inviabilizando o desenvolvimento de sistemas detectores de anomalias em tempo real.

A utilização de modelos imprecisos de comportamento padrão do tráfego de rede e a análise de grandes volumes de dados de tráfego corrente resultam em baixo desempenho do módulo de detecção de anomalias do SDI (Sistemas de Detecção de Intrusos) e altas taxas de alarmes falsos.

Para detectar anomalias em redes TCP/IP, deve-se ter conhecimento de quais tipos de serviços são fornecidos, bem como da quantidade de usuários e da dinâmica do acesso aos serviços (JUN, 2009).

#### **3.1. Detecção de Anomalias no Tráfego de Rede**

A premissa central da detecção de anomalias é que a atividade intrusiva é um subconjunto de atividades anômalas (AZZINI et al., 2008).

Na construção de sistemas de detecção de anomalias são usados modelos estatísticos ou técnicas de Inteligência Computacional, tais como: aprendizagem de máquina, redes Bayesianas, análise de componente principal, modelos de Markov, mineração de dados, lógica fuzzy, redes neurais, algoritmos genéticos, *clustering*, entre outros (PATCHA; PARK, 2007). A monitoração rigorosa de padrões de ataques bem conhecidos são adotados quando se usa detecção por assinatura. Ambos os métodos trazem implícitas algumas suposições sobre a natureza das intrusões que podem ser detectadas por eles e, possuem determinadas aplicações e limitações. Além disso, os métodos de detecção podem variar de acordo com a técnica empregada, e pelo tipo de dados ou análises que utilizam dentro dos sistemas computacionais.

Os Sistemas de Detecção de Intrusos (SDI) são ferramentas de *software* de defesa utilizadas para relatar eventos suspeitos ou impedir que ações maliciosas tenham êxito e se propaguem pela rede. Os SDI são úteis não somente para se detectar falhas de segurança que foram, ou podem ser exploradas com sucesso, mas também para monitorar tentativas de ataque e fornecerem mecanismos importantes para adoção de contra-medidas imediatas (DEPREN et al., 2005).

Pode-se classificar os SDI com base em três critérios: método de detecção, arquitetura e comportamento após a detecção. Quanto ao comportamento após a detecção, um SDI é denominado ativo quando reage aos ataques, executando medidas reativas após a detecção de intrusos. Por exemplo, um SDI projetado para realizar um bloqueio de conexões provenientes de uma origem quando da identificação de determinado tipo de pacote. Por outro lado, SDI passivo é aquele que examina as informações da rede ou sistema e alertam sobre os ataques e ameaças, não reagindo aos ataques (CASWELL et al., 2003).

Quanto ao método utilizado para a detecção de ataques, pode-se classificar o SDI em duas categorias: detecção por assinatura e detecção por anomalia (BURGESS, 2006).

Técnicas de detecção por assinatura tentam modelar os ataques de um sistema como padrões específicos, então sistematicamente realizam a varredura do sistema em busca de ocorrências destes padrões pré-definidos. Em outras palavras, as decisões são tomadas com base nos conhecimentos adquiridos a partir do modelo do processo intrusivo e do traço observado que o ataque deixou no sistema. Comportamentos legais ou ilegais podem ser definidos e comparados com o comportamento observado. Tal sistema tenta coletar evidências de atividade intrusiva, independentemente do comportamento normal do sistema (ZARPELÃO et al., 2009).

Um dos principais benefícios no uso de detecção de assinaturas é que pode ser detectado com baixa taxa de falsos positivos. A existência de seqüências de ataque específico garante facilidades para o administrador do sistema

determinar exatamente que ataque o sistema está enfrentando atualmente. Se os dados da auditoria nos arquivos de *log* não contêm a assinatura de ataque, nenhum alarme é disparado.

Atualmente, a maioria das ferramentas de detecção de intrusão comerciais são baseadas em assinaturas, devido ao seu reduzido custo computacional e bom desempenho.

Na detecção por anomalia, admite-se que abusos ou intrusões são fortemente correlacionadas a comportamento anômalo apresentado pelo usuário, sistema ou rede. Seu principal objetivo é modelar o comportamento padrão adaptativo do alvo monitorado para que os eventos ilegítimos, fora do padrão, possam ser detectados o mais rápido possível. Porém, o comportamento padrão só pode ser determinado através da aprendizagem sobre os acontecimentos do passado. Com base no modelo do comportamento padrão aprendido, faz-se a classificação dos dados correntes. A execução desta tarefa em tempo real requer rápido processamento dos dados (DEPREN et al., 2005).

Os sistemas de detecção de anomalias são capazes de detectar ataques desconhecidos como desvios do comportamento padrão previamente mapeado.

Os alarmes gerados por estes SDI são classificados como sendo falso positivo ou falso negativo. O alarme falso positivo ocorre quando o SDI relata uma atividade de rede legítima como um evento intrusivo. O alarme falso negativo ocorre quando atividades de ataque ou maliciosas na rede ou sistema passam despercebidas pelo SDI (FONTUGNE et al., 2009).

O SDI híbrido ou composto combina as duas abordagens: assinaturas e anomalias. Em essência, um SDI híbrido é um sistema de detecção de assinatura inspirado na intrusão, que toma decisão utilizando um modelo híbrido que se baseia tanto no comportamento normal do sistema quanto no comportamento intrusivo dos invasores.

De acordo com Depren et al. (2005), a classificação dos SDI quanto à arquitetura considera o alvo monitorado e a localização do SDI na rede.

Segundo o alvo, o SDI é classificado como baseado em rede ou baseado em *host*. Os SDI baseados em *host* (HIDS – *Host Intrusion Detection Systems*) analisam a atividade do sistema através de dados coletados na própria máquina, enquanto sistemas baseados em rede (NIDS - *Network Intrusion Detection System*) capturam e analisam pacotes de rede em busca de anomalias no tráfego.

### 3.2. Problemas de Detecção de Anomalias em Redes

Diferentes problemas motivam as pesquisas realizadas no campo de detecção de anomalias em redes em nível mundial.

De acordo com Schmert et al. (2010), as principais limitações dos SDI baseados em assinaturas são:

- **Atualização contínua da base:** com o crescente número de ataques nas redes de computadores, é necessária atualização contínua das bases, com as assinaturas de ataques catalogadas;
- **Incapacidade de detectar novos ataques:** somente identifica ataques cujas assinaturas são conhecidas. Para tanto, é necessário que já tenham ocorrido;
- **Alta taxa de alarmes falsos negativos:** não alertam sobre ataques ocorridos cujas assinaturas não estejam modeladas em sua base.
- **Falta de método sistemático para modelagem de novas assinaturas:** a modelagem das novas assinaturas é um processo demorado, propenso a erros e executado de forma empírica, baseado em conhecimentos especializados e experiência.

Já alguns dos problemas enfrentados no desenvolvimento dos SDI baseados em anomalias no tráfego de rede são (BURGESS, 2006):

- **Subjetividade:** o comportamento incomum para um *host*, sistema ou rede pode não ser para outro. Portanto, é necessária uma interpretação humana e individual do comportamento das redes;

- **Precisão do método de detecção:** a quantidade de alarmes falsos positivos e falsos negativos gerados na análise do tráfego pode reduzir a produtividade, como decorrência da deficiência em distinguir comportamento legítimo (padrão) e comportamento anômalo;
- **Seleção de atributos:** O tráfego de rede possui muitas características passíveis de análise. A seleção de atributos que não influenciam na caracterização do tráfego de rede pode aumentar o tempo de computação e impactar na precisão de um SDI. Assim a seleção de atributos pode ser usada para encontrar características mais indicativas de anomalias, excluindo as informações menos importantes e/ou redundantes da base de conhecimento, de modo a melhorar a eficiência das técnicas de SDI em relação ao tempo de detecção de anomalias. A dificuldade em refinar a quantidade de atributos do tráfego de rede a analisar é devido ao fato de não existirem procedimentos formais para aplicar na seleção;
- **Sintetização dos atributos:** muitas vezes, as informações sobre o comportamento do tráfego da rede não estão diretamente disponíveis nos atributos primitivos, mas devem ser sintetizadas em atributos derivados para que a compreensão do comportamento das informações em conjunto seja obtida;
- **Treinamento do modelo:** os SDI devem ser treinados para aprender sobre o comportamento padrão do tráfego de rede. Porém, o treinamento exige processamento de grande quantidade de dados para modelar o comportamento padrão histórico, requerendo maior espaço em disco e recursos de CPU. Esse processamento é contínuo, pois a base de conhecimento precisa ser atualizada.
- **Caracterização do tráfego:** caracterizar o tráfego de rede é extrair a base de conhecimento do comportamento padrão do tráfego e armazená-la para posterior treinamento dos SDI. O problema é que o comportamento padrão do tráfego de rede é dinâmico, ou seja, não é o mesmo em todos os momentos do dia. Somente com o domínio do

modelo dinâmico do tráfego padrão da rede será possível realizar um diagnóstico de qualidade para a detecção de anomalias. Além disto, deve ser considerado o tráfego de cada serviço fornecido, que apresentam suas particularidades. Outra dificuldade encontrada na caracterização do tráfego é a falta de consenso sobre qual modelo é capaz de caracterizar o tráfego de maneira eficiente;

- **Ruídos:** o tráfego apresenta um nível considerável de ruídos, ou seja, informações desnecessárias que podem alterar as características do modelo de caracterização do tráfego e inviabilizá-lo para uso futuro. Por exemplo, os roteadores enviam pacotes sobre o roteamento de dados que não são necessários na análise de tráfego de serviço HTTP. Devem ser capturados para o SDI apenas pacotes relacionados a informações de interesse dos serviços monitorados, informações consideradas ruídos devem ser descartadas;
- **Limiarização:** dificuldade no processo empírico de configuração e ajuste dos limiares de valores de atributos do tráfego considerado padrão ou não (limites que apontam a anormalidade), os quais treinam sobre os dados históricos e permitem construir modelos do comportamento padrão;
- **Desempenho do sistema:** em geral, o desempenho do SDI é baixo, principalmente devido à grande quantidade de dados para processar e aos cálculos complexos exigidos, requerendo uma metodologia para a redução dos dados sem perda das informações mais significativas para análise. A etapa de treinamento dos SDI requer rapidez no processamento dos dados para que a classificação do tráfego seja eficiente. Isto é importante e desejado na detecção de anomalias de rede em tempo real;
- **Comportamento do tráfego:** dificuldade na modelagem do comportamento do tráfego e no tratamento de mudanças do comportamento já mapeado. Caso as alterações sejam sutis, mas contínuas, elas podem comprometer a eficiência dos SDI. Os atacantes



podem desviar-se lentamente do comportamento padrão do sistema, fazendo o detector confundir o tráfego de ataque com o padrão da rede. Este processo malicioso, denomina-se evasão;

- **Identificação de falhas na detecção de anomalias:** problemas na identificação de anomalias de rede podem ter origem na coleta de dados, na redução de dados, na caracterização do comportamento ou na classificação dos dados. Encontrar os pontos de falha no sistema requer tempo para análise de cada etapa e retrabalho na solução;
- **Atualização da base de conhecimento:** a base de conhecimento do comportamento padrão do tráfego deve ser constantemente atualizada para reduzir a quantidade de alarmes falsos gerados, pois a frequência de acesso aos serviços, os tipos de serviços oferecidos, a quantidade de *hosts* e o comportamento dos usuários no uso dos recursos computacionais podem mudar o padrão de comportamento do ambiente de rede, sem, portanto, caracterizar uma anomalia.

### 3.3. Caracterização do Problema

Conforme o contexto apresentado, muitas dificuldades ainda estão presentes no campo de detecção de anomalias em redes de computadores, devido à natureza complexa e específica do tráfego. Diversos fatores que afetam os processos de treinamento e classificação dos SDI são continuamente investigados para obter modelos mais precisos e adaptáveis do comportamento padrão do tráfego, a saber: forma de construção e atualização da base de conhecimento, o tratamento de grandes volumes de dados e a definição de valores de limiares que representam as fronteiras de comportamento padrão e anômalo.

O escopo deste trabalho concentra-se na solução de dois problemas relacionados à modelagem do comportamento padrão do tráfego:

- a) Problema 1:** A modelagem incorreta do comportamento padrão do tráfego afeta a precisão de treinamento do módulo de detecção, gerando altas taxas de alarmes falsos.

O comportamento do tráfego varia de acordo com os tipos de serviços fornecidos, quantidade de usuários e *hosts*, frequência de acesso aos serviços, alteração na infra-estrutura física e lógica da rede e fatores temporais, incluindo período do dia e dia da semana em que os serviços são acessados. O comportamento dinâmico do tráfego dificulta a sua padronização.

Além disto, a característica do tráfego de rede de apresentar eventos normais com frequência significativamente maior que eventos anômalos, dificulta o treinamento do classificador que pode saturar e perder a capacidade de generalização.

Estes fatores dificultam a tarefa de construção de uma base de conhecimento que represente significativamente o comportamento padrão do tráfego ao longo do tempo.

**b) Problema 2:** A escolha de uma heurística (método) inadequada para a redução dos dados sem perda de informações representativas do conjunto afeta o tempo de treinamento e classificação do módulo de detecção, reduzindo o desempenho do SDI.

Visto que o tráfego de rede é produzido em grandes volumes de dados, a análise dos conjuntos requer tempo considerável de processamento do SDI para treinar o módulo de detecção e identificar anomalias em tempo satisfatório. Por exemplo, um arquivo de 10 minutos de tráfego HTTP coletado em uma rede de 100 usuários em horário comercial apresenta em média 125.850 pacotes correspondente a 1535 sessões de rede, sendo 144 arquivos para analisar diariamente, 221.040 sessões por dia.

Outra limitação de trabalhos nesta área consiste na dificuldade de criação de um modelo compacto (ou conciso) do comportamento do tráfego, contendo conjuntos de dados menores, porém relevantes. O tamanho da base de conhecimento do modelo padrão do tráfego, afeta o tempo de treinamento do classificador, que deve ser realizada com frequência satisfatória para contemplar as modificações constantes sofridas pelo tráfego.

### 3.4. Abordagens Existentes para Solução do Problema

A fim de se detectar anomalias nos complexos ambientes de rede de computadores, os pesquisadores e profissionais da área de segurança de redes têm proposto diferentes soluções para tratamento dos problemas acima apresentados.

Na literatura encontram-se diversos estudos recentes sobre metodologias e ferramentas de detecção de anomalias. Algumas destas pesquisas desenvolvidas nos últimos dez anos são abordadas a seguir.

#### **a) Modelo ACME! (*Advanced Counter-Measures Environment*) para Detecção de Novas Técnicas de Intrusão**

O modelo inicial ACME! consistia em um SDI baseado em métodos de detecção por assinatura, que utiliza um agente capaz de descobrir comportamento intrusivo em redes de computadores. Este agente utiliza técnicas de captura de pacotes e métodos de detecção, aliados a uma rede neural MLP (Multi-Layer Perceptron) que, além de descobrirem comportamento intrusivo, podem auditar e fornecer elementos que auxiliem na tomada de contramedidas (SOUZA, 2002).

Em sua primeira versão o ACME! era baseado apenas no método de detecção por assinatura, consistindo nos seguintes módulos:

- Módulo de Captura de Pacotes (CAP): a função básica deste módulo é identificar e coletar os pacotes, enviando esse conjunto de informações para os módulos de pré-seleção e sistema especialista, e também para o módulo de conexão.
- Módulo de Pré-Seleção e Sistema Especialista (PSSE): este módulo decide a partir de quando uma conexão é considerada suspeita. Quando isso ocorre, vários procedimentos são executados simultaneamente.

- Módulo de Conexão (CON): a finalidade deste módulo é a criação e manutenção de vetores de conexão. O vetor de conexão é um arquivo contendo a reconstrução do fluxo de dados, ou seja, contém todos os dados que trafegam naquela conexão a partir do momento em que ela foi considerada suspeita, incluindo desde informações de controle (portas, endereços, tipos de frame, etc.) até os dados efetivamente transportados. De posse desses dados, o analisador semântico faz a busca por trechos suspeitos que, combinados caracterizem um ataque.
- Analisador Semântico e Pré-Processador (ASPP): De posse do vetor de conexão, é criado o vetor de estímulo, que é a codificação binária do vetor de conexão. Com os dados condensados pelo módulo de conexão, o sistema pode interpretar e selecionar aquilo que for relevante.
- Módulo de Rede Neural Artificial (RNA): a interface com a RNA é o elemento principal deste módulo, o qual recebe todo o conjunto de bits que compõe o vetor de estímulo e, baseado no treinamento ao qual a rede foi submetida, retorna uma porcentagem que indica o grau de suspeita da sessão.

Na versão atual do ACME! o fluxo de pacotes passa por mais uma etapa de análise, localizada dentro do próprio Módulo PSSE (Pré-Seleção e Sistema Especialista), que verifica a conformidade dos pacotes de acordo com regras de aceitação. O Diagrama geral do modelo ACME! reestruturado é apresentado na Figura 3.1.

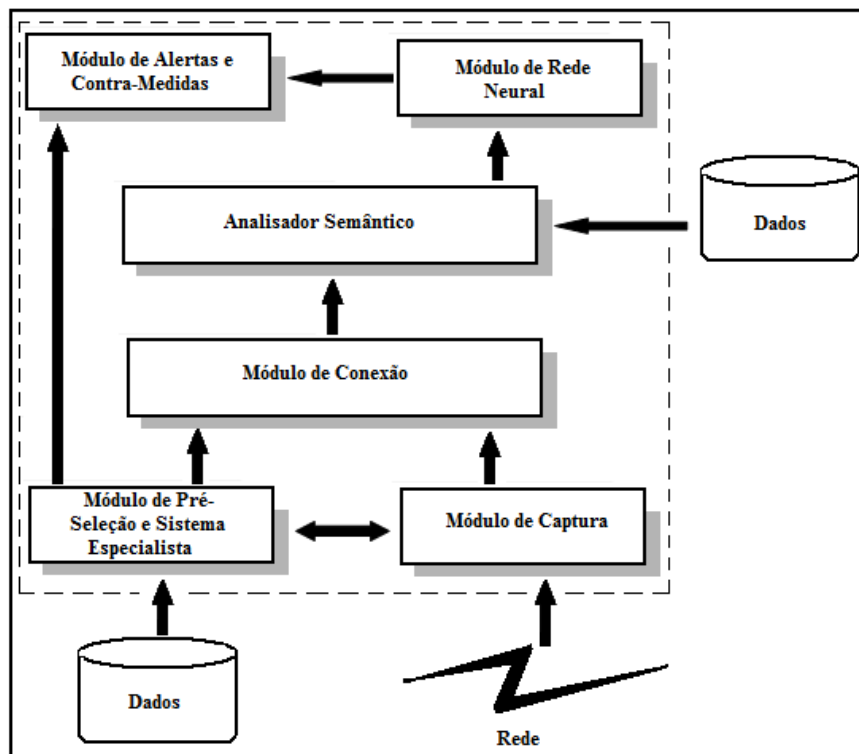


Figura 3.1 – Diagrama geral do modelo ACME! reestruturado  
Fonte: SOUZA ( 2002).

O conjunto de funções de alertas e contramedidas foi destacado do módulo de Rede Neural Artificial (RNA), já que este passou a não ser mais o único agente capaz de realizar funções reativas. Isto gerou o módulo ACM (Alertas e Contra Medidas), especificando que as ações a serem tomadas no momento da identificação de comportamento abusivo ou anômalo possam ser requisitadas por diferentes componentes do sistema, entre eles o PSSE e o RNA.

#### **b) Detectando ataques DoS (*Denial of Service*) usando SVM (*Support Vector Machines*)**

Ataques DoS - *Denial of Service* (Negação de Serviço) são caracterizados por ataques projetados para tornar um *host*, ou uma rede, incapaz de prover serviços normais. Um estudo referente à detecção de ataques DoS utilizando SVMs - *Support Vector Machines* (Máquinas de Vetor de Suporte) pode ser encontrado em (MUKKAMALA; SUNG, 2003).

Os dados de referência utilizados no trabalho foram de uma competição KDD-Cup 99 - *The Fifth International Conference on Knowledge Discovery and Data*

*Mining*, conduzida pela DARPA (*Defense Advanced Research Projects Agency*).

Os autores apresentaram uma abordagem de *mineração de dados* e identificaram 41 atributos quantitativos para cada conexão TCP/IP. Um subconjunto de 494.021 dados de treinamento e teste foi utilizado no trabalho, dos quais 20% representam padrões normais.

Na pesquisa foi realizada a subdivisão dos dados em duas classes de padrões “normal” e “DoS”, onde o ataque DoS era um conjunto de seis diferentes ataques (*back*, *neptune*, *ping-of-death*, *land*, *smurf* e *teardrop*). O objetivo foi separar padrões normais de ataques DoS.

Nos experimentos realizados foram utilizadas SVMs para classificar padrões de diferentes modos. No primeiro conjunto de experimentos, foram usadas SVMs para classificar padrões normais x padrões DoS, tendo um resultado de 99,69% de acerto nas classificações. No segundo conjunto de experimentos, foram classificados padrões DoS versus o restante dos padrões, os quais incluem outros tipos de ataques, obtendo o resultado de 99,25% de classificação correta. Posteriormente os experimentos foram ampliados para classificadores de DoS de instâncias-específicas, com resultados satisfatórios. Também foram aplicados dois diferentes métodos para categorizar os atributos relacionados a padrões de ataques DoS com o propósito de identificar os atributos-chave que podem ajudar no reconhecimento destes tipos de ataques com melhor precisão e/ou detecção mais rápida.

A detecção DoS usando SVMs consiste de 3 fases:

- Pré-processamento: um *parser* (analisador/processador léxico) automatizado é usado para processar os dados *dump* brutos TCP/IP em um formato apropriado;
- Treinamento: a SVM é treinada com diferentes tipos de ataques e dados normais. Foram utilizados no trabalho 41 atributos e 2 classes, uma representa dado normal (-1) e a outra é dado de ataque (+1);

- Teste: a performance da SVM treinada é testada para garantir que adquiriu capacidade adequada de classificação.

Na aplicação de SDI e, especificamente, para detecção de DoS, SVMs funcionam bem e atuam melhor que técnicas de aprendizagem de máquina, tais como redes neurais, em aspectos importantes de escalabilidade, tempo de treinamento, tempo de execução e precisão de detecção. Em particular o tempo de treinamento de SVMs apresenta, em geral, uma ordem de magnitude menor que o das redes neurais, enquanto a precisão da detecção é notavelmente maior.

### **c) ADR: Um Sistema de Detecção de Intrusos baseado em Mapas de Kohonen**

O ADR desenvolvido por Araújo et al. (2005), utiliza os conceitos de RNA, através da rede de Kohonen, para implementar um sistema de detecção de intrusão em um ambiente de rede. Seu objetivo principal é detectar ataques de varredura de portas em uma rede de computador e armazenar as informações coletadas em um arquivo *log*.

O sistema ADR utiliza o conceito de similaridade expresso pelo parâmetro de vigilância, sendo composto por dois mapas de Kohonen: o mapa de condições normais e o mapa de exceções. O parâmetro de vigilância é um limiar que determina a similaridade mínima entre o mapa e o vetor que lhe é apresentado, sendo rejeitado e enviado ao mapa de exceções o vetor com menor similaridade. O mapa de condições normais é formado por um conjunto de *clusters* que expressam uma similaridade adequada aos dados coletados do sistema em operação normal. Esse conjunto de dados de referência precisa ser representativo de todas as situações normais. Caso a classificação pelo mapa não atinja o grau de similaridade mínima proposto pelo parâmetro de vigilância, o dado é rejeitado e classificado em um mapa auxiliar, chamado mapa de exceções. Esse mapa é ajustado iterativamente de modo a determinar classes de situações de anormalidades obtidas pelo sistema.

O sistema ADR é composto por quatro módulos: captura, normalização, RNA e decisão:

- Módulo de captura: faz a coleta dos dados para o modelo. Também chamados de *sniffers*, são programas que gravam toda a atividade de uma rede, sobrepondo-se aos controles normais dos protocolos de comunicação.
- Módulo de normalização: é onde ocorrerá o tratamento dos valores coletados para que sejam entregues à análise da RNA. A normalização ajusta a escala de valores obtidos.
- Módulo RNA: neste módulo, o ajuste dos pesos dos mapas foi efetuado conforme descrito em Kohonen (1990).
- Módulo de decisão: baseia-se na informação de normalidade informada pelo módulo RNA. Efetua o alerta ao administrador da rede, indicando os dados coletados associados.

Os resultados encontrados no ADR foram comparados com os de uma ferramenta de propósito similar, denominada Beholder.

Os testes foram separados em duas etapas: etapa de acesso normal e de ataque. A máquina Cliente possui dois tipos de comportamento: como atacante efetua ataques tipo varredura de porta, como cliente utiliza serviços (acesso à página Web, FTP) nos servidores. Para efetuar os ataques foi utilizada a aplicação Nmap versão 3.75.

As duas ferramentas (ADR e Beholder) apresentaram comportamentos semelhantes, demonstrando um resultado satisfatório. Os resultados dos testes de ataque também foram similares. Por exemplo, segundo os autores, nos testes de ataque (varredura de porta ao servidor) o ADR reconheceu 99,4% dos 550 pacotes reconhecidos como suspeitos, enquanto o Beholder reconheceu 99,6% dos mesmos pacotes suspeitos.



#### **d) ADTRAF - *Attack Detection on the network TRAFfic***

O ADTRAF - (Detecção de Ataques no Tráfego de Redes) é um sistema de detecção de anomalias e assinaturas baseado em redes, que realiza a análise *off-line* (ou *pos-mortem*) das sessões TCP/IP contidas no tráfego de rede (SILVA, 2007).

O sistema possui uma arquitetura centralizada, em que o sensor do SDI coleta os dados de um ponto específico da rede e apresenta um comportamento passivo, visto que apenas alerta a ocorrência de eventos anômalos e não reage ao serem identificados tais eventos.

Dados extraídos dos cabeçalhos IP e TCP dos pacotes de rede são coletados e armazenados em arquivos para a obtenção de atributos, os quais são utilizados como entrada para os módulos de detecção do sistema. Esses atributos do tráfego de rede são examinados em busca de anomalias e assinaturas (padrões de ataques). Nove atributos de rede foram utilizados neste trabalho, incluindo: tamanho médio dos pacotes recebidos pelo cliente, tamanho médio dos pacotes recebidos pelo servidor, número de pacotes recebidos pelo cliente, número de pacotes recebidos pelo servidor, porcentagem de pacotes pequenos, direção do tráfego, total de dados recebidos pelo cliente, total de dados recebidos pelo servidor; duração da sessão.

O sistema foi desenvolvido com o propósito de realizar análises de sessões HTTP, mais especificamente das que utilizam a porta 80 da estação servidora Web, e detecção de ataques que envolvem uma única ou algumas sessões, mas pode ser modificado para análise de outros protocolos de aplicação e para detecção de ataques que envolvem múltiplas sessões.

A arquitetura do sistema de detecção de anomalias desenvolvido é composta de sete principais módulos apresentados na Figura 3.2, incluindo: “Módulo de Captura de Pacotes”, “Módulo de Reconstrução de Sessões”, “Módulo de Extração de Atributos”, “Módulo de Detecção de Assinaturas”, “Módulo de

Detecção de Anomalias”, “Módulo de Alerta de Ataques” e “Módulo de Representação Gráfica do Tráfego”.

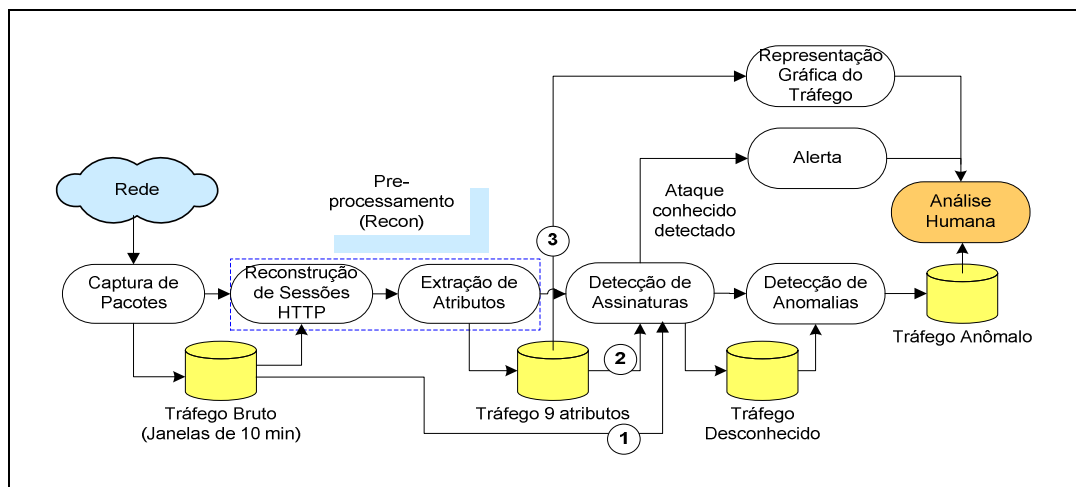


Figura 3.2 - Arquitetura Modular do Sistema ADTRAF  
Fonte: Silva (2007)

Com o sistema ADTRAF é possível realizar a detecção de assinaturas a partir de dados do cabeçalho dos pacotes, utilizando o tráfego reduzido para sessões de 9 atributos; em seguida, é possível realizar a detecção de anomalias sobre o tráfego desconhecido, ou seja, que passou pelo detector de assinaturas como tráfego normal. O sistema ainda permite a análise visual do comportamento do tráfego de rede.

#### e) **ADWICE** - *Anomaly Detection With fast Incremental Clustering*

O sistema de detecção ADWICE proposto pelos autores Berbeck e Tehrani (2007) utiliza mecanismos de grade de índice para melhorar o desempenho de detecção de anomalias, preservando a eficiência da busca. A evolução adaptativa do modelo caracterizado da normalidade do tráfego de rede incrementa novos elementos de comportamento normal e permite o esquecimento de elementos de comportamento ultrapassados.

O modelo característico do tráfego padrão do ADWICE é gerado a partir de técnica de clusterização inspirada no algoritmo BIRCH e adaptado com novas funcionalidades pelos autores. O conjunto de dados analisado é proveniente de dois contextos: uma rede de teste emulada na operadora de telecomunicações

mais importante da Europa (Swisscom) para avaliar a escalabilidade do algoritmo e os dados de ataques disponíveis no KDD-Cup 99.

O ADWICE foi desenvolvido como uma máquina de detecção de anomalia (núcleo) para uma instância de um agente Safeguard e demonstrado no domínio de telecomunicações. Nos últimos anos, o algoritmo foi melhorado e desenvolvido. Safeguard foi um projeto de pesquisa europeu desenvolvido nos anos de 2001 a 2004, cujo objetivo foi melhorar a sobrevivência das infraestruturas críticas por meio de tecnologia baseadas em agentes. A arquitetura do agente de Safeguard é apresentada na Figura 3.3:

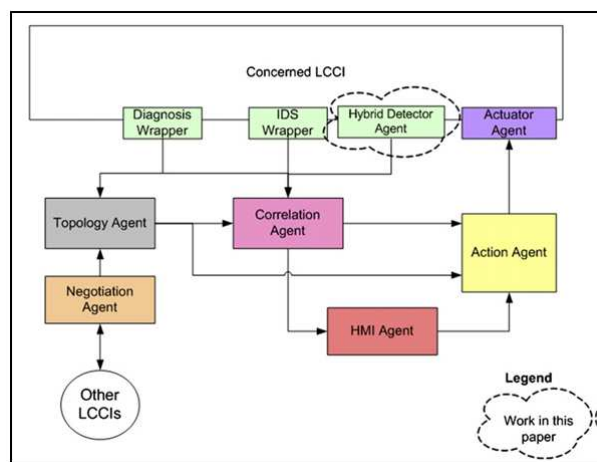


Figura 3.3 – Arquitetura do Safeguard  
Fonte: Berbeck e Tehrani (2007)

O modelo apresenta uma instância do agente detector Hybrid. A noção de híbrido vem da combinação do ADWICE junto com uma lista branca (comportamento normal), ou seja, uma especificação simples baseada em elemento, desenvolvida para detectar anomalias. Consiste em uma série de parâmetros, e um índice de árvore em que as folhas contêm os *clusters*. Utilizou-se a idéia central de BIRCH, ou seja, armazenar apenas as informações condensadas em vez de todos os pontos de dados. Um recurso de *cluster* (*Cluster Feature* – *CF*) é uma tripla  $CF = (n, S, SS)$  onde  $n$  é o número de pontos de dados,  $S$  é a soma linear dos pontos de dados  $n$  e  $SS$  é a soma dos quadrados de todos os pontos de dados. Dado  $n$  vetores de dados  $d$ -dimensional (denominados  $v_i$ ) e um *cluster*  $CF$  representante  $\{v_i \mid i = 1, \dots, n\}$ , o

centróide  $v_0$  (equação 3.1) e o raio  $R$  ( $CF$ ) (equação 3.2) são apresentados como:

$$v_0 = \sum_{i=1}^n v_i / n \quad (3.1)$$

$$R(CF) = \sqrt{\sum_{i=1}^n (v_i - v_0)^2} / n \quad (3.2)$$

onde  $R$  é a distância média dos pontos membros do *cluster* ao centróide e é uma medida da proximidade em torno do centróide.

O ADWICE utiliza os 41 atributos de sessões bem como o conjunto de dados de intrusos do KDD-Cup 99. Apesar das deficiências dos conjuntos de dados relacionados com a DARPA (Mahoney, Chan, 2003; McHugh, 2000) estes dados têm sido utilizados em pelo menos vinte trabalhos de pesquisa e são no momento os únicos conjuntos de dados disponíveis.

De acordo com os autores o ADWICE é um detector de anomalia geral que pode ser aplicado a dados multi-dimensionais a partir de outras infra-estruturas críticas. Aplicação do ADWICE em novos domínios, e em particular, sobre dados de sensores dos sistemas de gestão da água, é um assunto em estudo.

#### **f) Detecção de anomalias de volume usando evolução de classificador negativo**

O trabalho de Azzini et al. (2008) propõe uma técnica de detecção de anomalias de volume de tráfego baseada em redes neurais e algoritmos genéticos, usando critério de seleção negativa.

O mecanismo de seleção negativa é perfeitamente natural nos sistemas imunológicos. A abordagem de classificação negativa para detectar anomalias no tráfego de rede foi originalmente publicada como o algoritmo de seleção negativa, modelado a partir de um método utilizado pelo sistema imunológico para evitar a auto-imunidade. Os autores propõem enfrentar o problema de caracterizar a ampla gama de diferentes formas de anomalias de volume empregando redes neurais evolutivas, treinadas com o tráfego de rede normal, como detectores de positivo e gerenciá-los para a classificação negativa.

Todo o processo pode ser explicado da seguinte forma: é utilizado um mecanismo de classificação negativo, onde os classificadores são treinados para reconhecer a situação normal e são capazes de reagir quando encontram uma anomalia, pois esta não é reconhecida como parte do seu conhecimento.

Como os perfis de ataques são mais complexos e variáveis em relação aos perfis de situação normal, esta abordagem permite uma representação compacta das informações necessárias, tornando a técnica desenvolvida mais eficiente e robusta.

#### **g) ISABA – *Algoritmo Adaptativo Auto Bayesiano***

O ISABA é um SDI desenvolvido por Farid e Rahman (2010) para detecção de anomalias no tráfego de rede, tendo como meta a redução dos alertas falsos positivos. Desempenha atividades tais como: monitoramento das atividades dos usuários, monitoramento das atividades de sistema, identificação das atividades normais, reconhecimento de ataques e armazenamento de informações sobre intrusos.

O algoritmo Baysiano oferece uma abordagem probabilística para a classificação do tráfego, fornecendo uma maneira de prever a classe de uma nova instância. Inspirados neste algoritmo, os autores propuseram o ISABA e testaram com conjuntos dados de ataques do KDD-Cup 99.

Dado um conjunto de dados de treinamento, o ISABA é inicializado com os pesos  $w_i$  definidos como 1.0 e as estimativas de probabilidade  $p$  para cada classe. Os pesos são somados de acordo com a frequência que cada classe ocorre nos dados de treinamento. Para cada atributo  $a_{ij}$  de treinamento é contado o número de ocorrências através da soma dos pesos para determinar a probabilidade  $P(a_{ij})$ . Da mesma forma, a probabilidade  $P(a_{ij} | C_j)$  pode ser estimada pela soma dos pesos de quantas vezes cada valor de atributo ocorre na classe dos dados de treinamento.

Os vetores de entrada de treinamento podem ter muitos atributos diferentes. A classe com maior probabilidade é então escolhida e os pesos de cada exemplo dos dados de treinamento são atualizados com o valor mais elevado da

probabilidade  $P(a_{ij})$ . Se nenhum exemplo dos dados de treinamento é erroneamente classificado, o ISABA calcula nova probabilidade  $P$  utilizando os pesos atualizados para novamente classificar os exemplos de treinamento e atualizar os seus pesos. Este ciclo continua até que todos os exemplos do treinamento sejam classificados corretamente.

Após classificar os exemplos de treinamento, o ISABA classifica os exemplos de teste utilizando as probabilidades condicionais  $P(a_{ij} | C_j)$ . Se algum exemplo do teste é erroneamente classificado, o ISABA atualiza novamente os pesos do treinamento. Compara cada um dos exemplos de teste com todos os exemplos de treinamento e calcula a similaridade entre eles, considerando similaridade de 0,5 e depois de 0,25 entre os valores de atributos. Então os pesos dos exemplos de treinamento são alterados multiplicando-os pela medida de similaridade correspondente. As iterações continuam até que todos os exemplos de teste sejam corretamente classificados. Se todos os exemplos de teste estão corretamente classificados, o ISABA armazena probabilidades condicionais para a futura classificação de intrusões.

O conjunto de dados do KDD-Cup 99 utilizado foi coletado de forma aleatória e representa os valores dos atributos do fluxo de rede de uma classe, rotulada como normal ou com algum tipo específico de ataque. Esses ataques são classificados em: conexões normais geradas pelo comportamento do usuário, DoS (*Denial of Service*), U2R (*User to Root*), R2L (*Remote to Local*) e Probing (Varredura).

Nos vários experimentos realizados com o ISABA, primeiramente o conjunto de dados foi testado com o classificador neural Bayesiano e com o ISABA para comparação dos resultados. No primeiro teste foram utilizados 41 atributos para analisar o tráfego. No segundo teste os atributos foram diminuídos para 17 e depois para 12. De acordo com os resultados, os autores relataram que o ISABA é duas vezes mais rápido em treino e teste do que o classificador Bayesiano Naïve (NB). Também destacam que, nos testes utilizando menos atributos, os resultados dos classificadores testados foram melhores em termos de desempenho.

No segundo momento de teste outras técnicas foram comparadas com o ISABA, tais como, Máquina de Vetor de Suporte (SVM), Redes Neurais (NN), Algoritmo Genético (GA), rede Bayesiana (NB). Os resultados comparativos são apresentados na Figura 3.4.

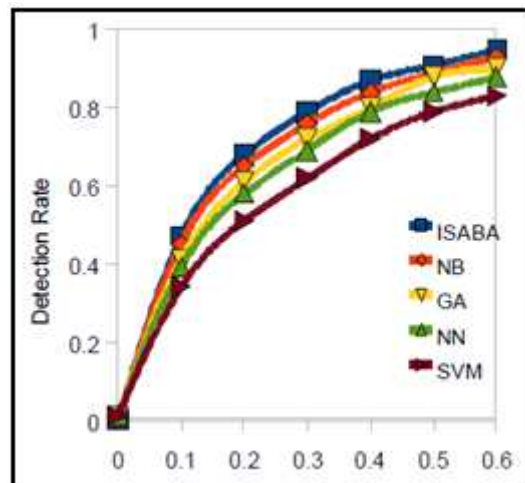


Figura 3.4 – Falsos positivos das técnicas comparadas  
Fonte: Farid e Rahman (2010)

#### **h) Detecção de anomalias de volume usando evolução de classificador negativo**

No trabalho de Celenk et al. (2010), uma abordagem para observar atributos de rede em curto prazo foi apresentada. As alterações significativas dos atributos de rede foram detectadas utilizando filtro adaptativo de Wiener e modelagem de média auto-regressiva. Desta forma, primeiro a entropia média de cada atributo foi calculada para cada segundo de observação. Estas medidas a base de uma nova abordagem para estimativa de anomalia baseada na discriminante linear de Fisher (FLD).

O algoritmo proposto foi testado em dados da rede HTTP da Universidade de Ohio em tempo real para detectar anomalias, tais como, vírus, dispositivos de redes mal configurados ou ataques DoS.

Para este fim, analisou-se estatisticamente o fluxo de dados da rede e aplicou-se o filtro de Wiener no tráfego normal para identificar o sinal correspondente das anomalias de rede, caracterizando o fluxo de rede nessa dimensão.

Ao estimar a função de autocorrelação do tráfego normal, a técnica de previsão ARMA foi aplicada usando a técnica de regressão Yule-Walker.

Para caracterizar a rede 11 atributos foram considerados, consistindo em protocolo, taxas de destino, origem e destino da carga, origem e destino dos *bytes*, porta de origem e destino e endereços IP de origem e destino. Dispositivos de controle no backbone da rede coletam essas informações em tempo real, salvando em disco para análise estatística e visualização quase em tempo real.

A fim de processar os dados como uma série temporal, os registros foram coletados de um mesmo período. O tempo para iniciar a janela de visualização do tráfego, foi empiricamente definido em 3 segundos e gradualmente aumentado (1 em 1 segundo) até o sistema detectar anomalia no tráfego. Desta forma, o período de tempo para a exploração da rede depende de vários fatores, tais como, a configuração do tempo da janela de visualização.

O SDI alerta visualmente os analistas sobre uma anomalia de tráfego de rede, permitindo que o analista possa capturar rapidamente e compreender as características estatísticas do evento, enquanto o software examina grandes quantidades de dados da rede sem a necessidade de interação humana. Quando uma anomalia é visualmente indicada, o analista pode decidir se o caso merece uma investigação mais aprofundada ou não.

Os experimentos realizados pelos autores têm mostrado que o SDI apresentado é capaz de identificar anomalias nos atributos de rede selecionados, incluindo porta média, porta alta, a porta do servidor, entre outros.



#### **4 A METODOLOGIA TRAFKIN PROPOSTA**

O trabalho aqui apresentado consiste em propor a metodologia TRAFKIN desenvolvida à partir da combinação de técnicas estatísticas e de Inteligência Computacional.

A TRAFKIN tem como objetivo propor um conjunto de técnicas e procedimentos para o processo de caracterizar o comportamento padrão do tráfego, que possa se tornar uma referência para atividades de detecção de anomalias em ambientes de redes operacionais. A idéia da TRAFKIN é criar modelos do comportamento padrão do tráfego de rede, compactos e significativos (grande valor semântico) dos dados espaço-temporais do fluxo de rede observado. A partir do modelo do tráfego caracterizado, é possível treinar adequadamente os SDI para detectar anomalias. A redução de dados elimina redundâncias, podendo equilibrar os classificadores de maneira a minimizar as taxas de alarmes falsos.

Nos testes realizados, a detecção de anomalias foi alcançada pela caracterização do tráfego de rede através da técnica de clusterização adotada para extração do conhecimento e redução da base de dados mantendo a expressividade da informação. Também foram observadas taxas pequenas de alarmes falsos.

Os processos de desenvolvimento e de avaliação da metodologia descritos são especificados e testados utilizando conjuntos de dados reais do tráfego HTTP da rede local de computadores localizada no prédio Beta (RedeBeta), do Instituto Nacional de Pesquisas Espaciais em São José dos Campos (INPE), e dados sintéticos de tráfego anômalo gerados no ambiente do Laboratório de Redes da Divisão de Desenvolvimento de Sistemas de Solo (LabRedes-DSS) do INPE.

Sabendo-se que a remodelagem do perfil do tráfego deve ser realizada com certa periodicidade para acompanhar a dinâmica do comportamento do tráfego, objetiva-se que a TRAFKIN seja uma metodologia que agiliza a construção do modelo do comportamento do tráfego.

Possíveis benefícios provenientes da aplicação desta metodologia para a construção do modelo do tráfego padrão incluem: redução drástica da base de dados para o treinamento do classificador, taxas baixas de falsos positivos ou alarmes falsos.

A TRAFICIN foi desenvolvida em quatro etapas consecutivas: pré-processamento, modelagem e mineração de dados, caracterização do tráfego padrão e detecção de anomalias. Os processos que a compõem são apresentados na Figura 4.1 e descritos nas próximas seções, juntamente com a descrição das técnicas e ferramentas utilizadas.

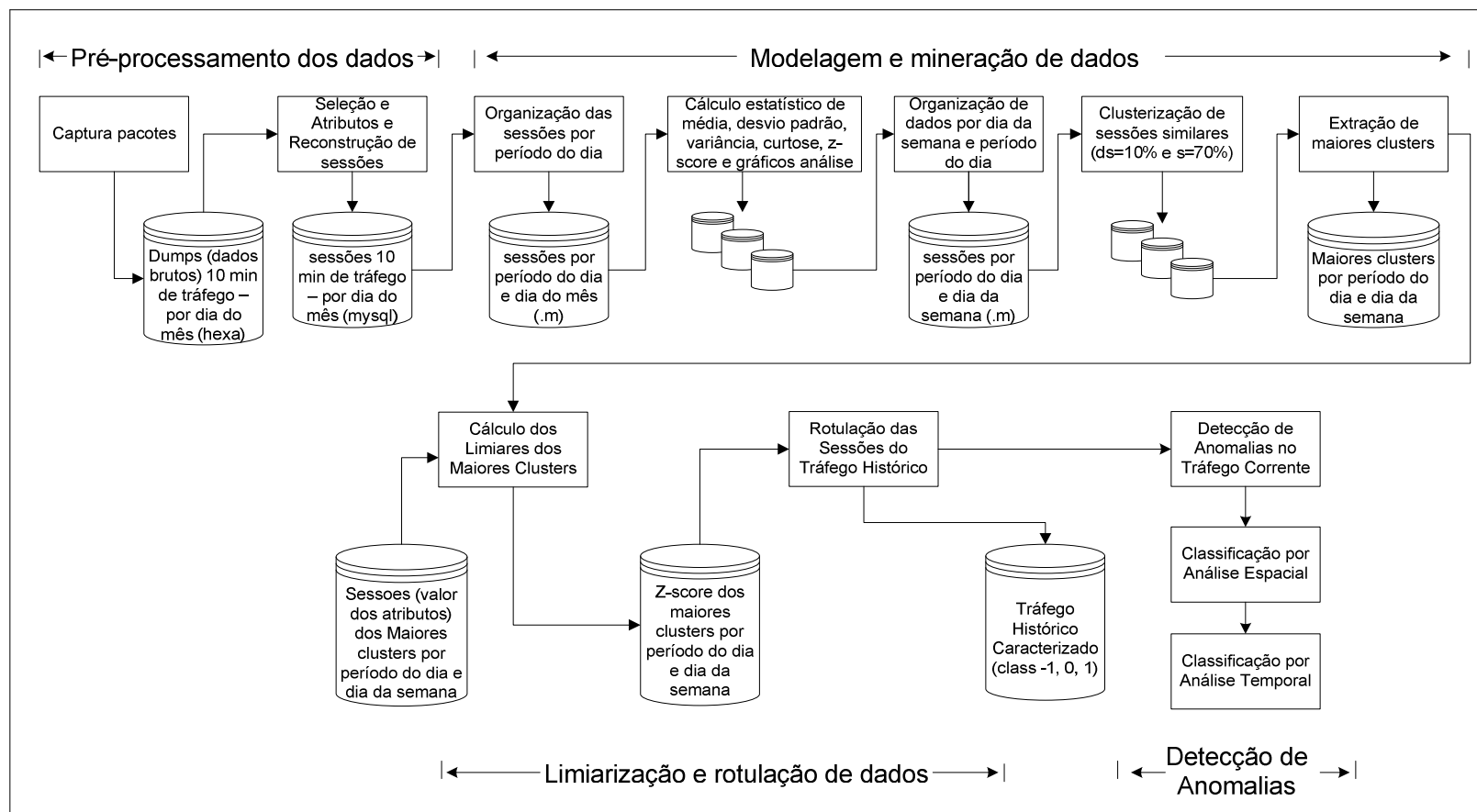


Figura 4.1 - Diagrama da metodologia TRAF CIN

#### 4.1. Pré-processamento dos Dados

Na etapa de pré-processamento, os dados brutos do tráfego de rede são coletados e recebem os primeiros processamentos para extração de informações relevantes para análise. A Figura 4.2 apresenta o diagrama de atividades nesta fase.

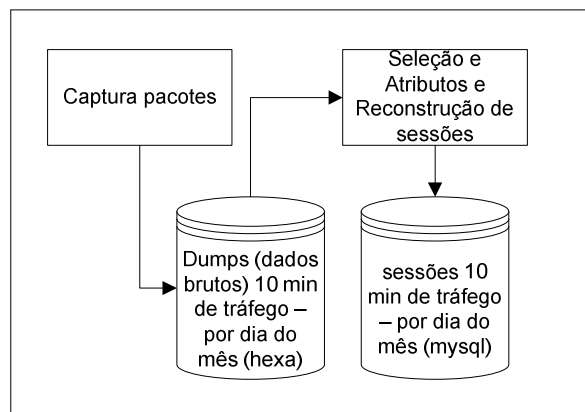


Figura 4.2 - Etapa de pré-processamento de dados

As principais atividades realizadas na fase de pré-processamento dos dados são: *coleta de dados*, *seleção de atributos* e *reconstrução de sessões*.

O intervalo de tempo e a frequência de captura dos dados da rede resultaram em uma grande base de dados de sessões de tráfego, de uma rede real, suficiente para atender aos objetivos do estudo do tráfego ao longo do tempo, com base na hipótese de que a característica normal de um tráfego de rede só pode ser estabelecida com o monitoramento contínuo do uso do ambiente por um intervalo de tempo considerável.

Diferentemente de vários trabalhos relatados na literatura, que utilizam dados para análise em intervalos de tempo limitados a 1 dia, horas de um dia, até 4 semanas, nesta tese utilizou-se os dados disponíveis de 3 meses,

correspondentes ao período de 05/03/2010 a 08/06/2010, dos quais 2 meses (05/03/2010 a 06/05/2010) foram utilizados na análise, caracterização do comportamento e treinamento dos dados e 1 mês (07/05/2010 a 08/06/2010) foi aplicado na fase de testes dos classificadores. Para a caracterização do tráfego de rede derivada na tese considerou-se os dados de tráfego real disponíveis, restritos a um período de três meses.

Coletar os dados, desenvolver estratégias para armazenamento de grandes volumes de dados, bem como controlar *backups* destes e gerar o tráfego anômalo foram os grandes desafios para a realização desta etapa de trabalho.

#### 4.1.1. Coleta de Dados

Para este trabalho, um sensor de SDI foi configurado fora dos limites de proteção do *firewall* para permitir a coleta e observação dos pacotes de rede dirigidos ao *firewall* e também à rede interna. A porta do *firewall* da RedeBeta foi espelhada no *switch* da RedeBeta para a porta do Sensor SDI, a fim de capturar, através do sensor IDS, todo o tráfego de entrada e saída desta rede. A Figura 4.3 ilustra o ambiente de coleta de dados reais do tráfego da RedeBeta.

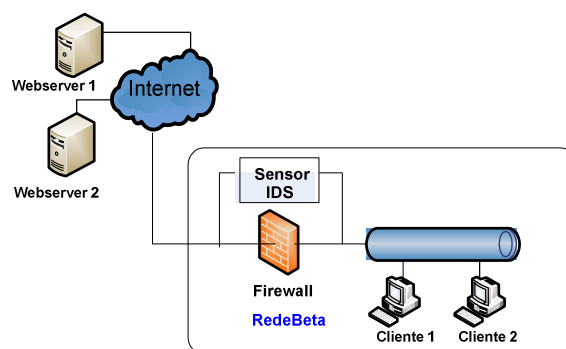


Figura 4.3 - Ambiente de coleta de dados reais

O *software* utilizado para a captura dos pacotes de rede foi o tcpdump, que utiliza a biblioteca libpcap (*Packet Capture Library*) para a comunicação com o dispositivo de rede, e coloca a interface de rede da estação de captura em modo promíscuo, de modo que todo o tráfego que passe pelo segmento de rede, seja capturado.

Os pacotes IP do tráfego de rede foram capturados através do comando:

```
# tcpdump -nv -i eth0 -s 1500 -w /var/log/dumps/arquivo.dump (ip and not udp port 520)
```

Através do *script* “*rotatedump*”, executado continuamente no sensor SDI, realizou-se a captura dos pacotes de rede em intervalos de 10 em 10 minutos, os quais foram armazenados em disco, no processo de rotacionamento de logs.

De acordo com a estratégia de detecção de anomalias adotada é escolhido o intervalo ou janela de tempo para a captura dos pacotes. O conjunto de dados utilizado para modelagem do tráfego foi previamente sanitizado para remoção de eventos ilegítimos através do SDI Snort (FOSTER et al., 2003).

Dados sintéticos de tráfego de ataque foram produzidos no Laboratório da Divisão de Desenvolvimento de Sistemas de Solo (LabRedes-DSS) para simular anomalias na rede. Estes dados são referenciados ao longo deste trabalho como “dados de tráfego anômalo ou anomalias”. A produção dos dados de tráfego anômalo sintéticos foi realizada através do lançamento de ataques contra máquinas no Laboratório de modo controlado.

Os pacotes de rede do tráfego interno do LabRedes (comunicação entre máquinas clientes e servidora internas) foram capturados, conforme ilustrado na Figura 4.4.

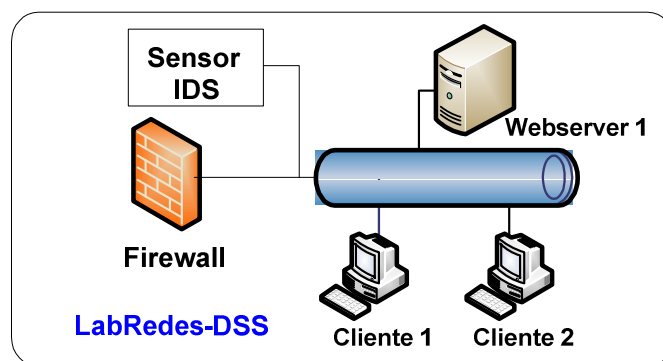


Figura 4.4 - Ambiente de coleta de dados anômalos

Através de duas estações clientes da rede interna do Laboratório foram lançados ataques do tipo DoS (negação de serviço) e Probing (varredura de portas) contra o servidor Web residente nesta rede para produzir dados de tráfego anômalo para análise. Para isto, foram utilizados *exploits* obtidos em endereços na internet, tais como Milworm<sup>1</sup>, Security Focus<sup>2</sup> e Packetstorm<sup>3</sup>. Exploits é um termo genérico para descrever pequenos utilitários ou exemplos de códigos que podem ser usados para explorar vulnerabilidades específicas no *host* ou rede.

O *software* de captura de tráfego *tcpdump* é executado e, posteriormente o ataque é lançado. Após lançado o ataque, o tráfego malicioso é capturado. Após o término de execução do ataque, a execução do *tcpdump* é interrompida. O ciclo de atividades se encerra quando todos os ataques foram disparados contra o alvo (FAGUNDES, 2002).

As sessões anômalas associadas aos ataques, ou seja, provenientes das comunicações ilegítimas entre as estações clientes e a servidora web, foram, então, reconstruídas e armazenadas no banco de dados MySQL através do

---

<sup>1</sup> <http://www.milworm.com>

<sup>2</sup> <http://www.securityfocus.com>

<sup>3</sup> <http://www.packetstorm.com>

sistema Recon. Uma ilustração do ciclo de atividades de coleta de tráfego anômalo é apresentada na Figura 4.5.

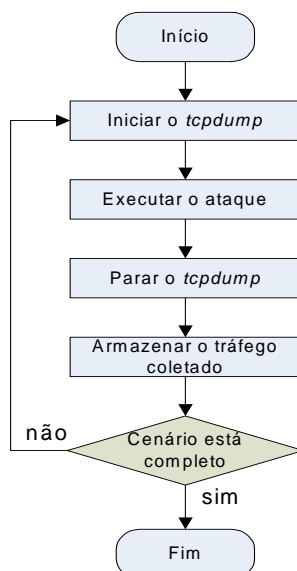


Figura 4.5 – Ciclo de atividades de coleta de tráfego anômalo.  
Fonte: Adaptado de Fagundes (2002)

Na Tabela 4.1 são apresentados os “*exploits*” lançados contra a servidora alvo.

Tabela 4.1 – “Exploits” utilizados para geração de anomalias sintéticas

Tipo	Exploits
Probing	Ipscan, probe_tcp_ports, ttcp, nmapscan, httpscan, portscan
DoS	synflood, synful, killwin, kkill, sumo, syndrop

#### 4.1.2. Seleção de Atributos

Na TRAFKIN, nove atributos de rede, descritos na tabela 4.2, foram selecionados para o propósito de caracterização do tráfego de rede. Estes atributos foram utilizados dando continuidade à linha de estudo dos autores



Chaves (2002), Silva et al. (2004), Silva et al. (2005a), Silva et al. (2005b), Silva et al. (2006a), Silva et al. (2006b), Silva (2007), levando-se em consideração que traços de ataques DDoS e de varredura, muito comuns na atualidade, podem ser detectados de modo satisfatório a partir destes atributos.

Tabela 4.2 - Descrição dos atributos utilizados para representar cada sessão

Nome do Atributo	Descrição	Classe
psizeCL ( <i>bytes</i> )	Tamanho médio dos pacotes recebidos pelo cliente	2
psizeSV ( <i>bytes</i> )	Tamanho médio dos pacotes recebidos pelo servidor	2
pnumCL	Número de pacotes recebidos pelo cliente	2
pnumSV	Número de pacotes recebidos pelo servidor	2
smallpkt	Porcentagem de pacotes pequenos	2
dataDIR	Direção do tráfego	2
brecvCL ( <i>bytes</i> )	Total de dados recebidos pelo cliente	2
brecvSV ( <i>bytes</i> )	Total de dados recebidos pelo servidor	2
Duration	Duração da sessão	1

A descrição do cálculo dos nove atributos derivados utilizados encontra-se a seguir:

- **psizeCL:** valor calculado através de uma média aritmética da quantidade de dados (em *bytes*) da carga útil dos pacotes recebidos pelo cliente em uma sessão;
- **psizeSV:** valor calculado através de uma média aritmética da quantidade de dados (em *bytes*) da carga útil dos pacotes recebidos pelo servidor em uma sessão;

- **pnumCL**: quantidade de pacotes recebidos pelo cliente em uma sessão;
- **pnumSV**: quantidade de pacotes recebidos pelo servidor em uma sessão;
- **smallpkt**: valor calculado através da soma do número de pacotes com uma quantidade de dados inferior a um dado limiar (130 *bytes*), seguido da divisão da soma pelo total de pacotes de uma sessão;
- **dataDir**: valor inicializado com 0 (zero). Para cada pacote recebido pelo servidor na sessão, o valor é subtraído de uma unidade, e para cada pacote recebido pelo cliente, o valor é incrementado de uma unidade;
- **brecvCL**: valor calculado através da soma da quantidade de dados (em bytes) de todos os pacotes recebidos pelo cliente em uma sessão;
- **brecvSV**: valor calculado através da soma da quantidade de dados (em bytes) de todos os pacotes recebidos pelo servidor em uma sessão;
- **duration**: valor (em segundos) calculado pela diferença entre o momento em que o último pacote da sessão foi capturado (*timestamp* do ultimo pacote) e o momento em que o primeiro pacote foi capturado (*timestamp* do primeiro pacote).

#### 4.1.3. Reconstrução de Sessões

Para reconstruir as sessões do tráfego de rede do conjunto de dados de interesse foi utilizado o sistema Recon (Sistema de Reconstrução de Sessões TCP/IP) desenvolvido por Chaves (2002) utilizado no trabalho de Silva (2007).

O sistema Recon reconstrói e rastreia o estado das sessões TCP/IP, a partir de um modelo gerado dos dados extraídos dos cabeçalhos e conteúdo dos pacotes da pilha de protocolos TCP/IP. O Recon lê um arquivo em formato tcpdump e reconstrói as sessões ICMP, TCP e UDP entre os pares de endereços IP, de acordo com o protocolo de aplicação determinado, por exemplo HTTP (porta 80). As sessões do tráfego reconstruídas são

organizadas e facilmente obtidas, o que torna esta ferramenta de base uma opção adequada para caracterizar o tráfego padrão de rede e detectar anomalias a posteriori (Chaves, 2002; Silva et al., 2004; Silva, 2007).

O seguinte comando foi utilizado para reconstruir as sessões HTTP do tráfego de rede armazenadas em arquivos *.dump* e armazená-las em base de dados MySQL:

```
# recon -H -r /var/log/dumps/arquivo.dump
```

Este comando reside no *script* desenvolvido “autoRecon.sh” para executar a reconstrução de sessões de arquivos de um dia, organizados em janelas de 10 minutos.

Após a reconstrução das sessões em base MySQL, os dados da base são copiados para arquivos de formato reconhecido com dados organizados por dia do mês e período do dia.

A conclusão desta fase ocorre com o fornecimento da base de dados do tráfego de rede armazenada na base MySQL por dia do mês.

#### **4.2. Modelagem e Mineração de Dados**

Devido ao grande volume de dados de tráfego gerado na rede e da diferença sutil existente entre alguns dos eventos normais e anômalos, a caracterização do comportamento padrão do tráfego de rede é uma tarefa que exige um esforço significativo de mineração de dados.

Dados de 2 meses de tráfego coletados nos meses de março a abril de 2010 e reconstruídos em sessões foram previamente analisados neste trabalho. Estas sessões do tráfego foram armazenadas na base MySQL em tabelas organizadas por dia do mês (exemplo: S21062010), contabilizando um total de

60 tabelas, cada uma com nove campos. É válido ressaltar que os dados coletados podem diferir muito de um mês para outro, por isso a metodologia TRAFICIN deve ser sintonizada para diferentes meses ou períodos.

Conforme ilustra a Figura 4.6, a etapa de modelagem e mineração de dados é composta por três principais atividades: análise preliminar, modelagem e clusterização dos dados do tráfego de rede. A descrição destas atividades é apresentada nas seções a seguir.

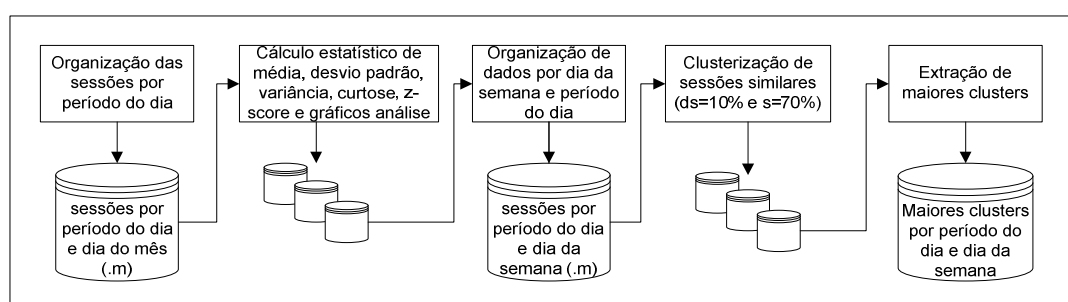


Figura 4.6 - Etapa de modelagem e mineração de dados

#### 4.2.1. Análise Preliminar dos Dados

Nesta fase do presente trabalho, análise preliminar dos dados e técnicas estatísticas foram aplicadas ao conjunto de dados multivariado e de grande volume para explorar possíveis relacionamentos sistemáticos, regularidades, correlações, fatores agrupadores ou diferenciadores entre os atributos do tráfego. Através de tais técnicas, dois modos de análise foram realizados:

- Análise estatística do comportamento dos atributos das sessões, por meio dos cálculos de média, desvio padrão, e curtose; e
- Análise temporal dos dados para identificar se é possível caracterizar e prever os dados de tráfego coletados, ou seja, se existe uma frequência de ocorrência (repetição) no tráfego para caracterizá-los. As técnicas aplicadas nos atributos das sessões de rede foram DFA (*Detrended Fluctuation Analysis*) e PDF(*Probability Density Function*).

Os atributos correspondentes de todas as seções foram avaliados em conjunto através destas técnicas.

A primeira caracterização do tráfego padrão de rede gerada na TRAFICIN, foi extraída de faixas de valores de parâmetros de curtose ( $k$ ) e DFA ( $\alpha$ ). Os valores de atributos indicavam a pertinência a uma sessão normal.

Deste modo, fez-se uma caracterização preliminar do tráfego pela análise individual dos atributos das sessões. A partir do cálculo do expoente de flutuação DFA ( $\alpha$ ) e curtose ( $k$ ), foi encontrada uma faixa de valores padrão para cada atributo, de acordo com o dia semana e período do dia analisados. Valores de atributos dentro da faixa indicam que pertencem a sessões normais.

Pelo fato destas técnicas permitirem caracterizar apenas atributos correspondentes das sessões individualmente e não combinados com os demais atributos da sessão, não foi possível identificar as sessões que produziram os traços de anomalia existentes no tráfego de rede de teste analisado. Todavia, mostrou que determinados atributos selecionados para este trabalho são mais afetados que outros na análise de sessões anômalas.

#### **4.2.2. Modelagem dos Dados**

Através de análise estatística dos dados do tráfego coletados e agrupados por dia do mês, observando os parâmetros de média, desvio-padrão, curtose, PDF e DFA constatou-se a existência de persistência entre os atributos das sessões do conjunto de dados de dois meses de tráfego, ou seja, é possível prever o comportamento futuro do tráfego a partir do comportamento passado.

Nesta etapa a exploração dos dados do tráfego de rede foi aprofundada através da reorganização dos mesmos em períodos do dia, mantendo a estrutura física das tabelas.

O objetivo da reorganização dos dados foi encontrar com maior precisão as anomalias durante períodos do dia, devido à variação grande do fluxo, o qual se modifica ao longo do tempo. A questão a investigar foi: se durante o dia há muitas variações no comportamento do fluxo, será que a noite ou madrugada é igual? A reorganização dos dados aumentou a probabilidade e simplificou o processo de detecção de anomalias, pois não foi mais necessário analisar o volume todo dos dados e sim os dados em períodos de tempo diferentes.

Desta forma, os dados foram reorganizados em quatro períodos do dia para cada dia do mês, mantendo a estrutura física das tabelas com registros de sessões contendo 9 campos de atributos. Os períodos do dia considerados foram:

- P1 (madrugada – 00h às 06h59min);
- P2 (manhã – 07h às 12h59min);
- P3(tarde – 13h às 18h59min);
- P4 (noite – 19h às 23h59min).

Por exemplo, foi gerada a Tabela P00\_07-S21062010 contendo sessões do tráfego coletados na data 21/06/2010 no período de 00h às 07h da manhã (P1). Os dados analisados neste trabalho provêm de uma rede de produção, em uso no INPE. Deste modo, foi necessária a reorganização dos dados em períodos do dia, considerando o horário de trabalho de expediente normal do Instituto (manhã e tarde) e horários noite e madrugada foram escolhidos com base na observação do comportamento dos dados desta rede. Os fatores tamanho dos períodos de tempo (em horas) e a quantidade de dados de tráfego gerada por período (em sessões) definiram o caráter assimétrico dos conjuntos de dados analisados.

Após a análise estatística realizada em cada atributo individualmente, foi utilizada a técnica de clusterização “Mapa de Kohonen Adaptável” (MKA) descrita na Seção 3.4.3, em busca de melhores resultados na extração do conhecimento dos dados de grande volume do tráfego de rede. Como esta técnica analisa a sessão, isto é, analisa os atributos simultaneamente, proporcionou o desenvolvimento de modelo mais realístico dos dados, considerando as características multivariadas do amplo conjunto de sessões descritas por nove atributos.

Para explorar maior similaridade entre as sessões analisadas pelo MKA, fez-se nova reorganização dos dados por dia da semana e período do dia da coleta. Em seguida, aplicou-se a técnica de clusterização MKA novamente sobre os conjuntos de dados agrupados e observou-se um aumento na frequência de ocorrência de padrões nas sessões. Isto nos permitiu inferir que, as sessões das segundas-feiras no horário da manhã têm mais características em comum entre si do que quando comparadas com as sessões de outro dia da semana no mesmo horário.

O conjunto de dados foi reorganizado considerando o período do dia e, também, o dia da semana em que as sessões foram coletadas, gerando assim arquivos de dados, como por exemplo, “P00\_07-Quin”, contendo dados do período de 00h a 07h de todas as quintas-feiras de dois meses de tráfego.

#### **4.2.3. Clusterização do Tráfego**

Para tratar o grande volume de dados de tráfego de rede, no presente trabalho, a abordagem “Mapa de Kohonen Adaptável” (MKA) foi utilizada para a clusterização das sessões do tráfego de dois meses de dados, por ser de rápido processamento e simples implementação. O MKA é uma rede neural SOM (*Self Organization Map*) baseado em matriz de pesos estimada a partir dos valores de cada atributo do vetor a ser clusterizado e considerando-se a

distribuição de valores destes atributos nas sessões de cada *cluster*. Esta matriz de peso denomina-se  $W$ , com dimensão  $n$  para armazenar os vetores representantes de cada *cluster* descoberto. Os dois principais atrativos deste esquema são: abstração do vetor representante de cada *cluster* e não o valor médio (a maioria dos algoritmos de clusterização, inclusive os que utilizam a rede neural SOM, buscam o centro como representante de cada *cluster*) e flexibilidade na formação dos *clusters* através de dois parâmetros: taxa de desvio ( $ds$ ) aplicada a cada um dos 9 atributos analisados e taxa de similaridade ( $sim$ ) aplicada a sessão do tráfego histórico.

Os passos para a implementação da abordagem de mineração por clusterização MKA são apresentados a seguir.

Seja  $X=[x_1, x_2, \dots, x_n]$  um vetor de sessões  $x_j$  ( $j=1..n$ ) contendo nove atributos a ser classificado como pertencente a um determinado *cluster* e  $n$  o número de sessões apresentadas para clusterização. Para cada variável  $x_j$  de  $X$  (ou seja, para cada  $j$ ) são realizados os seguintes passos:

**Passo 1:** Para cada sessão  $x_j$  ( $j=1..n$ ) apresentada ao clusterizador, o valor do atributo  $x_j$  é ponderado com os valores do atributo correspondente de cada uma das  $p$  linhas de  $W$ , ou seja, percorre todos os vetores representantes de cada *cluster* armazenados em  $W$  (equivalentes aos neurônios de Kohonen), gerando o vetor  $d_j = x_j / w_{pi}$ .

**Passo 2:** Considerando-se as  $p$ -ésimas linhas de  $W$ , os valores de  $d_j$  são comparados com o intervalo  $1 \pm ds_j$ ;

**Passo 3:** Em seguida, é calculado o total de valores ( $\text{soma}_j$ ) de  $d_j$  que satisfazem o intervalo, onde  $ds_j$  corresponde a um limiar de desvio do atributo  $a_j$ .



**Passo 4:** A soma obtida é comparada com a taxa de similaridade definida  $sim$  e, se  $soma_j \geq sim \cdot 9$ , a sessão  $x_j$  é classificada como pertencendo ao *cluster* corrente;

**Passo 5:** Caso contrário ( $soma_j < sim$ ), uma nova linha é criada na matriz recebendo o vetor sob análise, ou seja,  $W_{p+1} = \{x_j\}$ , o vetor  $x$  passa a ser o vetor representante de um novo *cluster* na matriz  $W$ .

**Passo 6:** Repetem-se os passos de 1 a 5 até que todas as sessões do vetor  $X$  tenham sido clusterizadas.

A abordagem acima descrita agrupa cada sessão do tráfego ao seu *cluster* mais similar, sendo este *cluster* parte do espaço amostral analisado.

Visto que anomalias podem deixar traços em um ou mais atributos diferentes da sessão, é pertinente que estes sejam analisados em conjunto para detectar anomalias com maior precisão e melhor desempenho. A idéia da análise de similaridade entre as sessões do tráfego é explorada nesta abordagem para a formação dos *clusters*, sem, no entanto, deixar de analisar os atributos da sessão multivariada.

Após vários experimentos com o conjunto de dados do tráfego utilizando valores de parâmetros de  $ds$  e  $sim$  diferentes, decidiu-se por continuar o trabalho aplicando os seguintes parâmetros de flexibilidade na abordagem de clusterização:  $ds= 10\%$  e  $sim=70\%$ .

Com a abordagem MKA foi possível diminuir significativamente o volume de dados a ser analisado para a caracterização do comportamento padrão do tráfego de rede, sem, no entanto, perder informação útil.

### 4.3. Limiarização e Rotulação de Dados

Para a etapa de limiarização e rotulação de dados do tráfego em classes pré-definidas, três hipóteses foram admitidas:

**Hipótese 1:** Os *clusters* mais populosos representam o comportamento padrão do tráfego, pois as sessões destes *clusters* são mais comuns (maior frequência de ocorrência);

**Hipótese 2:** Pelo menos 90% dos dados coletados nestes dois meses de tráfego são padrão, uma vez que o conjunto de dados utilizado para modelagem do tráfego foi previamente sanitizado para remoção de eventos ilegítimos;

**Hipótese 3:** Possíveis desvios no perfil de comportamento padrão do tráfego caracterizado são consideradas como anomalias, sessões anômalas ou eventos de comportamento anômalo.

Nesta etapa de desenvolvimento da metodologia, ilustrada na Figura 4.7, dois principais processos foram conduzidos: cálculo dos limiares que determinam o tráfego de rede padrão e anômalo baseada em *z-score* e rotulação das sessões do tráfego histórico.

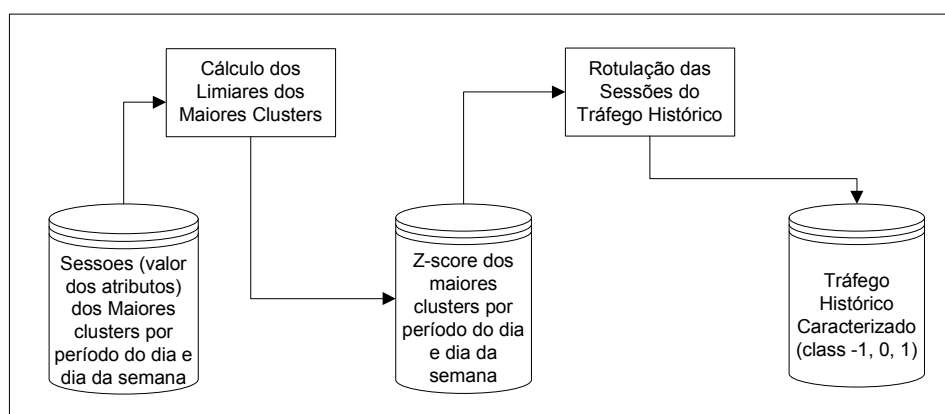


Figura 4.7 – Etapa de limiarização e rotulação de dados

Uma das atividades essenciais nesta etapa, que influencia diretamente na precisão da caracterização do tráfego de rede, consiste em descobrir a melhor forma de abstrair informações relevantes para a construção do perfil do tráfego.

Partindo-se da *hipótese 1*, o próximo passo da metodologia TRAF CIN foi selecionar, em cada conjunto organizado por dia da semana e período do dia, apenas as sessões dos maiores *clusters* para caracterizar efetivamente o comportamento padrão do tráfego. Com isto, foi possível eliminar as sessões do tráfego que se comportaram de modo diferente da maioria e o volume de dados do tráfego a ser processado foi reduzido drasticamente.

Dos conjuntos analisados, foram escolhidos entre 20 e 2000 maiores *clusters*, devido à quantidade de sessões existentes nestes agrupamentos, que se mostrou extremamente significativa em relação aos demais. A escolha dos maiores *clusters* foi com base na observação dos gráficos gerados, ilustrando a quantidade de sessões por *cluster* e nos relatórios contendo o total de *clusters* de cada conjunto e o total de sessões em cada *cluster*. Convém ressaltar que os primeiros *clusters* de todos os conjuntos continham aproximadamente 80% das sessões. O restante dos *clusters* representa uma porcentagem muito pequena de sessões (aproximadamente 0.24 a 0.04%).

Após selecionar as sessões do tráfego relevantes dos maiores *clusters*, outra atividade relevante foi definir quais seriam os limiares para rotulação dos dados, que permitiriam separar as sessões do tráfego da rede em três classes: (-1) anômala abaixo da média, (0) padrão e (1) anômala acima da média. Estas classes foram definidas através de análise gráfica dos valores de *z-score* (transformada Z) dos atributos das sessões.

Na metodologia TRAF CIN, adotou-se o cálculo da limiarização baseada em *z-score* para rotular as sessões selecionadas dos maiores *clusters* a uma dada classe ou grupo, a partir do cálculo da média e desvio padrão do conjunto de

dados de dois meses de tráfego histórico coletado. O cálculo aplicado em cada um dos nove atributos das sessões do tráfego foi:  $z_j = (a_j - medh) / dph$  ( $j=1, \dots, 9$ ), onde  $a_j$  é o atributo da sessão,  $medh$  e  $dph$  são, respectivamente, a média e o desvio padrão históricos de cada atributo do tráfego analisado.

Após o cálculo do *z-score* para todo o conjunto analisado, foram calculados e armazenados os valores de *z-score* mínimos e máximos de cada um dos nove atributos das sessões, de acordo com o período do dia e dia da semana. O intervalo de valores entre o *z-score* mínimo e o *z-score* máximo de cada atributo define a faixa ou fronteira de normalidade das sessões do tráfego. Sessões fora desta faixa, ou seja, muito acima ou muito abaixo da média de valores do conjunto, devem ser analisadas, pois indicam sinais de anomalias.

Para automatizar a rotulação de dados do tráfego histórico, um sistema baseado em regras de produção foi implementado.

Definidos os limiares e considerando as hipóteses 2 e 3 acima descritas, a rotulação do tráfego consistiu em: comparar o valor de *z-score* dos nove atributos da sessão histórica com os respectivos limiares mínimos e máximos. Se até 90% dos atributos da sessão possuísem os valores de *z-score* menores que os respectivos limiares mínimos, essa seria classificada com o rótulo -1, ou seja, “anômala abaixo da média”. Se até 90% dos atributos da sessão possuísem os valores de *z-score* dentro da faixa dos limiares mínimos e máximos, é classificada com o rótulo 0, ou seja, “padrão”. Caso contrário, se até 90% dos atributos comparados estivessem acima dos respectivos limiares máximos a sessão seria classificada com o rótulo 1, ou seja, “anômala acima da média”. A taxa de 90% de comparação de quantidade de atributos a satisfazer os limites inferiores e superiores da fronteira de decisão de normalidade foi escolhida para proporcionar maior precisão nos resultados.

Duas bases de dados para treinamento do classificador foram geradas nesta etapa do trabalho: arquivos contendo valores de z-score dos atributos necessários para rotulação da base histórica e arquivos de valores reais de atributos rotulados (base histórica efetivamente caracterizada com sessões rotuladas). A Tabela 4.3 apresenta uma amostra dos dados rotulados.

Tabela 4.3 - Sessões aleatórias - valores reais de atributos com rótulos (*class*).

psizeCL	psizeSV	pnumCL	pnumSV	smallpkt	dataDir	brecvSV	brecvCL	duration	class
122.67	442.09	21.00	22.00	0.63	-1.00	2576.00	9726.00	369.01	0
1114.52	161.79	44.00	39.00	0.45	5.00	49039.00	6310.00	7.00	0
1070.03	95.43	101.00	82.00	0.45	19.00	108073.00	7825.00	70.4323	1
1198.00	58.26	208.00	151.00	0.43	57.00	249184.00	8798.00	130.73	1
0.00	0.00	0.00	0.00	0.81	0.00	0.00	0.00	3.62	-1
0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	20.22	-1

A etapa de rotulação dos dados, adicionando a coluna “*class*” nos registros, é a fase final do processo de caracterização do tráfego de rede padrão.

Como resultados desta etapa, têm-se as sessões dos conjuntos de dados do tráfego de rede organizados por dia da semana e período do dia, classificadas como “sessão anômala abaixo da média” (*class* -1), “sessão padrão” (*class* 0) ou “sessão anômala acima da média” (*class* 1).

A conclusão desta etapa ocorre com o fornecimento do modelo de caracterização do tráfego de rede armazenado em arquivos textos.

#### 4.4. Detecção de Anomalias

Uma vez criada a base de conhecimento reduzida e povoada de informações relevantes do tráfego de rede, esta foi utilizada para treinamento dos classificadores selecionados e foram observados o tempo de treinamento e a precisão obtida nos resultados de treinamento e teste dos classificadores.

Na metodologia TRAF CIN, duas abordagens para detectar anomalias nas sessões do tráfego corrente da rede foram adotadas: classificação baseada em análise espacial e classificação baseada em análise temporal, como apresenta a Figura 4.8.

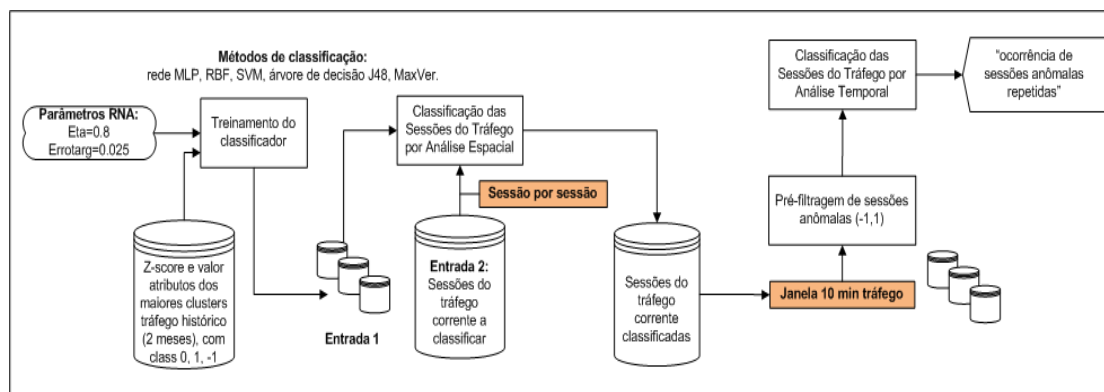


Figura 4.8 – Etapa de detecção de anomalias

No presente trabalho, a classificação espacial do tráfego de rede analisa o conteúdo sessão por sessão, enquanto a classificação temporal do tráfego analisa a frequência de ocorrência de sessões com mesma característica durante um determinado tempo. Nas próximas seções estas duas abordagens de classificação são descritas.

#### 4.4.1. Classificação por Análise Espacial do Tráfego

Na classificação espacial do tráfego cada valor de atributo das sessões correntes é analisado para classificar a sessão como anômala ou padrão. Desta forma, busca-se identificar os traços de anomalias que possam estar registrados nas sessões. Os procedimentos desta abordagem são apresentados na Figura 4.9.

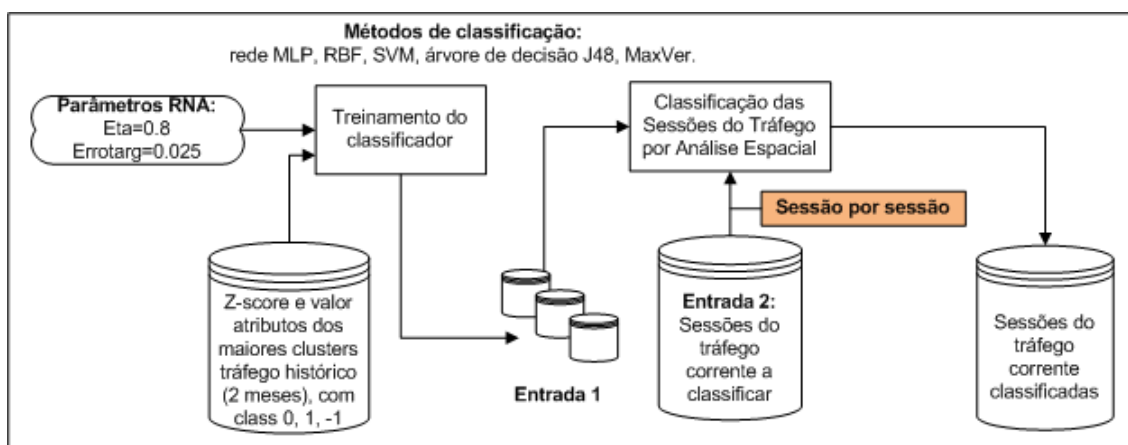


Figura 4.9 – Classificação de dados por análise espacial

A base de dados histórica de dois meses de tráfego de rede caracterizada na Seção 4.3, é utilizada na etapa de treinamento dos métodos de classificação aplicados nesta. Após o treinamento, novas sessões correntes foram aplicadas para teste/validação do TRAFICIN, sendo classificadas como anômala abaixo da média (-1), padrão (0) ou anômala acima da média (1).

A fim de obter uma comparação de resultados em relação a taxas de falsos positivos gerados na detecção de anomalias, vários métodos foram aplicados na classificação do tráfego de rede. Estes métodos foram: rede neural MLP (*MultiLayer Perceptron*), SVM (*Support Vector Machine*), RBF (*Radial Basis Functions*), Árvore de Decisão e Máxima Verossimilhança, os quais foram apresentados no Capítulo 2.

#### 4.4.2. Classificação por Análise Temporal do Tráfego

Neste trabalho, a classificação por análise temporal do tráfego de rede consiste em analisar a frequência de ocorrência de sessões com mesma característica, usando uma janela de 10 minutos de dados. Os procedimentos desta abordagem são apresentados na Figura 4.10.

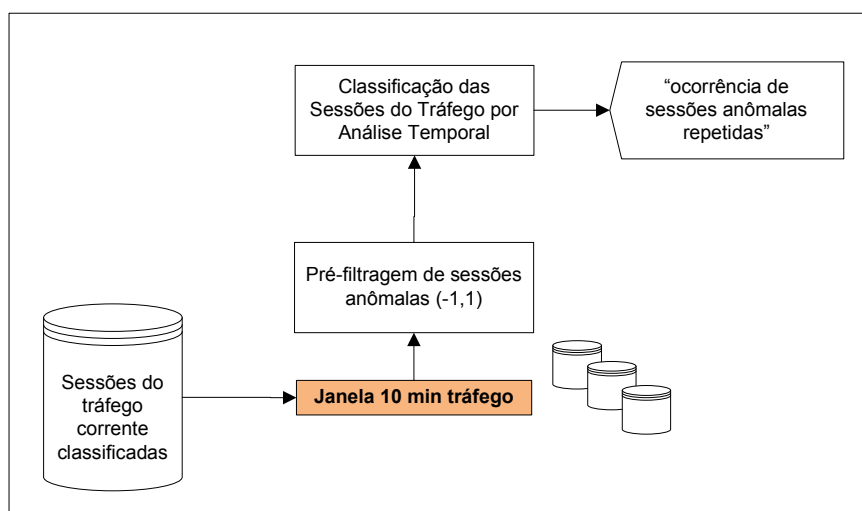


Figura 4.10 – Classificação de dados por análise temporal

A análise temporal das sessões do tráfego de rede se fez necessária para classificar anomalias no comportamento do tráfego de rede não possíveis de serem detectadas através da análise espacial. Por exemplo, ataques de DoS e Probing, em geral, disparam inúmeras sessões anômalas de características similares em ordem seqüencial durante um determinado período de tempo. A característica destas sessões anômalas pode ser encontrada esporadicamente em alguma sessão do tráfego classificada, em um determinado período de tempo, como padrão. Porém, estas sessões não podem ser repetidas com grande freqüência, dentro de um curto espaço de tempo. Como a análise espacial leva em consideração apenas os atributos de uma sessão para classificá-la, pode não detectar a anomalia e gerar alarme falso positivo.

De forma a resolver esta limitação, a classificação dos tráfegos de rede baseada em análise temporal do TRAFICIN é composta de duas partes. Na primeira parte, a base de dados histórica de dois meses de tráfego de rede caracterizada na Seção 4.5, é utilizada para treinar o classificador baseado na análise espacial apresentada na Seção anterior. Após o treinamento, novas sessões correntes foram aplicadas para teste/validação do TRAFICIN, sendo



classificadas como anômala abaixo da média (-1), padrão (0) ou anômala acima da média (1). Na segunda parte, as sessões já classificadas na análise espacial (conteúdo), passam por um sistema que utiliza uma janela de tempo de 10 minutos para análise da existência de grande ocorrência seqüencial de sessões anômalas similares, que possa caracterizar o comportamento anômalo do tráfego, não identificado na classificação anterior.

A fim de obter uma comparação de resultados em relação a taxas de falsos positivos gerados na detecção de anomalias, os mesmos métodos de classificação da análise espacial, foram aplicados na análise temporal, tais como, rede neural MLP (*MultiLayer Perceptron*), SVM (*Support Vector Machine*), RBF (*Radial Basis Functions*), Árvore de Decisão e Máxima Verossimilhança, acrescidos da implementação de um sistema baseado em regras de produção do tipo se <condição> então <ações> cuja definição pode ser encontrada em Russel e Norvig (2004).

Com a execução dos quatro processos da TRAFICIN juntamente com as técnicas e abordagens empregadas conseguiu-se detectar anomalias na rede, a partir da análise de grande volume de dados, com baixa taxa de alarmes falsos.



## 5 RESULTADOS DE ANÁLISES

A metodologia TRAFICIN foi avaliada a partir de análises de conjuntos de dados reais de tráfego HTTP da rede local de computadores (RedeBeta) no Instituto Nacional de Pesquisas Espaciais em São José dos Campos (INPE) e dados sintéticos de tráfego anômalo gerados no ambiente do Laboratório de Redes da Divisão de Desenvolvimento de Sistemas de Solo (LabRedes-DSS) do INPE.

Cada etapa desta metodologia foi construída com base em estudos, aplicação e avaliação de métodos, processos, técnicas e ferramentas estatísticas e de inteligência computacional visando a construção de um modelo de comportamento do tráfego padrão preciso e compacto.

Os principais resultados dos experimentos conduzidos são apresentados nas seções a seguir.

### 5.1. Cenário dos Testes

Para os experimentos, a coleta de dados reais do tráfego da RedeBeta foi realizada através de uma estação de trabalho funcionando como sensor de SDI, com o software *tcpdump* instalado em plataforma Linux, em uma configuração de *hardware* com CPU Intel Xeon 2.33 GHz, HD 600GB e RAM 4GB.

Como o tráfego de rede produz grandes volumes de dados, foi usada uma estação dedicada para armazenamento de dados, com as seguintes características: Intel Xeon, CPU 2,33 GHz, HD 500GB e RAM 4GB e dois HDs externos com a capacidade para armazenamento de 750GB e 1TB. Para processamento dos dados, principalmente na etapa de clusterização que envolveu 1.375 GB de dados, foram utilizados três computadores com as seguintes especificações: Intel Xeon CPU 2.4GHz, HD 500GB e RAM 2GB; Intel Core 2 CPU 2.83GHz, HD 500GB e RAM 4GB; e Notebook Sony Vaio

Intel Core 2 CPU 2.1GHz, HD 500G e RAM 3GB, cada um executando programas para tarefas de análise estatística, clusterização e classificação de dados.

O desenvolvimento de *scripts* em Linux para coleta, armazenamento e análise dos dados de tráfego de rede foi fundamental para as atividades de cópia dos *dumps* do sensor para a máquina de armazenamento, automatização da chamada do sistema Recon para o processo de reconstrução de sessões de vários conjuntos (*script* “autoRecon.sh”) e *backup* dos dados.

Para cópia de dumps foi desenvolvido o *script* “cpdumpsentredatas-v1.sh” com a finalidade de facilitar e agilizar a cópia dos arquivos de dados do tráfego de rede da máquina de captura para o HD externo e vice-versa, obedecendo o intervalo de datas fornecido.

O *script* “autoRecon.sh” foi desenvolvido para automatizar o processo de reconstrução das sessões de 144 arquivos diariamente, ou seja, automatizar o processo de inclusão dos dados capturados (“*dumps*”) na base MySQL. Os arquivos de captura do tráfego de rede são gerados a cada 10 minutos continuamente. Ao final de cada dia, são gerados 144 *dumps* que são inseridos na base MySQL através da ferramenta Recon. O script lista os arquivos de um determinado dia e executa a ferramenta Recon automaticamente.

O sistema Recon foi adaptado nos módulos som.c e recon.c para gerar arquivos do tipo texto necessários para análise estatística, contendo: data/hora da captura do pacote, total de pacotes e total de sessões referente ao período da captura determinado, a partir de dados da base de dados em MySQL criada pelo Recon.

O cenário de teste para geração de dados de tráfego anômalo envolveu o uso de duas estações clientes Linux, duas estações servidoras web (Linux e

Windows) e uma estação de captura de tráfego Linux no LabRedes-DSS. O sensor de SDI consistiu de uma estação dedicada Linux contendo o *tcpdump* e Wireshark, com a finalidade de captura de todo tráfego interno gerado na comunicação das estações do Laboratório. Através do analisador de protocolo de rede Wireshark (WIRESHARK, 2011) foi possível uma análise visual do fluxo de pacotes para entendimento do tráfego.

Através das duas estações clientes foram lançados ataques do tipo DoS (negação de serviço) e Probing (varredura de portas) contra as estações servidoras web residentes nesta rede para produzir dados de tráfego anômalo.

## **5.2. Dados Utilizados na Análise**

Os conjuntos de dados reais do tráfego HTTP da RedeBeta, coletados nos meses de março/2010 a junho/2010 e armazenados em bases de dados compostas por atributos de sessões do tráfego de rede, foram utilizados na análise. Estes dados são referenciados ao longo deste trabalho como “dados de tráfego histórico” (de 05/03/2010 a 06/05/2010) e “dados de testes” (de 06/05/2010 a 07/06/2010).

A Tabela 5.1 apresenta a descrição dos dados utilizados para análise do comportamento do tráfego da RedeBeta, através de exemplos de arquivos.

Tabela 5.1 – Modelos de arquivos utilizados na construção da TRAF CIN

Arquivo	Extensão	Conteúdo
05032010-00_00_01	.dump	Dados brutos coletados de 2 meses de tráfego (5/mar/2010 a 06/mai/2010)
S05032010	.sql	sessões 10 min de tráfego organizadas por dia do mês em tabelas MySQL
P00_07-S04052010	.m	sessões organizadas por período do dia e dia do mês (em Matlab)
mediaP13_19, curtoseP13_19, dpP13_19, varianciaP13_19, transfZP13_19_05032010	.txt	arquivos contendo valores de media, variância, desvio padrao, curtose e z-score por atributo
ImgMonP13_19_05032010-7	.jpg	gráfico do momento estatístico contendo valores de media, variância, desvio padrão e curtose por atributo
P00_07-Domingos	.m	sessões organizadas por período do dia e dia da semana (em Matlab)
Posicao_Sessoes_porGrupo-P13_19-S05032010	.txt	arquivos contendo as posicoes das sessoes nos conjuntos de dados analisados (tabelas mysql)
Sessoes_por_Grupo_P13_19-S05032010	.txt	sessões organizadas por período do dia e dia da semana
VA	.txt	arquivo contendo o total de grupos gerados
AgrupaDados-P13_19-S05032010	.jpg	gráfico de apresentação das sessões clusterizadas por grupo
MC_13_19-VA-domingos	.txt	maiores <i>clusters</i> por período do dia e dia da semana
Sessoes_MC_P00_Domingos	.txt	arquivo contendo o total de sessões por grupo
P13_trn1.txt	.txt	dados rotulados -1, 0, 1 a equilibrar: arquivos por período, com os dados Z-score e valor de atributos do tráfego histórico 2 meses
P13_trn1_c0.txt	.txt	dados rotulados -1, 0, 1 equilibrados para treinamento do classificador: arquivos por período, com valor de atributos do tráfego histórico balanceado
pesos	.mat	dados de aprendizagem do classificador: arquivos contendo as matrizes de pesos
P00_00-S03042010	.m	dados do tráfego corrente: arquivos com as sessões (com valor dos atributos) do tráfego corrente (amostra) a serem classificadas

As primeiras análises estatísticas foram através do cálculo da média, variância, desvio padrão e curtose foram aplicadas sobre os dados do tráfego histórico

organizados por dia do mês, coletados de 05/03/2010 a 06/05/2010, contendo as seguintes características:

- N° de arquivos = 60;
- Tamanho total = 949 MB ;
- Tamanho médio de cada arquivo = 6,4 MB;
- N° médio de pacotes capturados por hora: 00h as 01h = 30.715 pacotes por hora; 14h as 15h = 696.448 pacotes por hora;
- N° médio de pacotes capturados por dia: 17.871.823 por dia;
- N° médio de sessões reconstruídas por hora: 00h as 01h = 542 sessões por hora; 14h as 15h = 28.152 sessões por hora;
- N° médio de sessões reconstruídas por dia: 217.878 por dia.

A Tabela 5.2 apresenta um modelo da Tabela MySQL “S05042010” com dados brutos coletados no dia 05/04/2010 organizados por dia do mês e período do dia.

Tabela 5.2 - Amostra de dados coletados na rede de produção

ID	Hora	psizeCL	psizeSV	pnumCL	pnumSV	smalpkt	dataDir	brecvCL	brecvSV	duration
42275	10_00	0	0	1	1	1	0	0	0	0,193788
42276	10_00	78,5	63,43	4	7	0,82	-3	314	444	2,59089
42277	10_00	0	0	0	1	1	-1	0	0	0
42278	10_00	78,5	56	4	6	0,8	-2	314	336	2,58257
42279	10_00	78,5	56	4	6	0,8	-2	314	336	0,915566
42280	10_00	0	0	1	1	1	0	0	0	0,192344
42281	10_00	62,8	63,43	5	7	0,83	-2	314	444	2,58633
42282	10_00	78,5	56	4	6	0,8	-2	314	336	1,55623
42283	10_00	0	70,25	5	8	0,85	-3	0	562	4,34058
42284	10_00	0	0	0	1	1	-1	0	0	0
42285	10_00	62,6	63,29	5	7	0,83	-2	313	443	4,02176
42286	10_00	0	0	2	1	1	1	0	0	0,196725
42287	10_00	115,5	74	4	7	0,73	-3	462	518	2,59601
42288	10_00	113,25	56	4	6	0,8	-2	453	336	2,58935
42289	10_00	78,25	56	4	6	0,8	-2	313	336	2,59158
42290	10_00	0	0	1	1	1	0	0	0	0,195752
42291	10_00	150,67	55,83	3	6	0,78	-3	452	335	2,58973
42292	10_00	0	0	1	1	1	0	0	0	0,195477
42293	10_00	78,5	55,83	4	6	0,8	-2	314	335	2,58439
42294	10_00	113,25	56	4	6	0,8	-2	453	336	2,66828
42295	10_00	138,25	56	4	6	0,8	-2	553	336	2,64439
42296	10_00	104,33	55,83	3	6	0,78	-3	313	335	2,58112
42297	10_00	0	0	1	1	1	0	0	0	0,193738
42298	10_00	78,5	55,83	4	6	0,8	-2	314	335	2,59056
42299	10_00	0	0	2	1	1	1	0	0	0,196051

As sessões do tráfego de rede foram reconstruídas a partir dos dados brutos coletados de 2 meses de tráfego (5/mar/2010 a 06/mai/2010). Os arquivos *dumps* (arquivos com dados brutos do tráfego de rede) de janelas de 10 minutos do tráfego foram armazenados em tabelas MySQL contendo sessões organizadas por dia do mês.

Os dados das tabelas MySQL foram processados pelo *script* “convertetxt.c” O convertetxt.c também tem a característica de agrupar os dados por períodos P1 (madrugada – 00h às 06:50h), P2 (manhã – 07h às 12:30h), P3 (tarde – 13h às 18:50h), P4 (noite – 19h às 23:50h) e por dia capturado, permitindo assim um posterior estudo do comportamento do tráfego ao longo do tempo.



As próximas análises utilizando a técnica de clusterização foram aplicadas sobre os dados do tráfego histórico organizados por dia do mês e período do dia, coletados de 05/03/2010 a 06/05/2010 em arquivos (por exemplo: “P10\_00-S05042010”), contendo as seguintes características:

- N° de arquivos = 251
- Tamanho total = 949 MB
- Tamanho médio de cada arquivo = 1,5 MB

Em busca de melhores resultados na clusterização dos dados, foi desenvolvida uma rotina de clusterização para reorganizar os conjuntos de dados por período do dia e dia da semana. Os conjuntos de dados resultantes nesta etapa foram armazenados em arquivos (por exemplo: P13\_19-Segundas) e apresentam as seguintes características:

- No. de arquivos = 28
- Tamanho total = 949 MB
- Tamanho médio de cada arquivo = 30,3 MB

Uma amostra de tráfego anômalo gerado por um ataque do tipo DoS lançado no ambiente do LabRedes-DSS é apresentada na Tabela 5.3.

Tabela 5.3 – Amostra de tráfego anômalo gerado por um ataque do tipo DoS

ID	Hora	psizeCL	psizeSV	pnumCL	pnumSV	smallpkt	dataDir	brecvCL	brecvSV	duration
6	00_30	0	6	8	15	1	-7	0	90	1,00098
7	00_40	0	1,2	4	5	0,89	-1	0	6	1,01427
8	00_50	0	7,5	5	8	0,92	-3	0	60	1,00114
9	00_50	0	7,5	5	8	0,92	-3	0	60	1,00086
10	00_50	0	7,5	5	8	0,92	-3	0	60	1,0007
11	01_00	0	7,5	5	8	0,92	-3	0	60	1,00082
12	01_00	0	7,5	5	8	0,92	-3	0	60	1,00073
13	01_00	0	7,5	5	8	0,92	-3	0	60	1,0007
14	01_00	0	7,5	5	8	0,92	-3	0	60	1,0007
15	01_00	0	7,5	5	8	0,92	-3	0	60	1,00079
16	01_00	0	7,5	5	8	0,92	-3	0	60	1,0007
17	01_00	0	7,5	5	8	0,92	-3	0	60	1,00075
18	01_00	0	7,5	5	8	0,92	-3	0	60	1,00068
19	01_00	0	7,5	5	8	0,92	-3	0	60	1,00066
20	01_00	0	7,5	5	8	0,92	-3	0	60	1,00067
21	01_00	0	8,57	5	7	0,92	-2	0	60	1,00067
22	01_00	0	7,5	5	8	0,92	-3	0	60	1,00086
23	01_00	0	7,5	5	8	0,92	-3	0	60	1,00068
24	01_00	0	7,5	5	8	0,92	-3	0	60	1,00069
25	01_00	0	7,5	5	8	0,92	-3	0	60	1,00076
26	01_00	0	7,5	5	8	0,92	-3	0	60	1,00068
27	01_00	0	7,5	5	8	0,92	-3	0	60	1,00073
28	01_00	0	7,5	5	8	0,92	-3	0	60	1,00071
29	01_00	0	7,5	5	8	0,92	-3	0	60	1,00068
30	01_00	0	7,5	5	8	0,92	-3	0	60	1,0007
31	01_00	0	7,5	5	8	0,92	-3	0	60	1,00074

### 5.3. Análise Estatística Preliminar

Nesta etapa, foram realizados dois tipos de análise: análise baseada em técnicas estatísticas e análise baseada em técnicas de análise de séries temporais para entendimento do comportamento do tráfego.

#### 5.3.1. Análise baseada em técnicas estatísticas

Os primeiros estudos dos conjuntos de dados históricos do tráfego de rede foram conduzidos por meio da aplicação de técnicas estatísticas sobre os dados. Para análise inicial do grande volume de dados em busca de conhecimento relevante, vários cálculos estatísticos, tais como média, desvio

padrão e curtose foram visualizados em gráficos, aplicados aos dados organizados por hora, dia, semana, 1 mês e dois meses.

Outra análise estatística compreendeu o cálculo do *z-score* para cada atributo das sessões analisadas, nove no total, plotados de diferentes formas: juntos, separados, sobre os dados de 1 mês e 2 meses de coleta de tráfego.

### 5.3.2. Análise estatística para um dia de tráfego

Nesta Seção, são apresentados os gráficos utilizados para análise visual do conjunto de dados de tráfego de rede gerado em um dia. A Figura 5.1 ilustra a apresentação dos atributos das sessões do tráfego de um dia em coordenadas paralelas.

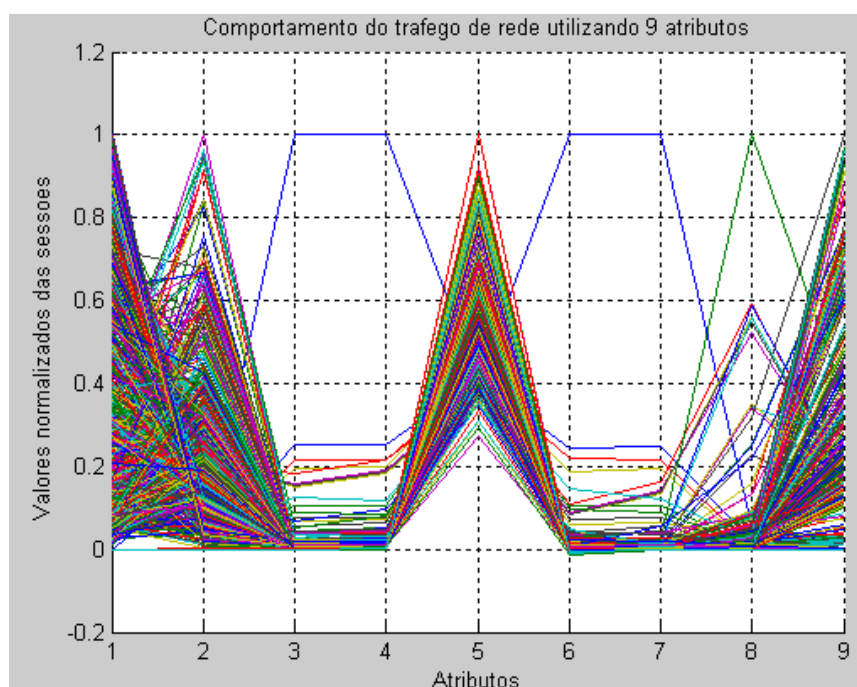


Figura 5.1 – Nove atributos das sessões do tráfego de rede em coordenadas paralelas

Os nove atributos foram impressos em diferentes coordenadas no gráfico denominadas “Coordenadas Paralelas”. Através das coordenadas paralelas é

possível o mapeamento de valores com unidades diferentes de forma proporcional.

Isto permite construir uma representação global dos dados, permitindo a visualização comparativa entre os padrões de comportamento dos atributos medidos.

Os pontos de atributos selecionados são interligados através de linhas e representam o comportamento de uma sessão do tráfego de rede. Ao conjunto de sessões impressas no gráfico, pode-se observar o comportamento do tráfego como um todo (MÜLLER, 2002).

Conforme ilustrado na Figura 5.1, pode-se observar que alguns atributos como o psizeCL e psizeSV possuem comportamento variável ao longo de um dia, sendo impossível analisar anomalias através destes atributos graficamente. Já os atributos pnumCL, pnumSV é possível identificar anomalias graficamente. Se apenas estas variáveis fossem importantes para compreender o comportamento do tráfego padrão, resolveria o problema, mas às vezes não é.

Nas Figuras 5.2 a 5.10 são apresentados os valores dos atributos individuais das sessões do tráfego de um dia.

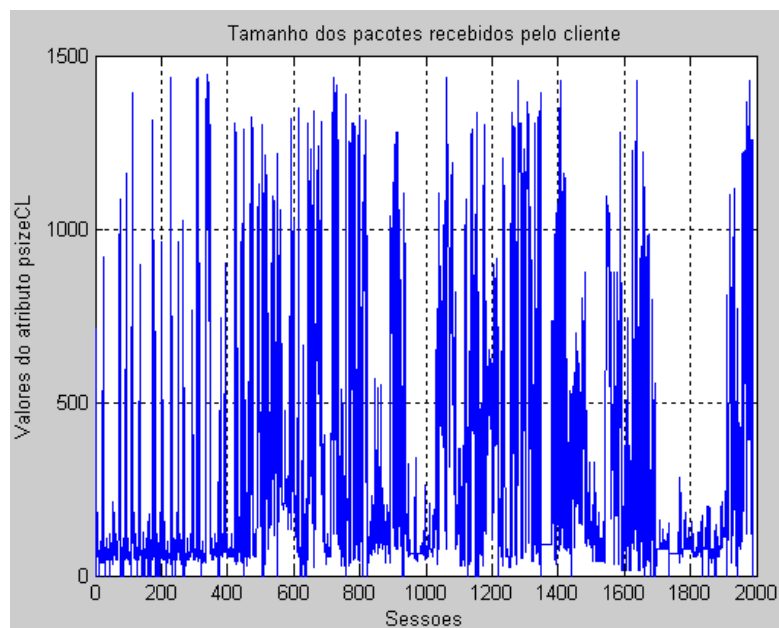


Figura 5.2 - Atributo psizeCL nas sessões do tráfego de um dia

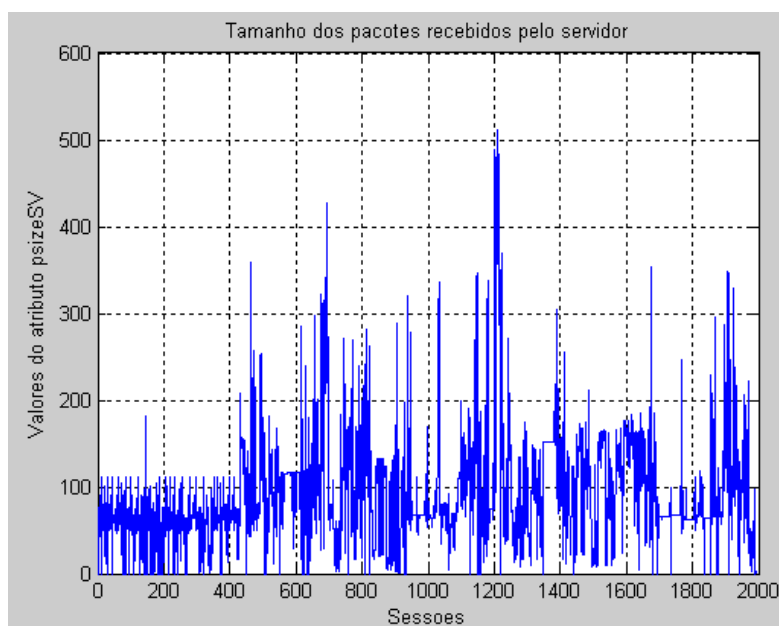


Figura 5.3 - Atributos psizeSV nas sessões do tráfego de um dia

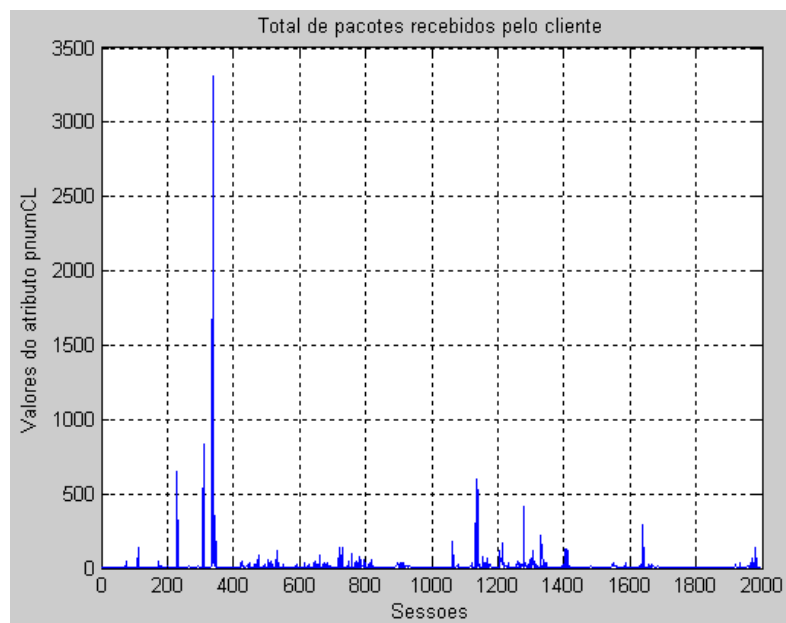


Figura 5.4 - Atributo pnumCL das sessões do Tráfego de um dia

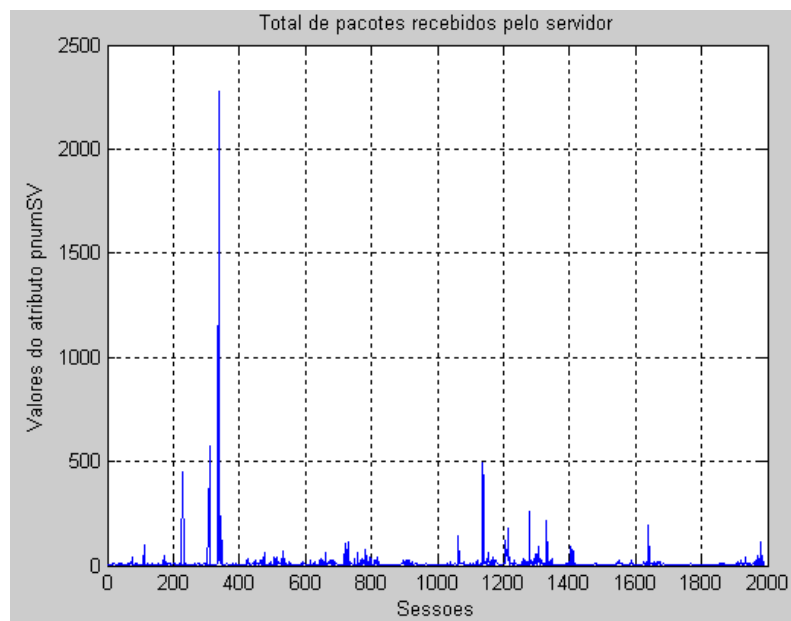


Figura 5.5 - Atributo pnumSV das sessões do Tráfego de um dia

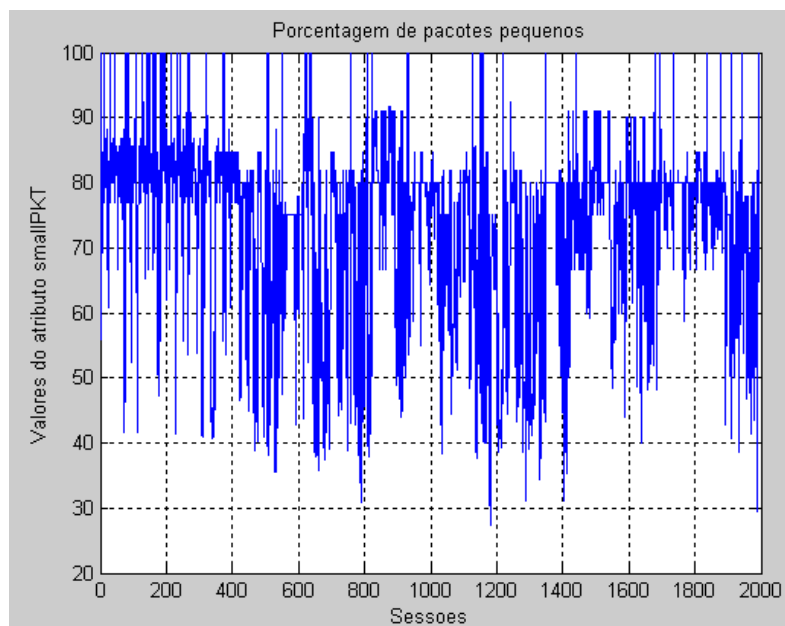


Figura 5.6 - Atributo smallPKT das sessões do tráfego de um dia

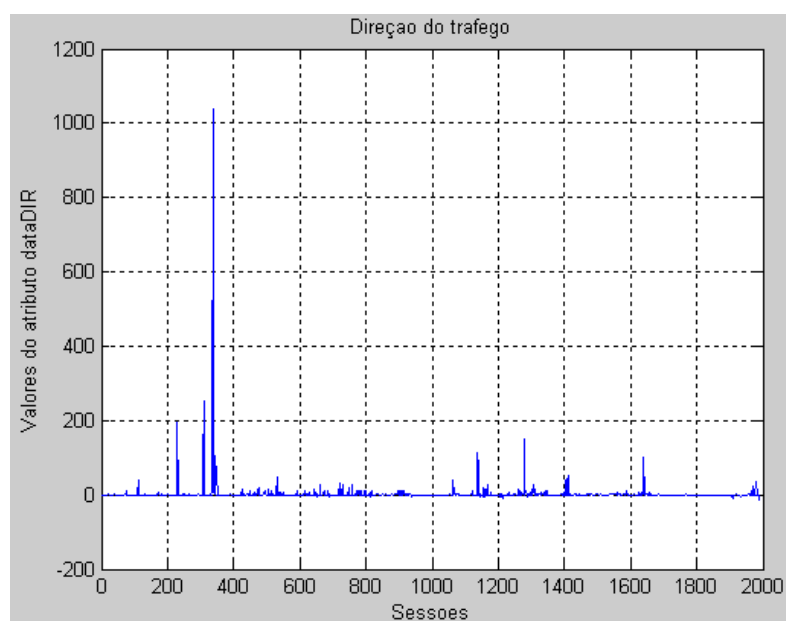


Figura 5.7 - Atributo dataDir das sessões do tráfego de um dia

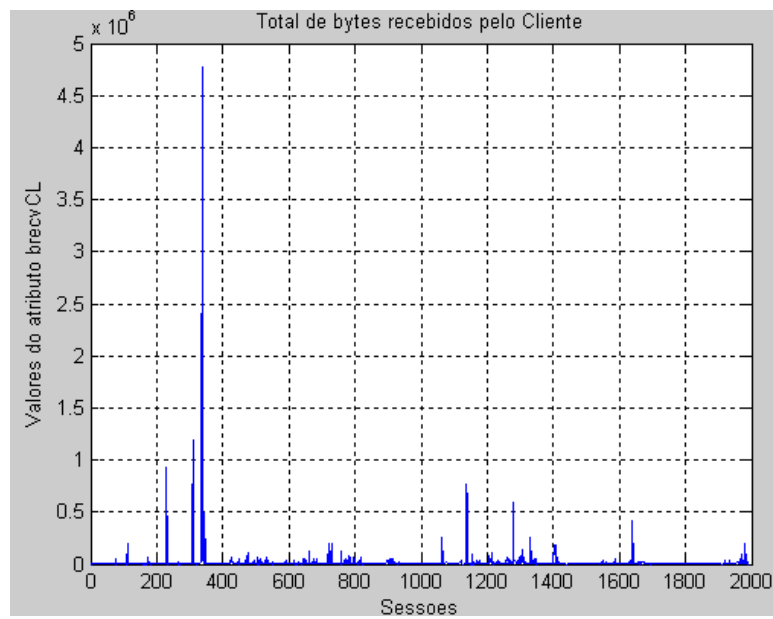


Figura 5.8 - Atributo `brevcCL` das sessões do tráfego de um dia

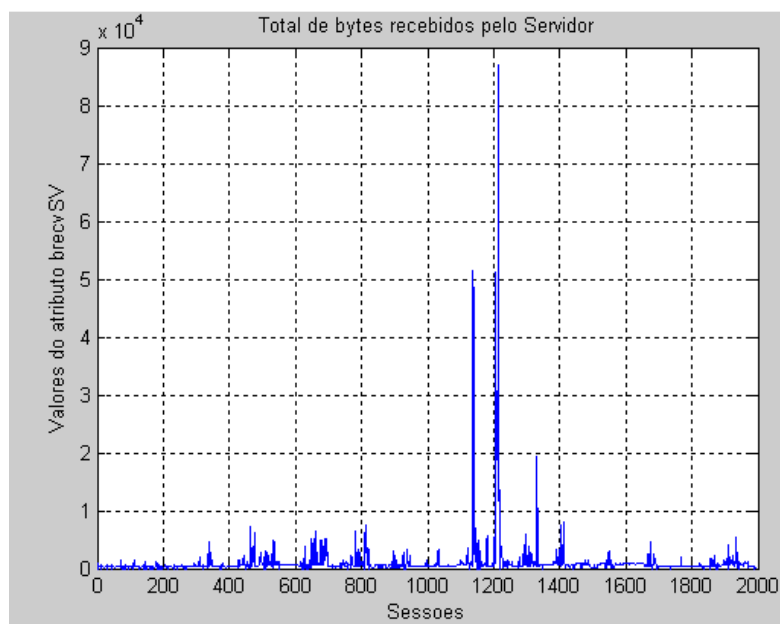


Figura 5.9 – Atributo `brevcSV` das sessões do tráfego de um dia



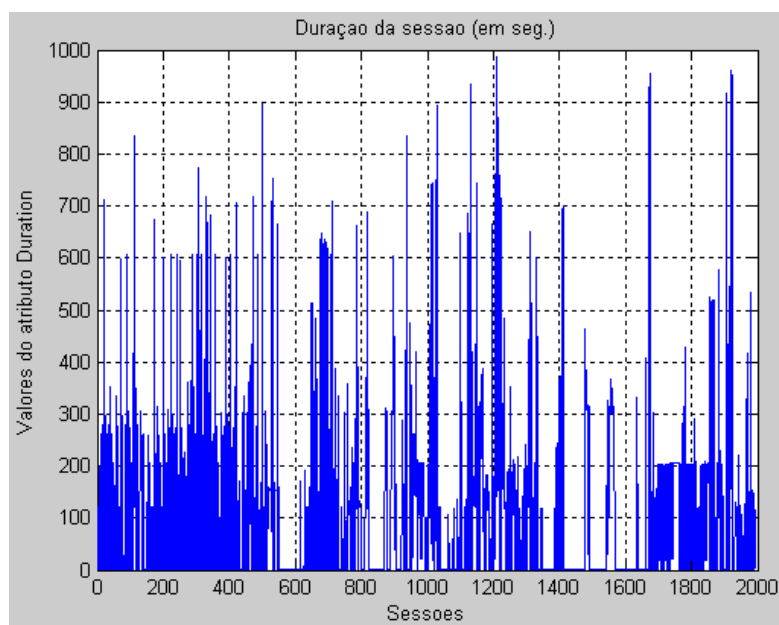


Figura 5.10 – Atributo duration das sessões do tráfego de um dia

Para observar o comportamento de cada atributo do conjunto analisado em um dia de tráfego, foram calculados a média, desvio padrão e curtose de cada atributo.

Através dos cálculos estatísticos dos atributos das sessões do tráfego de rede, apresentados na Figura 5.11, foram analisados o comportamento dos 9 atributos referentes a um dia de tráfego de rede.

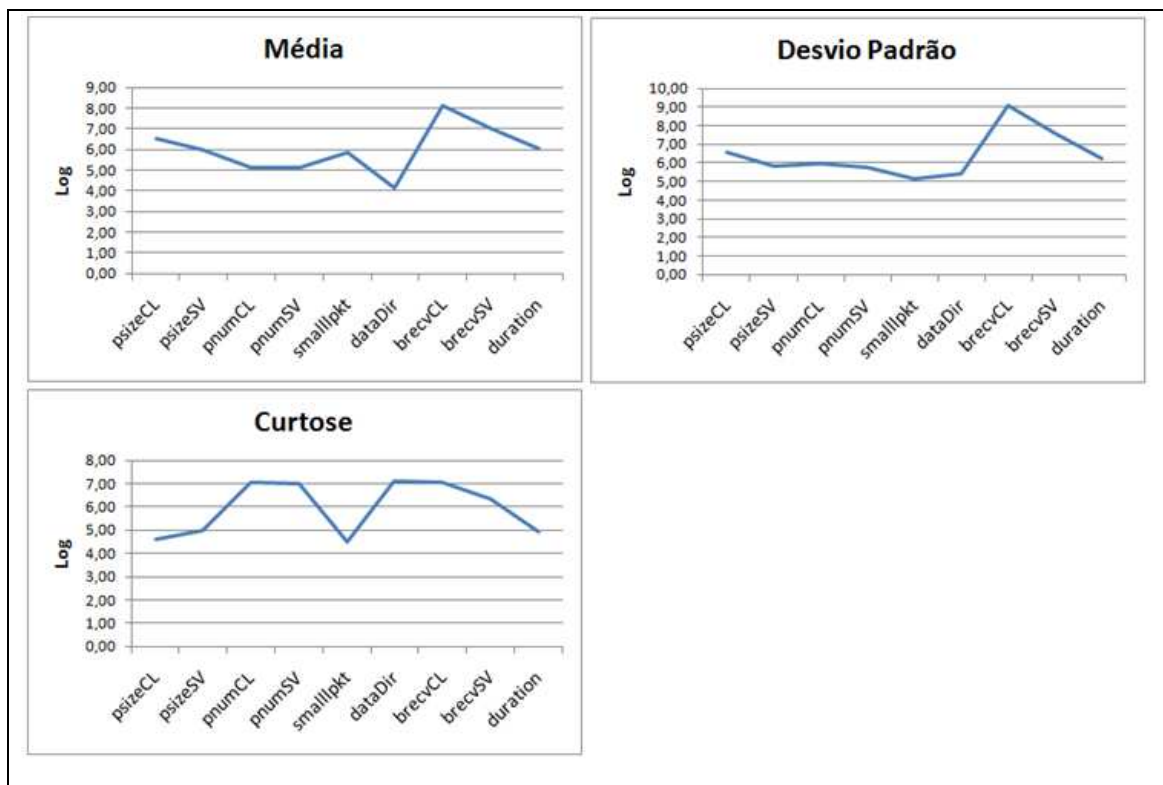


Figura 5.11 – Comportamento dos 9 atributos em um dia de tráfego

Conforme apresentado na Figura 5.11, o atributo brevcCL (total de *bytes* recebidos pelo cliente) apresenta valores médios muito acima dos demais valores médios de atributos.

Com os cálculos estatísticos da média, desvio padrão e curtose no tráfego de rede, foi possível constatar a diversidade estatística dos diferentes atributos das sessões analisadas. É interessante utilizar essa diversidade para fins de classificação de cada padrão de variabilidade temporal observado nas séries estudadas. É importante destacar que a natureza desta diversidade estatística pode estar relacionada à diferentes valores de auto-correlação e expoentes de escala associado a cada tipo de atributo.

Cada ponto no gráfico representa o comportamento de um determinado atributo em um dia de tráfego de rede.

### 5.3.3. Análise estatística para um mês de tráfego

Para observar o comportamento de cada atributo do conjunto analisado em um mês de dados de tráfego organizados por período do dia, também foram calculados a média, o desvio padrão e a curtose de cada atributo nos quatro períodos de tempo: manhã, tarde, noite e madrugada. Os resultados obtidos no período da manhã (das 07h as 13h) são apresentados na Figura 5.12.

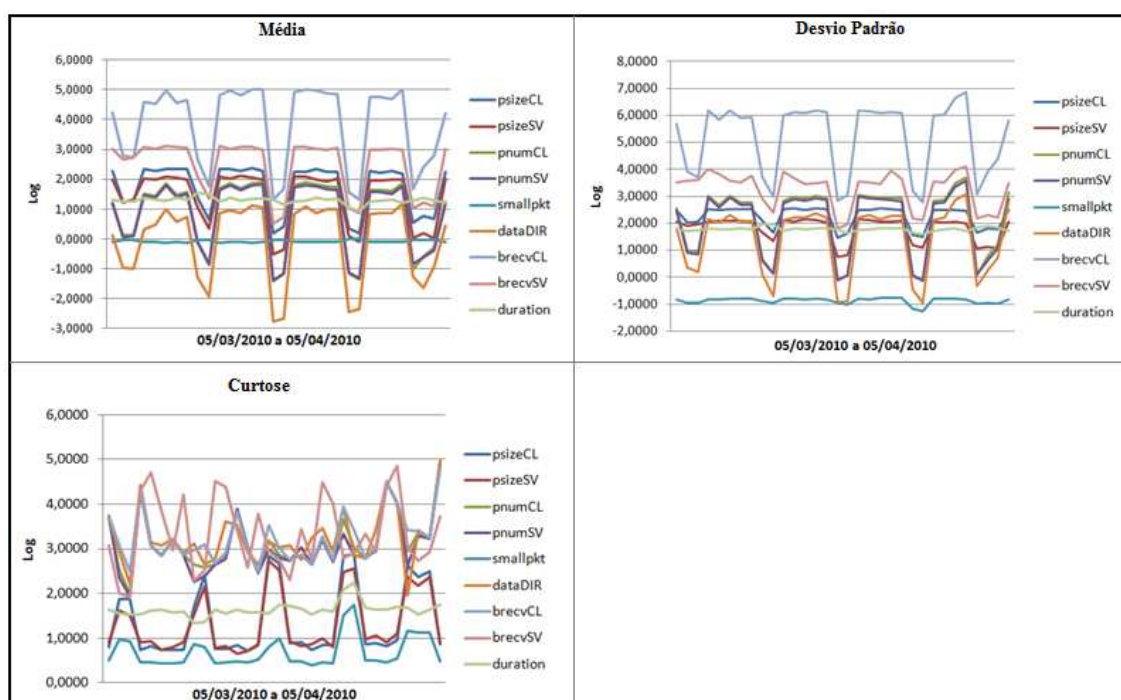


Figura 5.12 – Comportamento dos 9 atributos em um mês de tráfego

Conforme apresentado na Figura 5.12, ao analisar um mês de tráfego de rede foi possível observar a diversidade estatística nos valores de um mesmo atributo, em dias de semana e períodos do dia. Ainda pode-se observar uma diferença no comportamento do tráfego nos finais de semana, que foi demonstrada por valores menores de atributos decorrentes do menor número de tráfego na rede.

Fazendo uma análise dos métodos estatísticos acima referidos, pode-se concluir que foram influenciados por valores de atributos ou muito grandes ou muito pequenos. Estas medidas só dão informação útil sobre a localização do centro da distribuição dos dados e sobre a variabilidade se as distribuições dos dados forem aproximadamente simétricas.

#### 5.3.4. Análise baseada em séries temporais

A análise da série temporal dos registros diários de sessões do tráfego de rede selecionados permite observar a tendência da série.

Para analisar as séries temporais foi utilizado as técnicas de PDF e DFA com o objetivo de analisar se as flutuações dos valores dos atributos das sessões são ou não Gaussianas e as diferentes escalas de auto-correlação existentes entre as amostras.

#### 5.3.5. Análise baseada em PDF

As Figuras 5.13 a 5.17 apresentam os resultados obtidos através do cálculo da função de distribuição normal para cada atributo do conjunto de dados de um dia de tráfego.

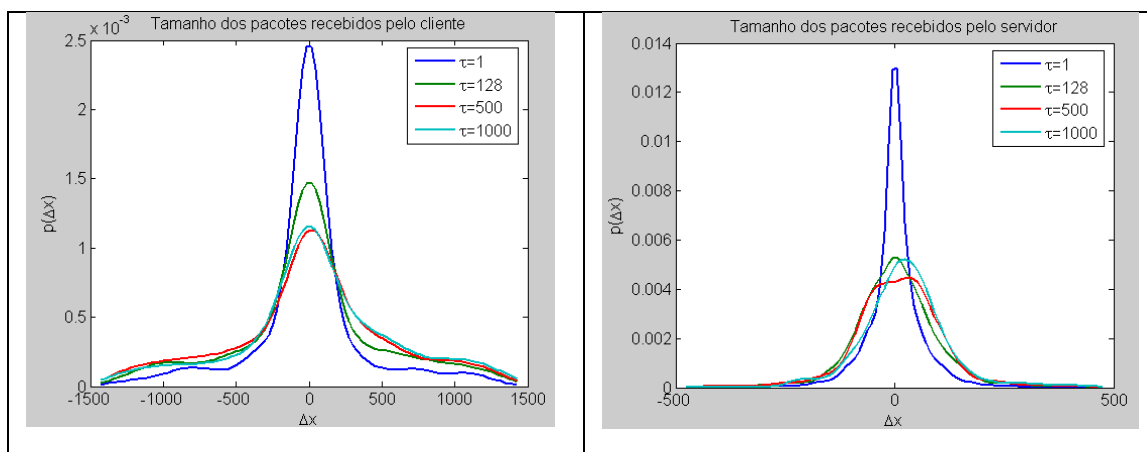


Figura 5.13 – Cálculo do PDF para os atributos psizeCL e psizeSV

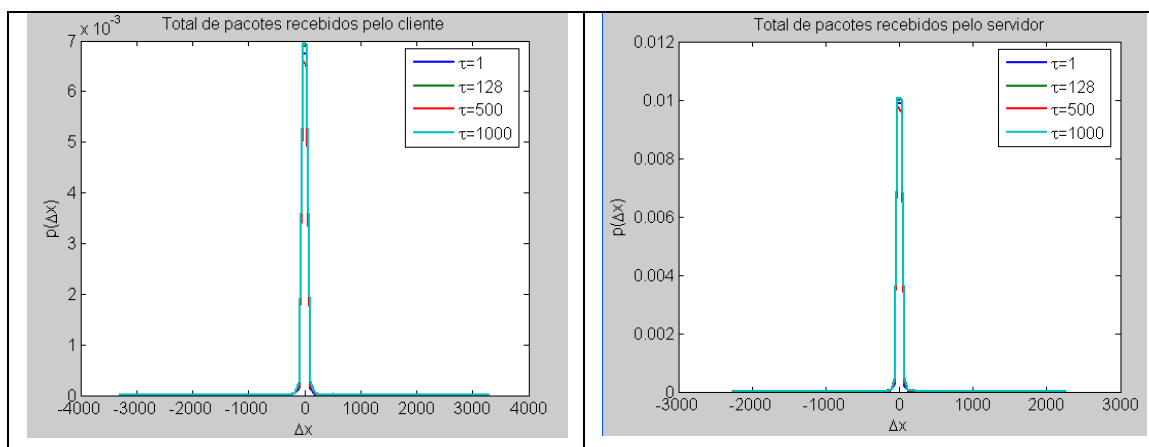


Figura 5.14 – Cálculo do PDF para os atributos pnunCL e pnunSV

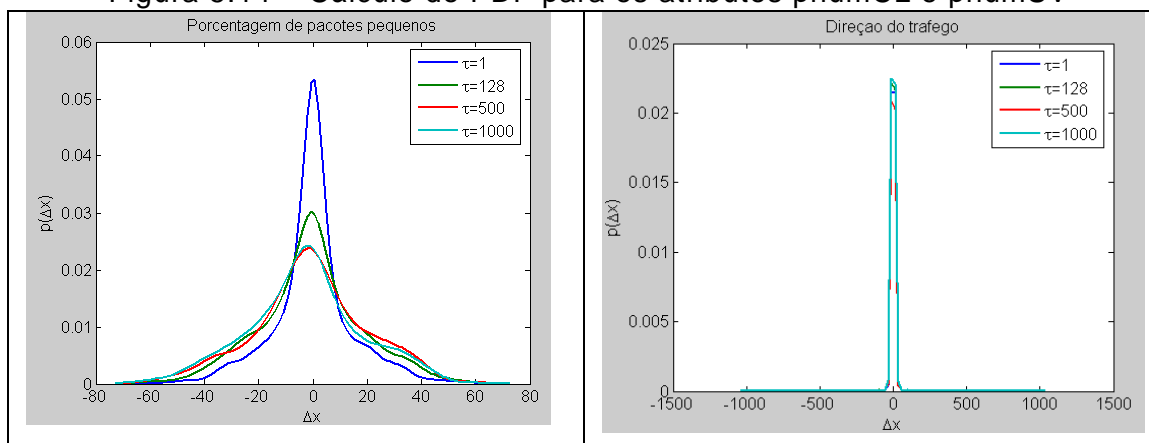


Figura 5.15 – Cálculo do PDF para os atributos smallPKT e dataDir

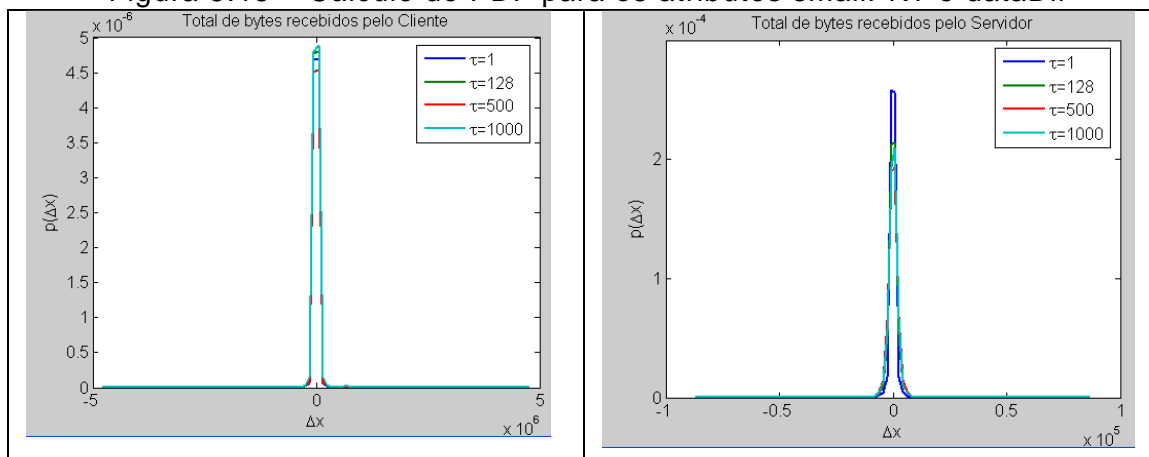


Figura 5.16 – Cálculo do PDF para os atributos brevcCL e brevcSV

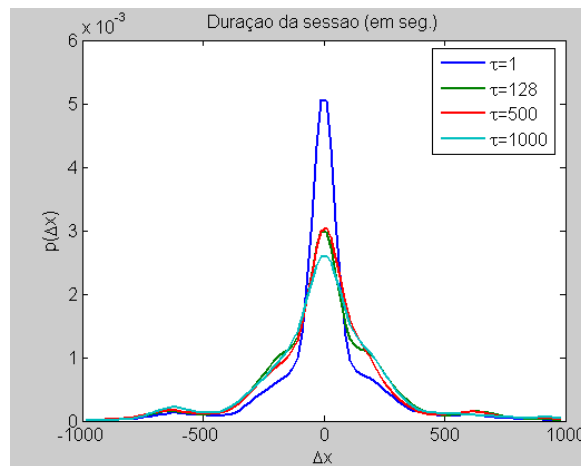


Figura 5.17 – Cálculo do PDF para o atributo duration

Pelos gráficos do PDF gerados dos valores dos atributos das sessões do tráfego pode-se observar que a flutuação é gaussiana. Os atributos pnumCL, pnumSV, dataDir, brecvCL e brecvSV apresentam menor amplitude, ou seja, menor espalhamento em relação a média. Por outro lado, os atributos psizeCL, psizeSV, smallpkt e duration possuem espalhamento maior, com maior variação no desvio em relação a média.

Outro ponto observado foi o número de amostras analisadas, sendo selecionadas em diferentes quantidades para análise de um mesmo atributo: 1, 128, 500 e 1000 amostras. De acordo com os gráficos gerados pelo PDF, quanto maior o número de amostras analisadas, mais o comportamento dos atributos tendem a convergir para uma única curva gaussiana.

### 5.3.6. Análise baseada em DFA

O expoente  $\alpha$ , obtido através do DFA, permite avaliar em que medida a tendência observada na série temporal passada implica em manutenção do comportamento no futuro, indicando um efeito de memória de longa duração na série.

A interpretação do expoente  $\alpha$  deve considerar três situações: na primeira, o expoente igual a 0,50 revela que a flutuação não influencia os componentes de tendência, ou seja, não se pode relacionar um padrão de flutuação passado com o padrão de flutuação de uma série futura; as outras duas situações (o expoente  $\alpha$  menor que 0,50 e expoente  $\alpha$  maior que 0,50) indicam que o padrão de flutuação é auto-correlacionado e influencia na tendência da série futura, sendo persistente se a tendência é aumentada e anti-persistente se a tendência é diminuída (tendência negativa).

A Tabela 5.4 apresenta os resultados obtidos através do cálculo da correlação de longo alcance via DFA das séries temporais (sessões do tráfego de rede) analisadas.

Tabela 5.4 – Comportamento do expoente  $\alpha$  nos 9 atributos do tráfego de rede

Atributos	psizeCL	psizeSV	pnumCL	pnumSV	smallpkt	dataDir	brecvCL	brecvSV	duration
<b>Expoente <math>\alpha</math></b>	0.82	0.85	0.71	0.71	0.84	0.73	0.71	0.80	0.65

Pelos valores do expoente de flutuação DFA ( $\alpha$ ) mostrados na tabela 5.4, notou-se que existe persistência das amostras analisadas dos nove atributos das sessões do tráfego de rede, ou seja, os elementos da observação são dependentes ( $\alpha > 0,5$ ). Isso significa que a flutuação dos atributos influencia os componentes da tendência, sendo possível relacionar o comportamento de um padrão de flutuação passado com o padrão de flutuação de uma série futura, permitindo previsão de séries temporais.

### 5.3.7. Análise baseada em DFA e Curtose para Caracterização do Comportamento padrão do Tráfego

Uma primeira tentativa de caracterização do comportamento padrão do tráfego de rede envolveu a análise dos atributos através da aplicação das técnicas DFA e Curtose, cujos resultados foram apresentados em (SANTOS et al., 2009; SANTOS et al., 2010c).

O valor de curtose foi calculado para analisar o achatamento da curva de distribuição de freqüência da série estudada e o expoente de flutuação DFA ( $\alpha$ ) foi utilizado para verificar a correlação das amostras.

Uma amostra das séries observada é ilustrada nas Figuras 5.18 a 5.21, apresentando a variação da DFA ( $\alpha$ ) e Curtose ( $K$ ) dos valores calculados para os atributos pnumCL, pnumSV, brecvCL e brecvSV (36 séries por atributo).

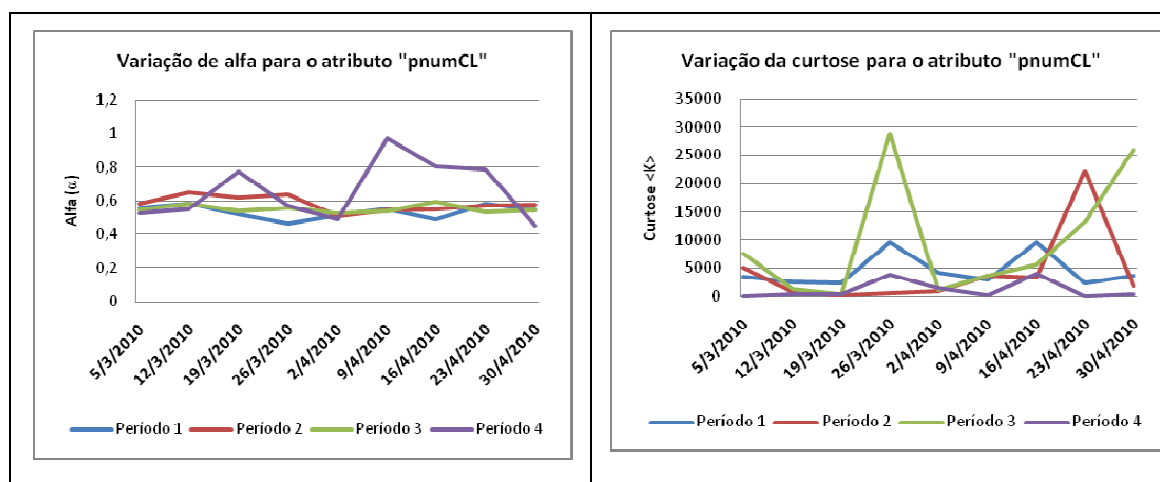


Figura 5.18 - Representação gráfica do alfa e da curtose para o atributo "pnumCL"



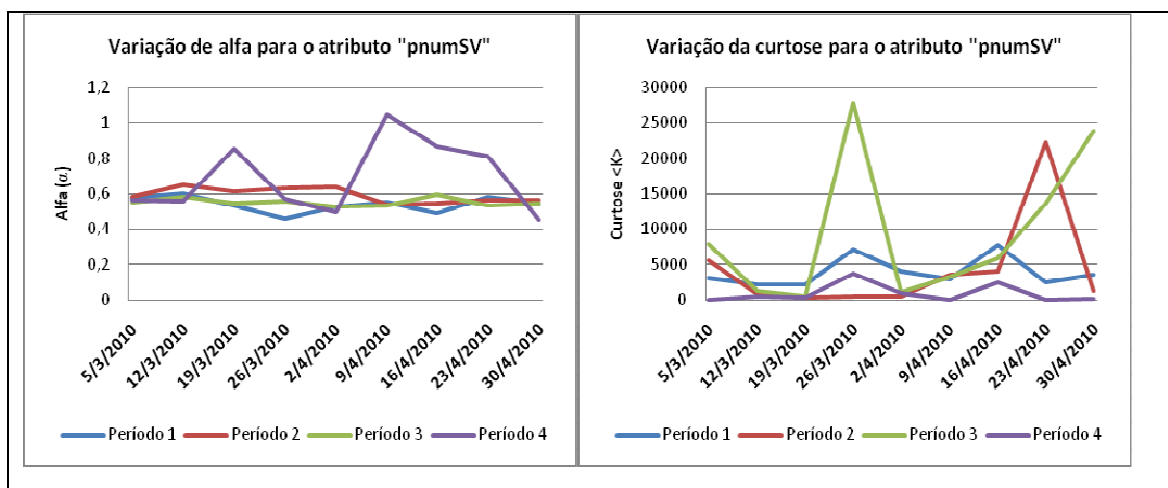


Figura 5.19 - Representação gráfica do alfa e curtose para o atributo "pnunSV"

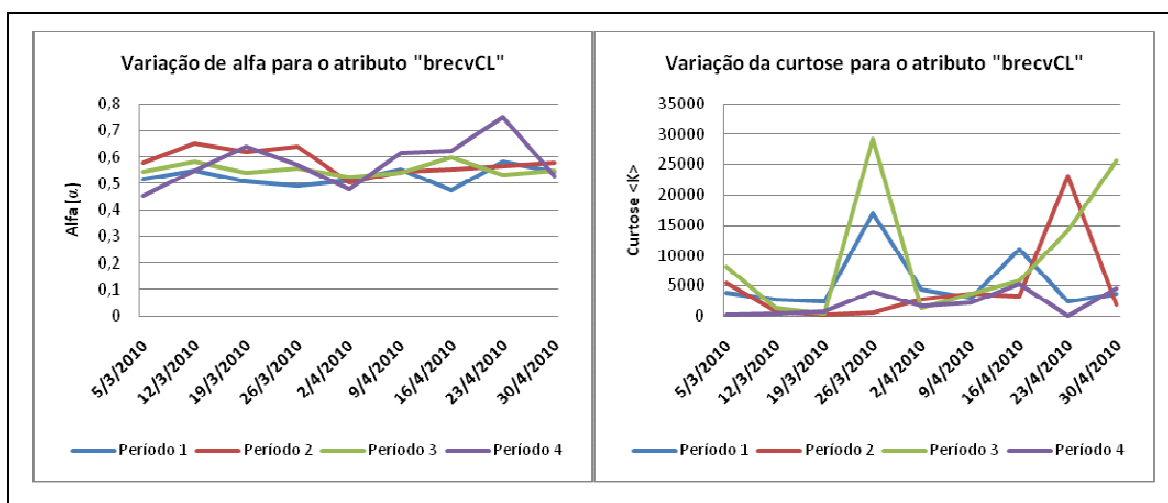


Figura 5.20 - Representação gráfica do alpha and curtose para o atributo "brecvCL"

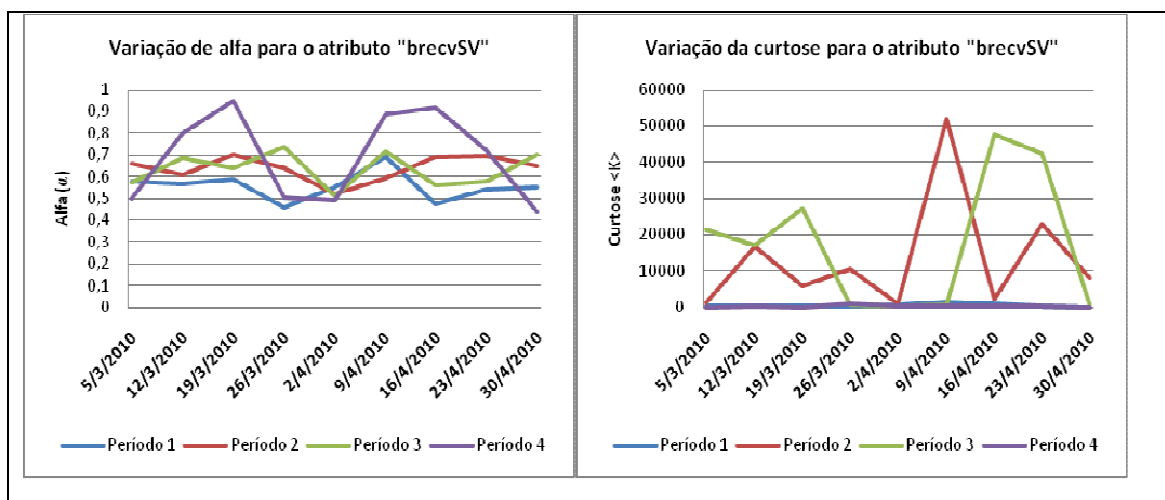


Figura 5.21 - Representação gráfica do alpha e curtose para o atributo "brecvSV"

Através dos valores de curtose obtidos para cada série ( $K > 3$ ), observou-se que a curva de distribuição de frequência para a série é semelhante a uma curva gaussiana, com apenas um pico e é mais afilado que o normal (curva leptocúrtica). Como esta curva está próxima da curva de Gauss, o conjunto de dados analisado tende a ser homogêneos, indicando que existem sessões com características semelhantes.

Pelos valores do expoente de flutuação DFA ( $\alpha$ ), notou-se que as amostras de nove atributos são persistentes ( $\alpha > 0,5$ ), possibilitando a análise e caracterização do comportamento das sessões do tráfego de rede no tempo.

A partir das imagens apresentadas, pode-se notar que, quando se compara um novo valor de curtose dos atributos pnumCL, pnumSV, brecvCL e brecvSV com aquela frequência mapeada para o padrão de tráfego, se este valor de curtose ( $k$ ) está dentro do intervalo de  $0 < k < 8000$ , significa que o atributo pertence a uma sessão padrão. Da mesma forma, se o valor da DFA está dentro da faixa de  $0,4 < \alpha < 0,7$ , o atributo observado também pertence a uma sessão padrão.

#### **5.4. Resultados da clusterização**

O grande volume e a complexidade dos dados históricos do tráfego de rede analisado foram fatores decisivos para a aplicação da abordagem de mineração de dados por clusterização, através do algoritmo “Mapa de Kohonen Adaptável” (MKA), para extrair as similaridades dos dados nos conjuntos organizados por dia da semana e período do dia.

Diferente do método de tentativa de caracterização do tráfego através de análise individual dos atributos, a técnica de clusterização possibilitou analisar o comportamento das sessões do tráfego a partir do processamento dos nove atributos simultaneamente.

Visto que anomalias podem deixar traços em um ou mais atributos diferentes da sessão, é pertinente que estes sejam analisados em conjunto para detectar anomalias com maior precisão e melhor desempenho (SANTOS et al., 2010a; SANTOS et al., 2010b).

Para os experimentos de clusterização realizados neste trabalho foi utilizado o conjunto de dados de 2 meses de tráfego. A Tabela 5.5 apresenta uma amostra escolhida aleatoriamente dos dados analisados, agrupados nos 4 períodos do dia.

Tabela 5.5. Amostra das Séries Temporais Analisadas

Série Temporal	Data/Dia da Semana	P1 (madrugada)	P2 (manhã)	P3 (tarde)	P4 (noite)
		Total de Sessões	Total de Sessões	Total de Sessões	Total de Sessões
S0903201	09/03/2010 - Ter	4477	90813	85204	18435
S1503201	15/03/2010 - Seg	4033	96721	177265	5249
S2803201	28/03/2010 - Dom	20043	18852	17682	12328
S0104201	01/04/2010 - Qui	5541	89820	119535	2979
S1404201	14/04/2010 - Qua	19130	76061	103961	7591
S1604201	16/04/2010 - Sex	21420	68565	98741	18993
S1704201	17/04/2010 - Sab	20085	86932	76890	4237

Como se observa na Tabela 5.5, o número médio de sessões ocorridas nos períodos P1, P2, P3 e P4 são respectivamente: 114.814, 527.764, 679.278 e 69.812.

Dos dados disponíveis, escolheu-se aleatoriamente a amostra do dia 09 de março para apresentar os resultados da clusterização de modo a observar a quantidade de sessões agrupadas por *cluster*. O desvio (*ds*) de 10% e similaridade (*sim*) de 70% foram utilizados nos dados de dois meses de tráfego, dos quais o processo de clusterização do dia 09/03/2010 (terça-feira), é apresentado nas Figuras 5.22 a 5.25.

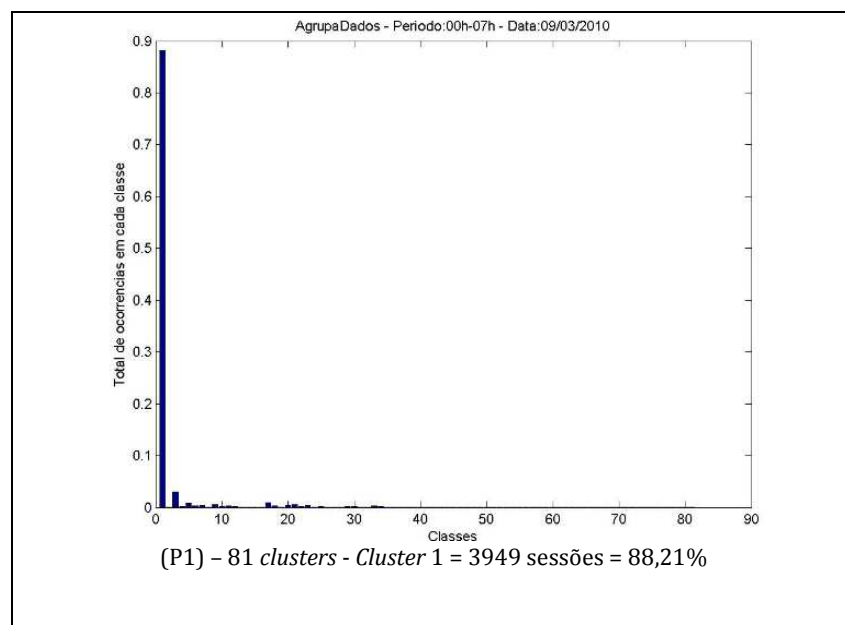


Figura 5.22 - Quantidade de sessões por *cluster* – Período das 0h às 07h

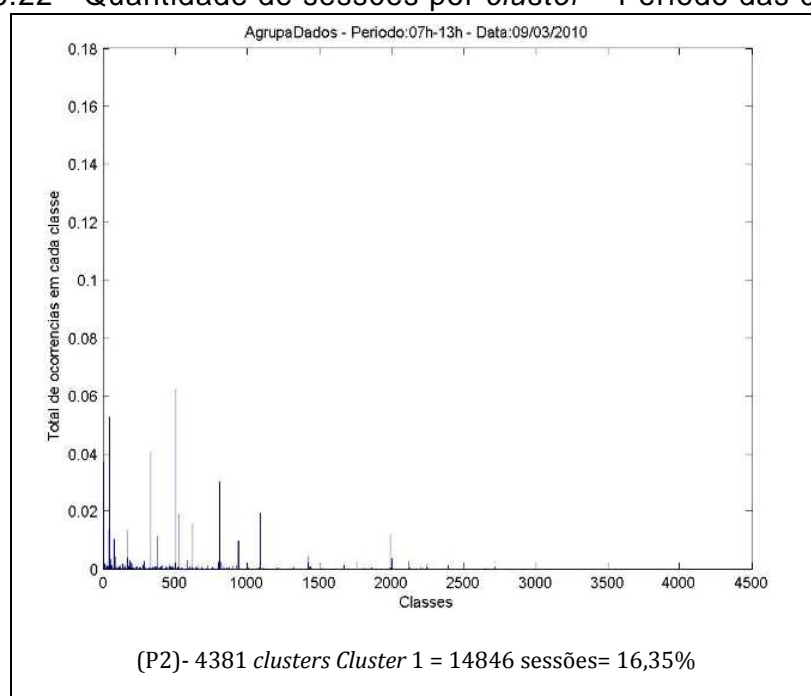


Figura 5.23 - Quantidade de sessões por *cluster* – Período das 07h às 13h

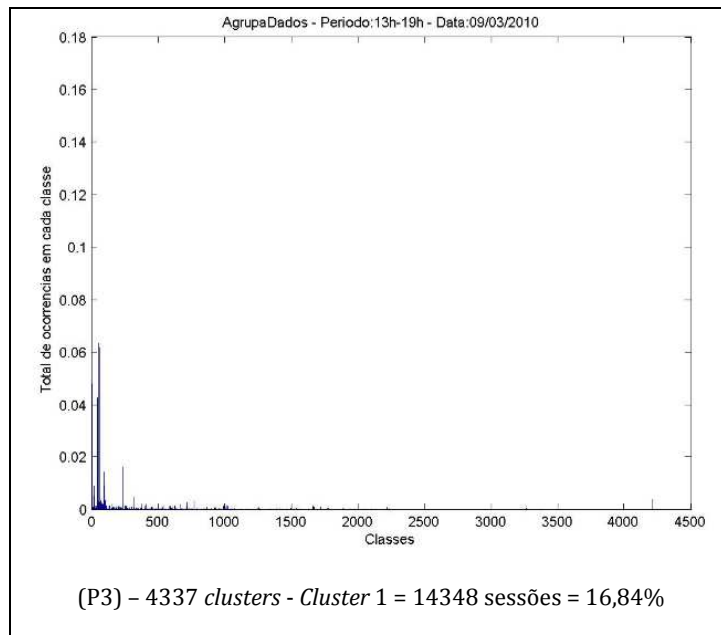


Figura 5.24 - Quantidade de sessões por *cluster* - Período das 13h às 19h

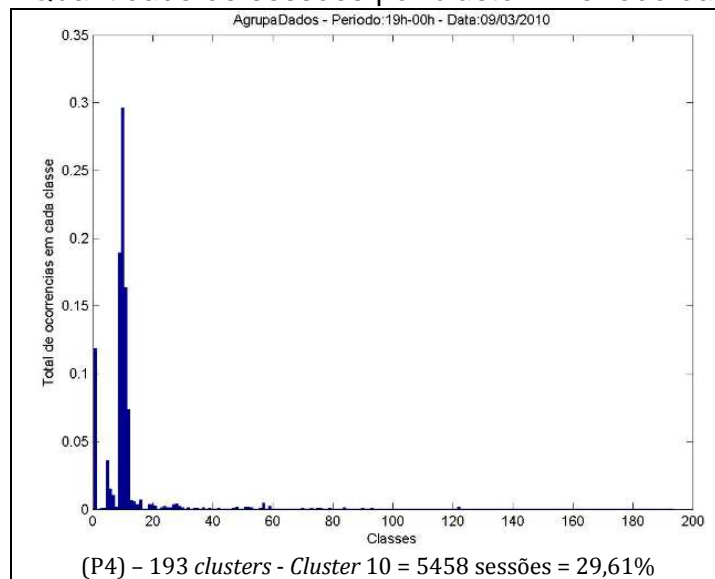


Figura 5.25 - Quantidade de sessões por *cluster* - Período das 19h às 0h

Diferentes análises foram realizadas a fim de extrair conhecimento para melhor caracterizar o tráfego. Foram realizados experimentos com o conjunto de dados históricos do tráfego utilizando valores de parâmetros  $ds$  (taxa de desvio) e  $sim$  (taxa de similaridade) diferentes ( $ds=15\%$  e  $sim=100\%$ ;  $ds=15\%$  e  $sim=70\%$ ;  $ds=10\%$  e  $sim=100\%$ ). Observaram-se melhores resultados de

clusterização (menos *clusters* gerados) utilizando os parâmetros  $ds= 10\%$  e  $sim=70\%$  como margem de tolerância para posicionamento das sessões no *cluster* de maior similaridade, os quais são apresentados na Tabela 5.6.

Tabela 5.6. *Clusters* gerados - Parâmetros  $ds= 10\%$  e  $70\%$  de similaridade

Parâmetros: (ds)= 10% e sim= 70%					
Série Temporal	Dia da Semana	Total de <i>Clusters</i> Gerados			
		P1	P2	P3	P4
S09032010	Terça-feira	98	8471	8262	254
S15032010	Segunda-feira	46	8710	8775	301
S28032010	Domingo	62	50	55	49
S01042010	Quinta-feira	46	6192	5708	24
S14042010	Quarta-feira	79	6620	7758	1020
S16042010	Sexta-feira	68	7317	7178	313
S17042010	Sábado	65	124	54	97

Na aplicação do parâmetro de desvio 10% com similaridade de 70%, o total de *clusters* formado para cada série analisada foi consideravelmente menor, como mostra a Tabela 5.6, quando comparado com os resultados obtidos a partir dos mesmos parâmetros de desvio com similaridade de 100%.

Após o processo de clusterização das séries temporais coletadas, foram investigados os *clusters* mínimos e máximos gerados para todos os dias da semana, nos quatro períodos do dia e utilizando os mesmos pares de parâmetros desvio e similaridade previamente escolhidos para agrupamento das sessões. A Figura 5.26 ilustra os resultados obtidos a partir de  $ds=10\%$  e  $sim=70\%$ .

Dados do tráfego de 05/03/2010 a 05/05/2010 (ds = 10% e 70%)			
Período: 00_07h			
Dia da Semana	Min Grupos	Max Grupos	
Domingo	22	141	
Segunda-feira	20	144	
Terça-feira	34	213	
Quarta-feira	50	223	
Quinta-feira	46	129	
Sexta-feira	34	141	
Sábado	19	162	
No período:	19	223	
Período: 13_19h			
Dia da Semana	Min Grupos	Max Grupos	
Domingo	13	376	
Segunda-feira	6565	8775	
Terça-feira	6376	8857	
Quarta-feira	111	8329	
Quinta-feira	5708	7864	
Sexta-feira	24	8158	
Sábado	14	334	
No período:	13	8857	
Período: 07_13h			
Dia da Semana	Min Grupos	Max Grupos	
Domingo	12	149	
Segunda-feira	7407	9302	
Terça-feira	6092	8471	
Quarta-feira	123	8819	
Quinta-feira	6192	8706	
Sexta-feira	34	8093	
Sábado	10	164	
No período:	10	9302	
Período: 19_00h			
Dia da Semana	Min Grupos	Max Grupos	
Domingo	9	170	
Segunda-feira	73	451	
Terça-feira	34	813	
Quarta-feira	37	1020	
Quinta-feira	24	1031	
Sexta-feira	22	422	
Sábado	2	322	
No período:	2	1031	

Figura 5.26 - Quantidade de *clusters* (Mínimos e Máximos) gerados com os parâmetros  $ds=10\%$  e  $sim=70\%$

Outros resultados observados com a clusterização foram: total de *clusters* gerados por arquivo de dado e um controle das posições das sessões nos arquivos de dados. A Tabela 5.7 apresenta uma amostra dos dados levantados na clusterização.



Tabela 5.7 – Exemplo dos dados armazenados no relatório VA.m

Cont	Arquivo	Ngrupos (max(va))
1	P00_07-S10032010.m	223
2	P00_07-S17032010.m	55
3	P00_07-S24032010.m	59
10	P7_13-S10032010.m	8819
11	P7_13-S17032010.m	7011
12	P7_13-S24032010.m	8181
19	P13_19-S10032010.m	8329
20	P13_19-S17032010.m	5691
21	P13_19-S24032010.m	7210
28	P19_00-S10032010.m	344
29	P19_00-S17032010.m	39
30	P19_00-S24032010.m	37

Um exemplo da porcentagem de sessões armazenadas em cada *cluster* é mostrado na Tabela 5.8.

Tabela 5.8 – Amostra da porcentagem de sessões armazenadas em cada *cluster*

Grupo	Sessões por Grupo	Porcentagem de Sessões por Grupo
1	15397	96,3
2	1	0,01
3	15	0,09
4	2	0,01
5	2	0,01
6	3	0,02
7	4	0,03
8	270	1,69
9	1	0,01
10	4	0,03

## 5.5. Caracterização dos Dados

Para caracterizar o conjunto de dados do tráfego de rede histórico, três hipóteses foram admitidas. A partir da primeira hipótese, isto é, que os *clusters* mais populosos representam o comportamento padrão do tráfego, foi possível selecionar, em cada conjunto de dados organizado por dia da semana e período do dia, apenas as sessões dos maiores *cluster* para continuar a caracterização do tráfego.

Estas sessões dos maiores *clusters* foram então armazenadas em uma nova base de dados. A fim de selecionar pelo menos 80% dos dados clusterizados, foram selecionados os 20 maiores *clusters* nos períodos P1 (madrugada) e P4 (noite) e 2000 maiores *clusters* nos períodos P2 (manhã) e P3 (tarde). A Tabela 5.9 apresenta uma amostra dos maiores *clusters*, ou seja, os *clusters* que agruparam maior número de sessões. Cada coluna representa um dia da semana no período da madrugada (P1). Devido ao horário, o tráfego tem um mesmo perfil, independente do dia da semana.

Tabela 5.9 – Os vinte maiores *clusters* do período P1 (madrugada – das 00h as 07h)

Domingos	Segundas	Terças	Quartas	Quintas	Sextas	Sábados
1	1	1	1	1	1	1
226	204	290	10	8	212	242
26	84	304	9	29	71	101
24	9	301	13	3	2	16
96	15	302	11	41	10	31
25	5	298	3	49	57	43
10	25	312	2	258	5	160
21	5	323	4	181	213	2
28	25	320	30	310	84	83
158	157	293	67	43	118	2
28	7	309	30	25	130	83
158	113	19	67	32	118	161
124	7	111	35	42	130	165
122	113	103	15	25	31	8
155	247	324	166	32	120	123
122	151	307	37	42	12	8
155	27	314	66	25	9	123
54	52	288	37	32	219	28
104	27	329	66	42	217	122
29	52	306	54	50	138	25

A Tabela 5.10 apresenta um resumo da quantidade de maiores *clusters* selecionados para cada dia da semana e período do dia. Nos períodos P1 e P2 é selecionada uma menor quantidade de *clusters* devido à maior similaridade destes dados, armazenando na média mais de 80% dos dados já no primeiro *cluster*.

Tabela 5.10 – Resumo dos maiores *clusters*

<b>Dia da Semana</b>	<b>00_07</b>	<b>07_13</b>	<b>13_19</b>	<b>19_00</b>
Domingo	20	20	20	20
Segunda	20	2000	2000	20
Terça	20	2000	2000	20
Quarta	20	2000	2000	20
Quinta	20	2000	2000	20
Sexta	20	2000	2000	20
Sábado	20	20	20	20

### 5.5.1. Limiarização dos Dados

A técnica de limiarização foi desenvolvida no intuito de definir os limiares inferiores e superiores (*z-score*) indicativos de comportamento padrão de cada atributo do tráfego histórico em cada período do dia.

A técnica de limiarização baseada em *z-score* foi aplicada, sendo os valores de *z-score* para todos os atributos das sessões definidos a partir do cálculo da média e desvio padrão do tráfego histórico todo analisado (2 meses de dados).

Os valores de *z-score* calculados para os atributos *brevCL* e *dataDir* das sessões de um mês de tráfego histórico, organizados por período do dia, são apresentados nas Figuras 5.27 a 5.34.

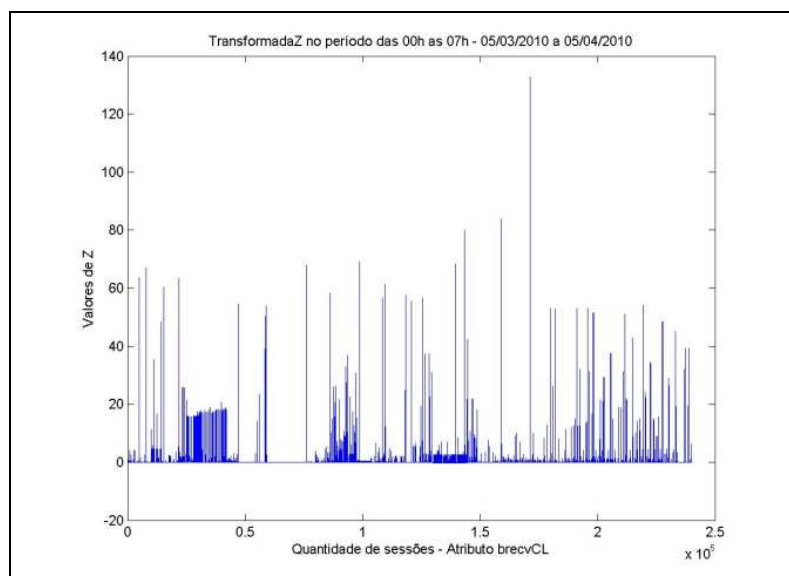


Figura 5.27 – Valores de z-score do atributo brevcCL - Período 00h as 07h durante um mês de captura de tráfego de rede

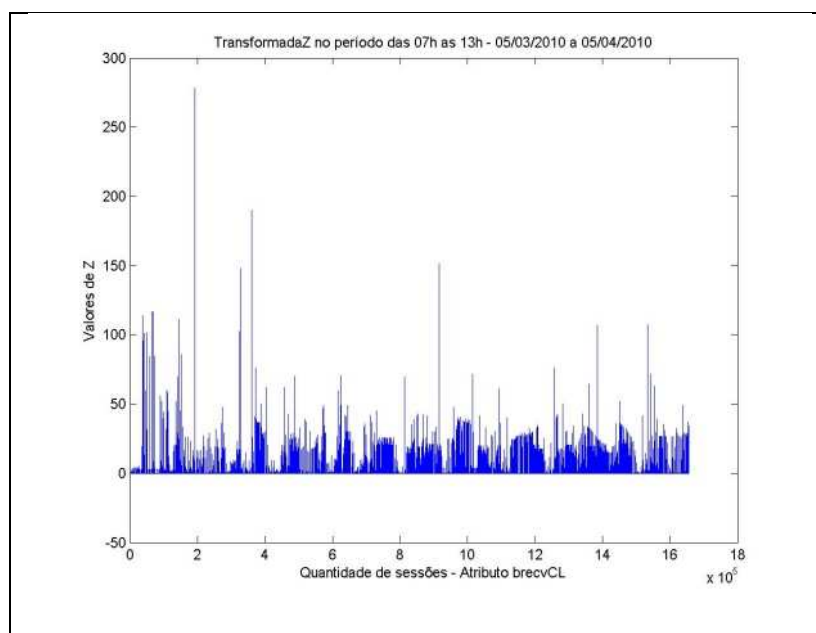


Figura 5.28 Valores de z-score do atributo brevcCL – Período das 07h às 13h durante um mês de captura de tráfego de rede

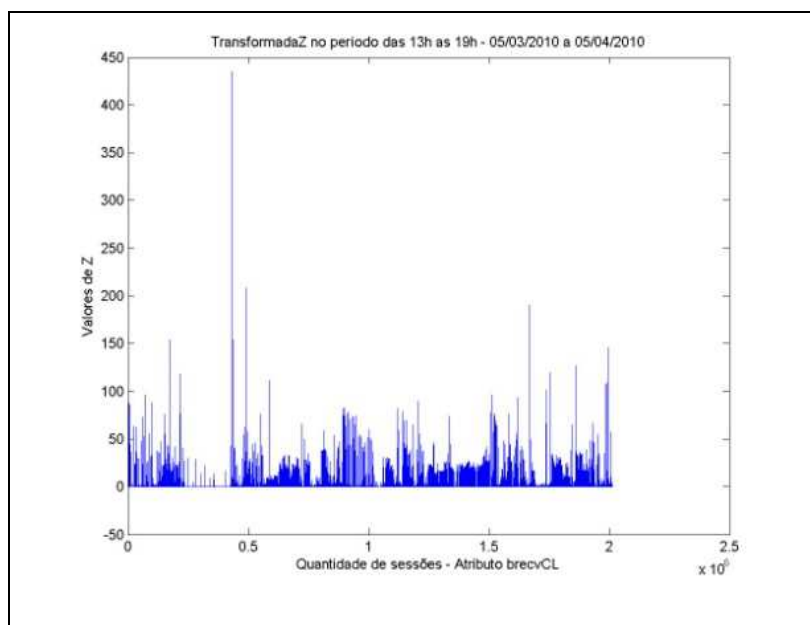


Figura 5.29 - Valores de z-score do atributo brevcCL – Período das 13h às 19h durante um mês de captura de tráfego de rede

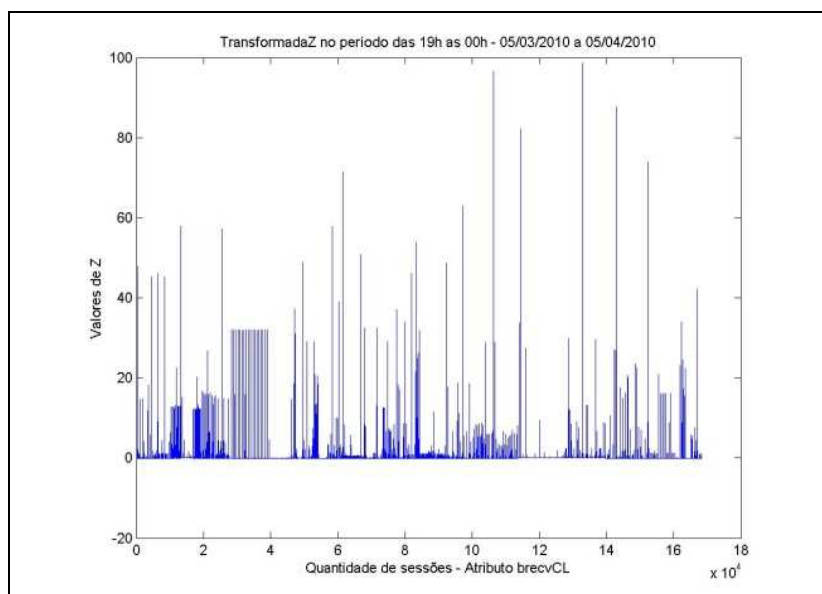


Figura 5.30 - Valores de z-score do atributo brevcCL - Período das 19h às 0h durante um mês de captura de tráfego de rede

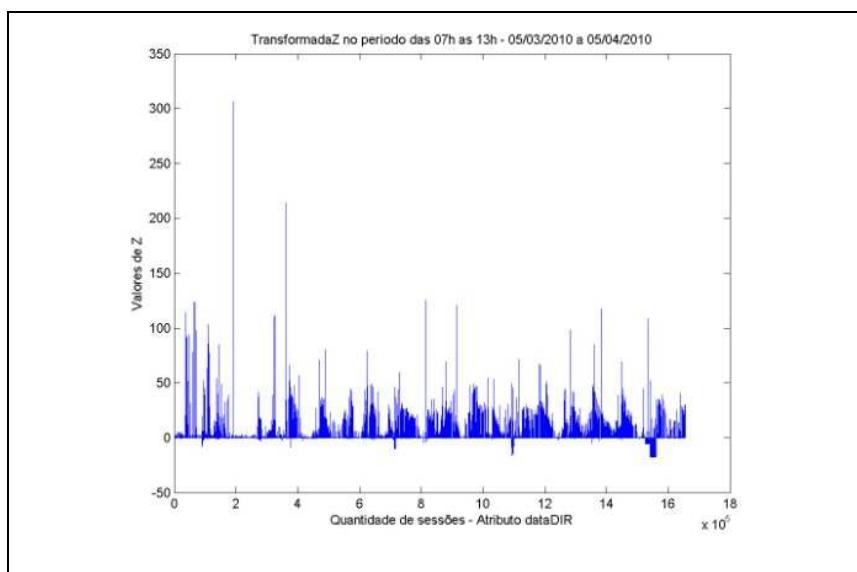


Figura 5.31 – Valores de z-score do atributo dataDir - Período 00h as 07h durante um mês de captura de tráfego de rede

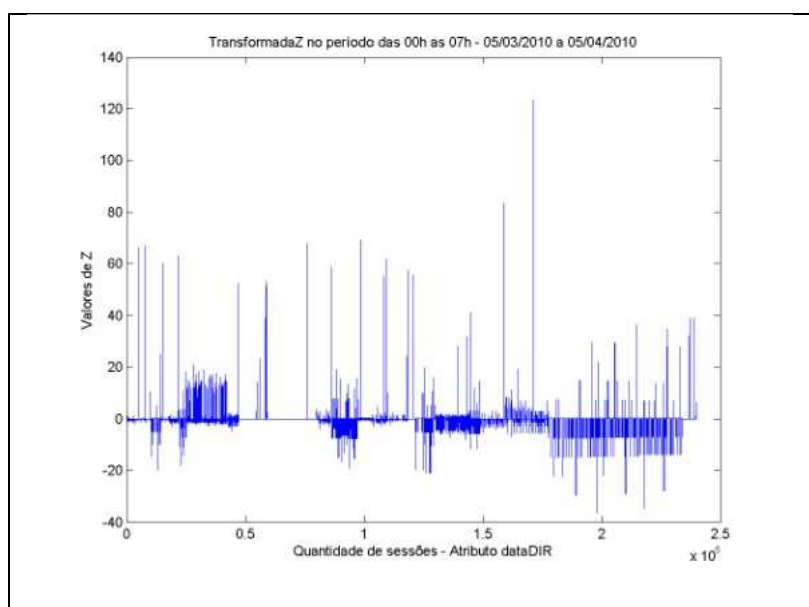


Figura 5.32 Valores de z-score do atributo dataDir – Período das 07h às 13h durante um mês de captura de tráfego de rede

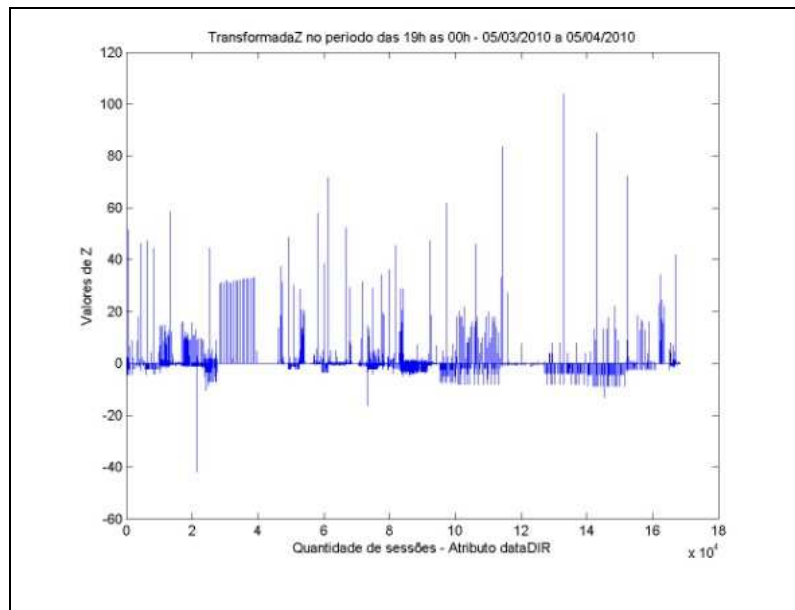


Figura 5.33 - Valores de z-score do atributo dataDir – Período das 13h às 19h durante um mês de captura de tráfego de rede

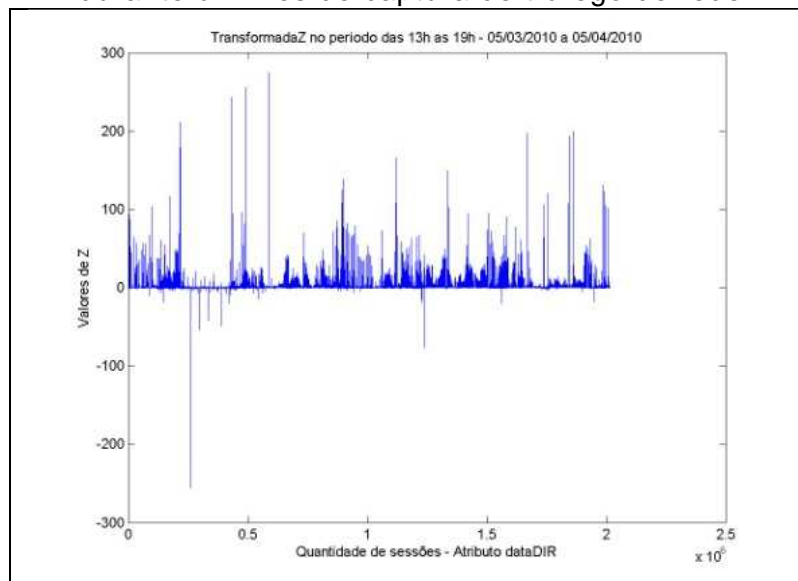


Figura 5.34 - Valores de z-score do atributo dataDir - Período das 19h às 0h durante um mês de captura de tráfego de rede

Valores de *z-score* também foram calculados para os nove atributos das sessões do tráfego histórico de 2 meses de dados. As Figuras 5.35 a 5.42 apresentam os valores de *z-score* dos atributos brecvCL e dataDir do tráfego.



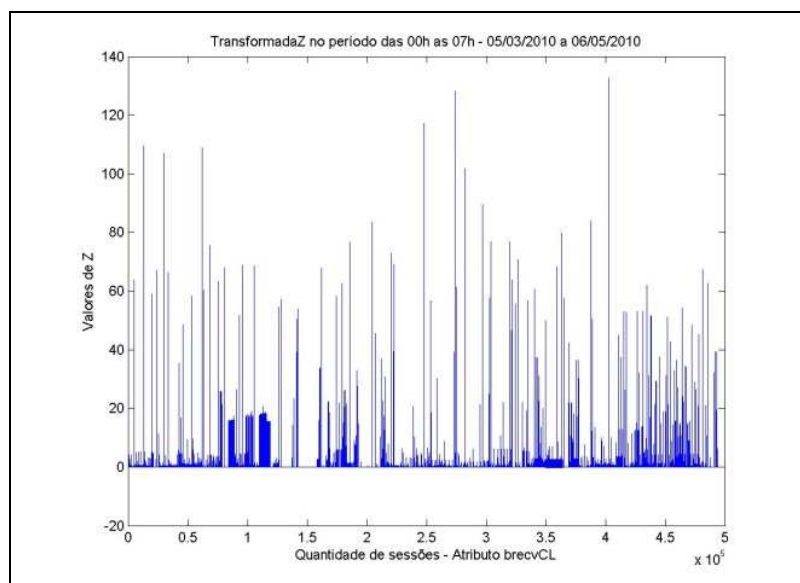


Figura 5.35 – Valores de z-score do atributo brevcCL - Período 00h as 07h durante dois meses de captura de tráfego de rede

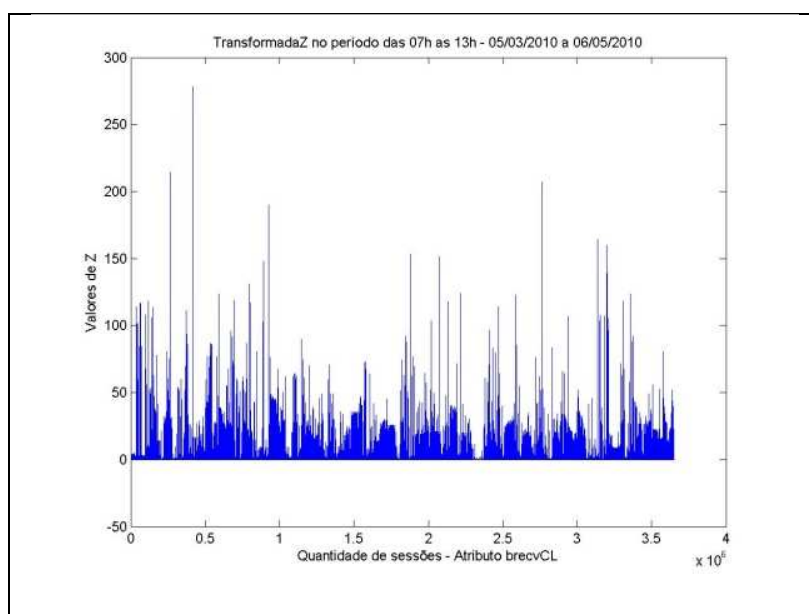


Figura 5.36 Valores de z-score do atributo brevcCL – Período das 07h às 13h durante dois meses de captura de tráfego de rede

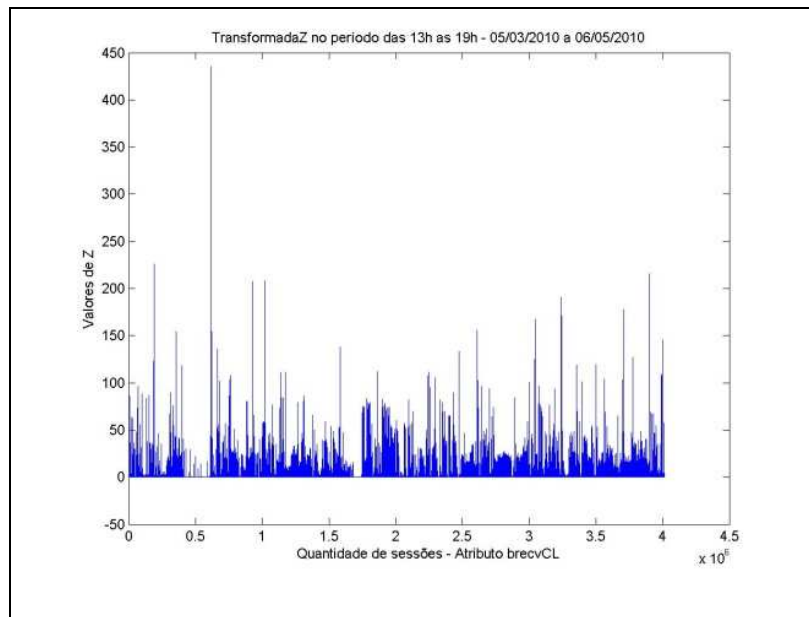


Figura 5.37 - Valores de z-score do atributo brevcCL – Período das 13h às 19h durante dois meses de captura de tráfego de rede

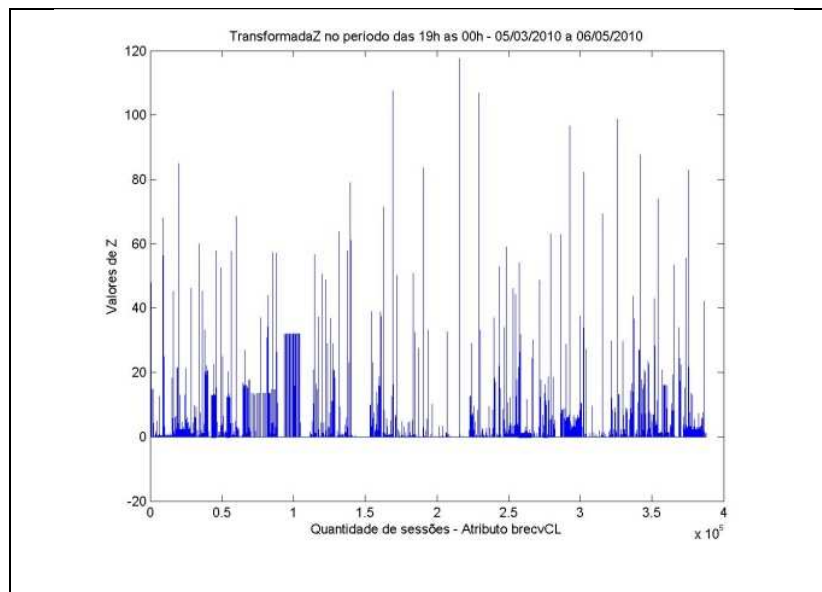


Figura 5.38 - Valores de z-score do atributo brevcCL - Período das 19h às 0h durante dois meses de captura de tráfego de rede

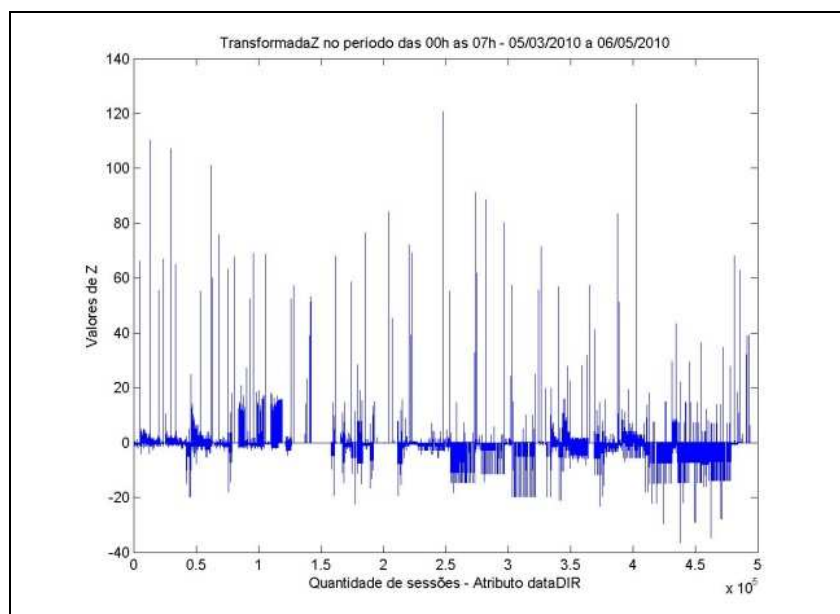


Figura 5.39 – Valores de z-score do atributo dataDir - Período 00h as 07h durante dois meses de captura de tráfego de rede

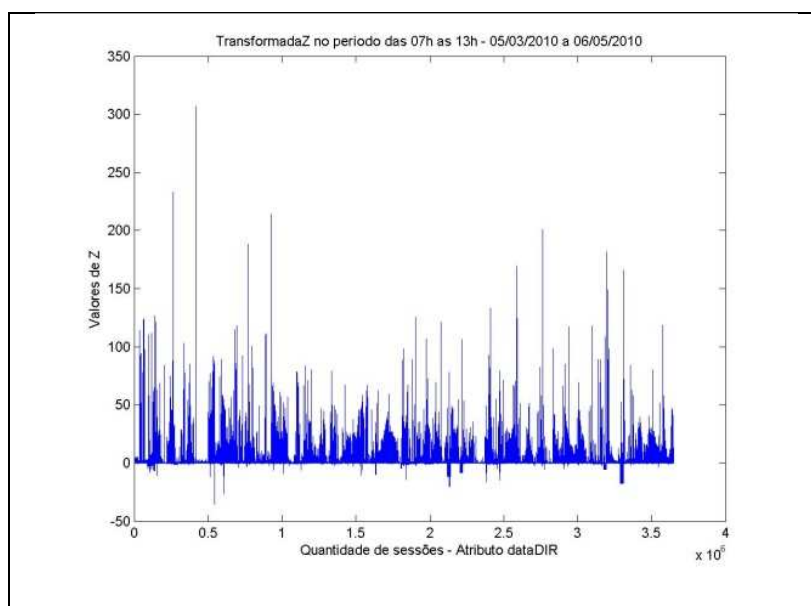


Figura 5.40 Valores de z-score do atributo dataDir – Período das 07h às 13h durante dois meses de captura de tráfego de rede

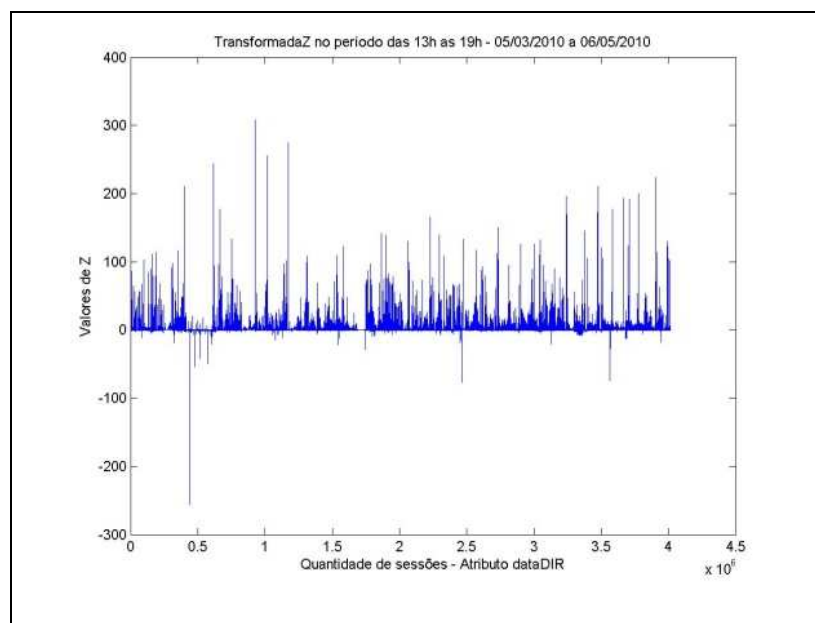


Figura 5.41 - Valores de z-score do atributo dataDir – Período das 13h às 19h durante dois meses de captura de tráfego de rede

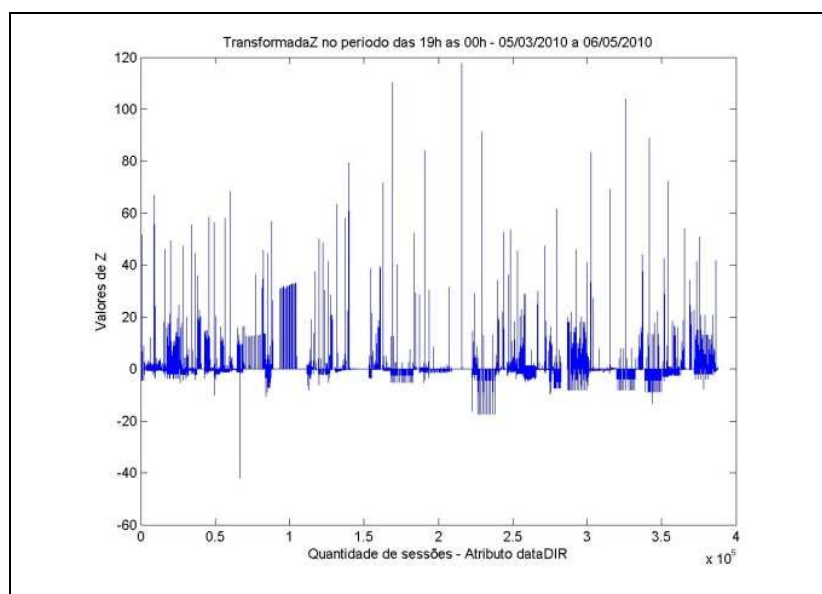


Figura 5.42 - Valores de z-score do atributo dataDir - Período das 19h às 0h durante dois meses de captura de tráfego de rede

Após o cálculo do *z-score* para cada um dos nove atributos de todas as sessões do conjunto do tráfego histórico analisado, realizou-se uma análise visual dos conjuntos de dados.

Observou-se, portanto, que a maioria dos valores de *z-score* dos atributos das sessões dos maiores *clusters* era similar, pois pertenciam as regiões de maior concentração de dados, ou seja, pertenciam as regiões do gráfico com fluxo de linhas mais intenso.

Com base nesta observação e considerando a hipótese de que a maior parte do tráfego histórico possui comportamento padrão, concluiu-se que os maiores *clusters* dos conjuntos analisados representariam o comportamento padrão do conjunto, contendo as sessões mais similares.

Deste modo, a partir da seleção dos maiores *clusters*, foi gerada uma base de dados menor (tamanho total de 759 MB), reduzindo 20% à base do tráfego histórico inicial. As sessões desta base reduzida foram, então, rotuladas com base nos limiares de *z-score* calculados nesta etapa do trabalho.

O cálculo do *z-score* considerou a média e desvio padrão de dois meses de tráfego de rede no conjunto de dados organizado por período do dia e dia da semana. A Tabela 5.11 abaixo apresenta os limiares de *z-score* mínimos para cada atributo do tráfego histórico.

Tabela 5.11 - Tabela com limiares z-score mínimos

Dia da Semana	Período do dia	psizeCL	psizeSV	pnumCL	pnumSV	samlpkt	dataDir	brecvCL	brecvSV	duration
Dom	P00	-0,10	-0,11	-0,05	-0,08	-10,73	-3,02	-0,02	-0,05	-0,27
Qua	P00	-0,43	-0,56	-0,03	-0,03	-6,01	-0,07	-0,02	-0,23	-0,29
Qui	P00	-0,09	-0,11	-0,02	-0,02	-8,22	-0,06	-0,02	-0,09	-0,31
Sab	P00	-0,10	-0,11	-0,01	-0,01	-8,83	-0,36	-0,01	-0,05	-0,30
Seg	P00	-0,09	-0,11	-0,07	-0,08	-8,54	-6,37	-0,04	-0,05	-0,28
Sex	P00	-0,10	-0,13	-0,01	-0,01	-9,12	-0,03	-0,01	-0,08	-0,31
Ter	P00	-0,26	-0,42	-0,13	-0,19	-5,58	-2,56	-0,03	-0,26	-0,31
Dom	P07	-0,09	-0,11	-0,04	-0,05	-10,78	-2,74	-0,02	-0,05	-0,26
Qua	P07	-0,60	-0,87	-0,03	-0,03	-5,01	-0,07	-0,03	-0,09	-0,33
Qui	P07	-0,60	-0,87	-0,02	-0,02	-4,99	-0,02	-0,02	-0,13	-0,33
Sab	P07	-0,11	-0,12	-0,06	-0,07	-9,18	-3,44	-0,04	-0,05	-0,29
Seg	P07	-0,59	-0,91	-0,03	-0,03	-5,10	-0,04	-0,03	-0,10	-0,33
Sex	P07	-0,60	-0,87	-0,05	-0,06	-4,95	-0,07	-0,05	-0,22	-0,34
Ter	P07	-0,58	-0,90	-0,04	-0,05	-5,12	-0,05	-0,04	-0,10	-0,33
Dom	P13	-0,11	-0,12	-0,01	-0,01	-9,92	-0,24	-0,01	-0,06	-0,27
Qua	P13	-0,58	-0,84	-0,05	-0,05	-5,22	-0,12	-0,04	-0,12	-0,32
Qui	P13	-0,54	-0,87	-0,03	-0,03	-5,50	-0,08	-0,03	-0,06	-0,31
Sab	P13	-0,13	-0,15	-0,01	-0,01	-8,92	-0,04	-0,01	-0,07	-0,28
Seg	P13	-0,47	-0,74	-0,03	-0,04	-5,46	-1,37	-0,03	-0,14	-0,31
Sex	P13	-0,54	-0,84	-0,04	-0,04	-5,38	-0,11	-0,04	-0,15	-0,31
Ter	P13	-0,53	-0,75	-0,03	-0,03	-5,24	-0,10	-0,03	-0,06	-0,31
Dom	P19	-0,11	-0,12	-0,09	-0,10	-10,25	-2,44	-0,05	-0,05	-0,28
Qua	P19	-0,50	-0,64	-0,04	-0,05	-2,62	-0,05	-0,04	-0,57	-0,30
Qui	P19	-0,23	-0,26	-0,02	-0,03	-6,28	-0,05	-0,03	-0,09	-0,31
Sab	P19	-0,14	-0,16	-0,01	-0,01	-9,04	-0,03	-0,00	-0,07	-0,27
Seg	P19	-0,29	-0,41	-0,03	-0,04	-5,35	-0,31	-0,02	-0,25	-0,30
Sex	P19	-0,20	-0,31	-0,02	-0,02	-7,60	-0,07	-0,01	-0,13	-0,27
Ter	P19	-0,43	-0,57	-0,02	-0,02	-5,12	-0,03	-0,02	-0,23	-0,31

Conforme apresentado na Tabela 5.11, foi calculado um limiar mínimo de z-score para cada atributo, por dia da semana e período do dia. Estes limiares definem os valores mínimos do comportamento padrão, aceitáveis para cada atributo.

A Tabela 5.12 apresenta os limiares máximos para cada atributo do tráfego histórico de 2 meses, de acordo com o dia da semana e período do dia.

Tabela 5.12 - Tabela com limiares z-score máximos

Dia da Semana	Período do dia	psizeCL	psizeSV	pnumCL	pnumSV	samllpkt	dataDir	brecvCL	brecvSV	duration
Dom	P00	17,38	9,41	7,14	9,39	0,41	2,25	6,02	2,23	9,26
Qua	P00	2,81	6,05	0,20	0,30	0,84	0,03	0,01	13,17	7,80
Qui	P00	19,82	6,61	46,63	46,79	0,63	47,05	46,48	5,30	7,36
Sab	P00	9,22	10,29	0,19	0,32	0,42	-0,00	0,07	2,19	8,24
Seg	P00	9,66	9,58	5,45	6,11	0,43	1,58	1,64	1,95	8,38
Sex	P00	6,56	11,18	0,02	0,03	0,52	0,00	-0,00	4,71	7,78
Ter	P00	7,40	4,62	1,92	3,17	0,78	0,65	0,60	8,81	7,64
Dom	P07	11,87	9,84	2,54	3,64	0,43	1,35	0,59	3,07	9,65
Qua	P07	3,84	5,74	0,11	0,13	1,40	0,14	0,10	2,45	9,60
Qui	P07	3,87	5,18	7,50	8,85	1,38	5,61	7,26	0,58	9,68
Sab	P07	10,34	10,58	3,83	4,62	0,43	0,85	0,73	2,33	8,94
Seg	P07	4,01	4,39	60,18	64,43	1,40	37,88	63,68	0,73	10,37
Sex	P07	3,58	5,01	0,08	0,09	1,35	0,11	0,07	3,20	9,46
Ter	P07	3,75	5,64	0,04	0,05	1,38	0,06	0,04	0,68	9,81
Dom	P13	15,06	8,95	0,36	0,41	0,44	0,29	0,29	2,02	8,13
Qua	P13	4,12	5,09	0,32	0,33	1,35	0,64	0,28	0,92	9,55
Qui	P13	4,23	4,87	0,09	0,11	1,31	0,13	0,11	0,36	9,63
Sab	P13	8,50	9,29	0,05	0,07	0,41	0,01	0,03	2,28	8,84
Seg	P13	5,03	5,19	30,16	32,32	1,04	27,11	31,98	1,74	11,01
Sex	P13	4,45	5,28	34,84	38,77	1,28	49,75	35,27	0,99	9,23
Ter	P13	4,43	4,85	12,61	15,28	1,14	9,90	13,17	0,39	9,40
Dom	P19	18,21	8,09	10,61	11,97	0,40	7,25	14,92	1,73	7,44
Qua	P19	4,22	2,27	-0,03	-0,02	0,94	-0,02	-0,03	1,51	7,16
Qui	P19	6,93	8,74	0,14	0,16	0,63	0,14	0,13	1,63	6,96
Sab	P19	9,00	8,16	0,04	0,06	0,38	0,02	0,02	1,93	8,80
Seg	P19	3,34	4,94	0,16	0,32	0,75	0,02	0,01	7,32	7,75
Sex	P19	8,14	6,08	0,06	0,07	0,55	0,04	0,04	1,82	7,42
Ter	P19	1,45	5,68	-0,02	-0,01	0,92	-0,02	-0,02	1,30	6,70

Da mesma forma, foi calculado um limiar máximo de z-score para cada atributo, por dia da semana e período do dia. Estes limiares apresentados na

Tabela 5.12, definem os valores máximos do comportamento padrão aceitáveis para cada atributo.

### **Rotulação dos Dados**

A caracterização do tráfego de rede, neste trabalho, compreende a construção de um modelo que represente o comportamento padrão do tráfego. Este modelo pode ser construído através da rotulação das sessões do tráfego.

Segundo a metodologia TRAFICIN, a rotulação das sessões do tráfego de rede consiste em: comparar o valor de *z-score* dos nove atributos da sessão histórica com os respectivos limiares mínimos e máximos. Se até 90% dos atributos da sessão possuir os valores de *z-score* menores que os respectivos limiares mínimos, a sessão é classificada com o rótulo “-1”, ou seja, “anômala abaixo da média”. Se até 90% dos atributos da sessão possuir os valores de *z-score* dentro da faixa dos limiares mínimos e máximos, a sessão é classificada com o rótulo “0”, ou seja, “padrão”. Caso contrário, se até 90% dos atributos comparados estiver acima dos respectivos limiares máximos, a sessão é classificada com o rótulo “1”, ou seja, “anômala acima da média”.

O modelo do comportamento padrão do tráfego de rede mapeado foi utilizado para o treinamento dos classificadores, no intuito de detectar possíveis anomalias na rede.

Para melhorar a precisão dos resultados de detecção, o comportamento da rede deve ser adequadamente mapeado e constantemente atualizado para contemplar as mudanças ocorridas no ambiente. A atualização do modelo característico da rede deve ser realizada periodicamente (recomenda-se uma vez ao mês) ou sempre que houver mudanças programadas e significativas na rede.



Um sistema baseado em regras de produção foi desenvolvido com o objetivo de rotular as sessões do tráfego de rede como padrão “0”, anômala acima da média “1” ou anômala abaixo da média “-1”, baseado nos limiares  $z$  dos *clusters* mais populosos.

Duas bases de dados para treinamento do classificador foram geradas nesta etapa do trabalho: arquivos contendo valores de  $z$ -score dos atributos necessários para rotulação da base histórica e arquivos de valores reais de atributos rotulados (base histórica efetivamente caracterizada com sessões rotuladas). A Tabela 5.13 apresenta uma amostra dos dados rotulados.

Tabela 5.13: Sessões aleatórias - valores reais de atributos classificadas pelo sistema

psizeCL	psizeSV	pnumCL	pnumSV	smallpkt	dataDir	brecvCL	brecvSV	duration	Class
122.67	442.09	21.00	22.00	0.63	-1.00	2576.00	2576.00	369.01	0
1114.52	161.79	44.00	39.00	0.45	5.00	49039.00	49039.00	7.00	0
1070.03	95.43	101.00	82.00	0,45	19.00	108073.00	108073.00	70.4323	1
1198.00	58.26	208.00	151.00	0,43	57.00	249184.00	249184.00	130.73	1
0.00	0.00	0.00	0.00	0,81	0.00	0.00	0.00	3.62	-1
0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	20.22	-1

A etapa de rotulação dos dados é a fase final do processo de caracterização do tráfego de rede padrão. Como resultado desta etapa, tem-se as sessões dos conjuntos de dados do tráfego de rede organizados por dia da semana e período do dia, classificadas como “sessão anômala abaixo da média” (-1), “sessão padrão” (0) ou “sessão anômala acima da média” (1).

As sessões dos conjuntos de dados de tráfego histórico de 2 meses reduzido para uma base contendo apenas sessões dos maiores *clusters*, foram então rotuladas pelo sistema como 0, +1 e -1.

Conforme apresentado na Tabela 5.14, a maior parte das sessões do tráfego histórico dos maiores *clusters* foi rotulada como padrão (0), o que permite

inferir que esta base de conhecimento é satisfatória para treinar o classificador para detecção de anomalias.

Tabela 5.14 – Percentual de Sessões Rotuladas

Período	Padrão (0)	Anomalias (+1)	Anomalias (-1)
P00-07	90,27%	9,46%	0,28%
P07-13	99,92%	0,07%	0,00%
P13-19	99,46%	0,01%	0,00%
P19-00	86,27%	0,13%	0,00%

## 5.6. Detecção de Anomalias no tráfego de rede

Duas abordagens para detectar anomalias nas sessões do tráfego corrente da rede foram adotadas no TRAFICIN: classificação baseada em análise espacial e classificação baseada em análise temporal.

### 5.6.1. Classificação baseada em Análise Espacial dos dados

A classificação baseada em análise espacial dos dados é um processo relacionado à associação de uma classe a cada uma das sessões do tráfego de rede analisadas.

Na classificação espacial do tráfego cada valor de atributo das sessões correntes é analisado para classificar a sessão como anômala ou padrão. A base de dados histórica de dois meses de tráfego de rede caracterizada com o TRAFICIN é utilizada na etapa de treinamento dos métodos de classificação para detectar anomalias na rede. Após o treinamento, novas sessões correntes foram aplicadas para teste/validação do TRAFICIN, sendo classificadas como anômalas abaixo da média (-1), padrão (0) ou anômalas acima da média (1).

A fim de obter uma comparação de resultados em relação a taxas de falsos positivos gerados na detecção de anomalias, vários métodos foram aplicados na classificação do tráfego de rede. Estes métodos foram: rede neural MLP (MultiLayer Perceptron), SVM (Support Vector Machine), RBF (Radial Basis

Functions), Árvore de Decisão e Máxima Verossimilhança.

Devido ao número bem maior de sessões caracterizadas como padrão (0) em relação aos outros padrões (1 e -1), foi necessário realizar uma nova clusterização dos dados da classe padrão, sub-categorizando a classe padrão, gerando novo *cluster* para cada período do dia, equilibrando o número de sessões das três classes utilizadas. Este processo foi necessário para evitar o problema de polarização ou saturação dos classificadores, evitando que eles aprendessem mais uma classe do que outra. Além da eliminação de dados redundantes classificados como padrão, a nova clusterização diminuiu significativamente o tamanho dos arquivos de dados utilizados no treinamento dos classificadores, sem perda de dados relevantes.

Nesta etapa do trabalho, para a classificação das sessões do tráfego utilizando a rede MLP, foi implementada uma topologia de rede com três camadas: uma camada de entrada, uma intermediária e uma de saída. Cada camada foi modelada com os seguintes números de neurônios: nove neurônios (um para cada atributo de sessão de rede estudado) na camada de entrada, dez neurônios na camada intermediária e um neurônio na camada de saída, conforme ilustrado na Figura 5.43.

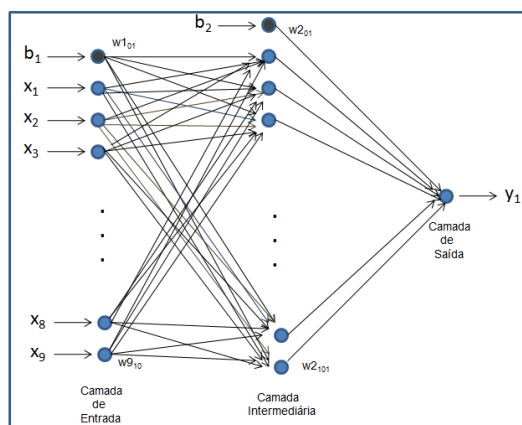


Figura 5.43 - Arquitetura 9-10-1 da MLP

O vetor de saída desejado para esta rede MLP consiste nas classes  $\{-1, 0, 1\}$  onde “0” equivale à classe de sessões padrão, “1” corresponde à classe de

sessões anômalas que apresentam valores de atributos muito acima da média (conforme o limiar máximo estabelecido) e “-1” equivale à classe de sessões anômalas que apresentam valores de atributos muito abaixo da média (conforme o limiar mínimo estabelecido).

Para o treinamento da rede MLP foi utilizado o conjunto de dados de dois meses de tráfego, cujas sessões foram rotuladas pelo sistema como “-1”, “0” ou “1”.

Na fase de treinamento, antes de apresentar os dados pré-classificados em vetores de entrada para a rede ( $x$ ), foi necessário escolher a função de ativação mais apropriada para processar os dados. Como as saídas desejadas encontram-se na faixa de valores entre “-1” e “1”, foi utilizada a função sigmóide bipolar, descrita a seguir:

$$f(x) = (2/(1 + \exp(-x))) - 1 \quad (5.1)$$

Antes dos dados serem inseridos na rede, um pré-processamento é necessário. Devido à natureza diferente das entradas e saída desejadas, os dados devem ser normalizados para que os pesos da rede não saturem. A normalização dos dados no início do algoritmo transforma cada conjunto de dados em valores no intervalo de -1 a 1. Dentre os tipos de normalização descritos na literatura, foi aplicada a fórmula descrita na equação 4.2, que normaliza os dados através dos valores máximos e mínimos de cada atributo da sessão:

$$y = (ls(x - min) + li(max - x))/(max - min) \quad (5.2)$$

Onde,  $ls$  = limites superiores dos atributos,  $li$  = limites inferiores e  $x$  o vetor de entrada da rede neural MLP. Na etapa de treinamento supervisionado da rede neural MLP, os parâmetros de taxa de aprendizagem e erro desejado foram definidos como 0.8 e 0.008 respectivamente.

Os outros classificadores aplicados neste trabalho, tais como, MaxVer, SVM, RBF e Árvore de Decisão foram configurados através do software livre Weka (*Waikato Environment for Knowledge Analysis*), que compreende um conjunto

de ferramentas de aprendizagem de máquina, escrito em Java, para visualização, análise e modelagem de dados preditiva.

Após a nova clusterização, com a base de dados armazenando somente as sessões mais representativas de cada classe, o treinamento foi iniciado. Vários estudos de caso foram realizados e são apresentados a seguir.

#### **a) Estudo de caso 1**

Para o estudo de caso 1, o subconjunto de treinamento utilizado nos classificadores foi gerado através da metodologia TRAFICIN, com duas clusterização: uma dos dados agrupados por dia da semana e período do dia e a outra nos dados rotulados como padrão (0). A Tabela 5.15 apresenta o subconjunto gerado com dados relevantes e compactos:

Tabela 5.15 - Subconjunto de treinamento dos classificadores com nova clusterização

<b>Conj. Treino</b>	<b>Período</b>	<b>N. de sessões</b>	<b>% Sessões -1</b>	<b>% Sessões 0</b>	<b>% Sessões 1</b>
TRE1	P1	3410	30,85	32,95	36,20
TRE2	P2	29.080	0	54,00	46,00
TRE3	P3	56.359	32,50	39,38	28,12
TRE4	P4	6.920	0	57,40	42,60

Como visto na Tabela 5.15, nos períodos P2 e P4 não houve ocorrência de classe rotulada como -1 (anômalo abaixo da média). Nos demais períodos as três classes foram classificadas.

As Tabelas 5.16 e 5.17 apresentam uma comparação entre os classificadores MLP, MaxVer, SVM, J48\_Tree e RBF, destacando a taxa de acerto e o índice kappa alcançados no treinamento.

Tabela 5.16 – Taxas de acerto no treinamento utilizando a base gerada no TRAF CIN

Treinamento					
Período	MLP	MaxVer	SVM	J48_Tree	RBF
	Taxa de Acerto	Taxa de Acerto	Taxa de Acerto	Taxa de Acerto	Taxa de Acerto
P1	90,47%	89,12%	87,00%	96,60%	79,04%
P2	92,35%	92,50%	83,90%	96,54%	76,71%
P3	92,50%	90,76%	83,11%	96,13%	80,30%
P4	92,33%	91,80%	80,27%	96,24%	62,79%
Média	91,91%	91,05%	83,57%	96,38%	74,71%

Tabela 5.17 – Índices Kappa no treinamento utilizando a base gerada no TRAF CIN

Treinamento					
Período	MLP	MaxVer	SVM	J48_Tree	RBF
	Índice Kappa	Índice Kappa	Índice Kappa	Índice Kappa	Índice Kappa
P1	0,86	0,84	0,80	0,95	0,68
P2	0,85	0,85	0,68	0,93	0,54
P3	0,89	0,86	0,74	0,94	0,70
P4	0,84	0,83	0,59	0,92	0,15
Média	0,85	0,85	0,70	0,94	0,52

O resultado do treinamento dos classificadores utilizando o conjunto de dados modelado pelo TRAF CIN apresentou uma taxa de acerto superior a 90% e índice kappa acima de 0,80, na maioria dos classificadores, comprovando o resultado positivo da metodologia TRAF CIN. Dos classificadores aplicados, o J48\_Tree e a rede MLP tiveram resultados mais satisfatórios.

O tempo de treinamento dos classificadores foi muito baixo (2 a 15 minutos), visto que o tamanho do conjunto de dados foi reduzido nas fases anteriores do TRAF CIN, restando apenas os vetores característicos de sessões significativas do comportamento do tráfego, sem redundância de sessões.

Uma vez treinados, os classificadores foram testados com novas sessões do tráfego coletado no mês de maio/2010, conforme mostra a Tabela 5.18.

Tabela 5.18 - Subconjuntos de Testes dos classificadores

Sub-conjunto	Período do tráfego	No. de sessões	% Sessões -1	% Sessões 0	% Sessões 1
TES1	P00	932	30,90	36,16	32,94
TES2	P07	7271	28,33	33,03	38,64
TES3	P13	14090	14,24	19,93	65,82
TES4	P19	1730	0	42,60	57,40

As Tabelas 5.19 e 5.20 apresentam uma comparação dos resultados dos testes entre os classificadores MLP, MaxVer, SVM, J48\_Tree e RBF, destacando a taxa de acerto e o índice kappa alcançados no estudo de caso 1.

Tabela 5.19 – Taxa de Acerto dos classificadores na fase de teste do estudo de caso 1

Teste					
Período	MLP	MaxVer	SVM	J48_Tree	RBF
	Taxa de Acerto	Taxa de Acerto	Taxa de Acerto	Taxa de Acerto	Taxa de Acerto
P1	93,88%	89,27%	87,44%	96,99%	78,65%
P2	98,65%	97,19%	92,84%	98,46%	82,12%
P3	96,99%	93,72%	87,45%	96,12%	82,88%
P4	93,06%	93,70%	83,70%	96,17%	60,98%
Média	95,65%	93,47%	87,86%	96,94%	76,16%

Tabela 5.20 – Índice Kappa dos classificadores na fase de teste do estudo de caso 1

Teste					
Período	MLP	MaxVer	SVM	J48_Tree	RBF
	Índice Kappa	Índice Kappa	Índice Kappa	Índice Kappa	Índice Kappa
P1	0,91	0,84	0,81	0,95	0,68
P2	0,97	0,94	0,86	0,97	0,65
P3	0,95	0,90	0,81	0,94	0,74
P4	0,86	0,87	0,66	0,92	0,07
Média	0,92	0,89	0,79	0,95	0,54

Dos cinco classificadores testados utilizando os novos dados do tráfego coletado no mês de maio/2010, o classificador RBF apresentou o pior índice kappa (0,54). Os outros classificadores (MLP, MaxVer, J48\_Tree, e RBF) tiveram resultados satisfatórios com taxa de acerto superior a 80%, na maioria destes.

Outra comparação das técnicas de classificação aplicadas foi através das matrizes de confusão.

A matriz de confusão mostra os erros e os acertos na classificação, e também a separabilidade das classes. Quanto menor o erro em cada classe, melhor a classificação do modelo. A matriz de confusão dos treinamentos e testes dos 5 classificadores utilizados na TRAFICIN foram calculadas. Nas Figuras 5.44 e 5.45 são apresentadas amostras da matriz de confusão do treinamento utilizando os classificadores MLP e MaxVer.

Matriz de Confusão - Treino MLP				
Periodo	Classes	anomaloac	padrao	anomaloab
P1	anomaloac	1109	89	22
	padrao	198	826	16
	anomaloab	0	0	1150
P2	anomaloac	12978	1775	-
	padrao	448	13879	-
P3	anomaloac	12386	3392	72
	padrao	746	21437	11
	anomaloab	0	3	18313
P4	anomaloac	2666	282	-
	padrao	249	3723	-

Matriz de Confusão - Teste MLP				
Periodo	Classes	anomaloac	padrao	anomaloab
P1	anomaloac	308	16	10
	padrao	29	272	2
	anomaloab	0	0	295
P2	anomaloac	3326	26	-
	padrao	72	3847	-
P3	anomaloac	1229	116	4
	padrao	15	1612	0
	anomaloab	0	1	1544
P4	anomaloac	2275	177	-
	padrao	231	3197	-

Figura 5.44 –Matriz de confusão do classificador MLP do estudo de caso 1

Matriz de Confusão - Treino MaxVer				
Periodo	Classes	anomaloac	padrao	anomaloab
P1	anomaloac	989	215	16
	padrao	128	900	12
	anomaloab	0	0	1150
P2	anomaloac	12572	854	-
	padrao	1327	14327	-
P3	anomaloac	11974	3834	42
	padrao	1307	20868	19
	anomaloab	0	3	18313
P4	anomaloac	2511	432	-
	padrao	130	3842	-

Matriz de Confusão - Teste MaxVer				
Periodo	Classes	anomaloac	padrao	anomaloab
P1	anomaloac	264	62	8
	padrao	28	273	2
	anomaloab	0	0	295
P2	anomaloac	3198	154	-
	padrao	50	3869	-
P3	anomaloac	1121	228	0
	padrao	47	1572	8
	anomaloab	0	1	1544
P4	anomaloac	2200	251	-
	padrao	118	3310	-

Figura 5.45 – Matriz de confusão do classificador MaxVer do estudo de caso 1

Como é possível observar nas Figuras 5.44 e 5.45, nos 4 períodos de tempo, várias sessões rotuladas como “1” foram erroneamente classificadas como “0”, sendo o recíproco também verdadeiro, na fase de treinamento. Isto mostra uma



similaridade no padrão das duas classes. Enquanto a classe “-1” teve 100% de classificação correta, no período P1 na fase de treinamento nos dois classificadores.

Na fase de teste, como mostrado também nas Figuras 5.44 e 5.45, os classificadores convergiram rapidamente (no máximo 15 minutos) e os resultados também foram satisfatórios. Porém, a confusão com as classes “1” e “0” permaneceu. A classe “-1” teve 100% de acerto no período P1 na fase teste nos dois classificadores.

## b) Estudo de caso 2

No estudo de caso 2, o subconjunto de dados utilizados para o destes classificadores, foi selecionado aleatoriamente como amostras de sessões coletadas a cada intervalo de 10 minutos, objetivando selecionar padrões diferentes coletados durante o período de tempo analisado. A Tabela 5.21 mostra o subconjunto de treinamento gerado.

Tabela 5.21 - Subconjuntos de Treinamento gerado aleatoriamente

Sub-conjunto	Período do tráfego	No. de sessões	% Sessões -1	% Sessões 0	% Sessões 1
TRE1	P1	5062	32,15	38,00	29,85
TRE2	P2	6736	0	52,50	47,50
TRE3	P3	5999	30,50	42,00	27,50
TRE4	P4	4035	0	56,37	43,63

As Tabelas 5.22 e 5.23 apresentam uma comparação entre os classificadores MLP, MaxVer, SVM, J48\_Tree e RBF, destacando a taxa de acerto e o índice kappa alcançados no estudo de caso 2.

Tabela 5.22 - Taxas de acerto dos classificadores no treinamento do estudo de caso 2.

Treinamento					
Período	MLP	MaxVer	SVM	J48_Tree	RBF
	Taxa de Acerto	Taxa de Acerto	Taxa de Acerto	Taxa de Acerto	Taxa de Acerto
P1	72,34%	74,67%	58,08%	83,62%	60,41%
P2	87,75%	82,53%	82,09%	91,67%	73,84%
P3	85,23%	97,71%	63,32%	99,96%	70,84%
P4	77,67%	77,60%	76,48%	82,00%	74,72%
<b>Média</b>	<b>80,75%</b>	<b>83,13%</b>	<b>69,99%</b>	<b>89,31%</b>	<b>69,95%</b>

Tabela 5.23 - Índices Kappa dos classificadores no treinamento do estudo de caso 2

Treinamento					
Período	MLP	MaxVer	SVM	J48_Tree	RBF
	Índice Kappa	Índice Kappa	Índice Kappa	Índice Kappa	Índice Kappa
P1	0,57	0,61	0,34	0,75	0,39
P2	0,75	0,65	0,64	0,83	0,47
P3	0,78	0,97	0,45	0,99	0,56
P4	0,55	0,55	0,53	0,64	0,49
<b>Média</b>	<b>0,66</b>	<b>0,70</b>	<b>0,49</b>	<b>0,80</b>	<b>0,48</b>

O resultado do treinamento dos classificadores utilizando o conjunto de dados selecionado aleatoriamente no tempo apresentou uma taxa de acerto e índice *kappa* médios inferiores ao estudo de caso 1 . Dos classificadores aplicados, os classificadores SVM e a rede RBF tiveram os piores resultados.

Para os testes foi utilizada a base de dados apresentada no estudo de caso 1.

As Tabelas 5.24 e 5.25 apresentam uma comparação entre os classificadores MLP, MaxVer, SVM, J48\_Tree e RBF, destacando a taxa de acerto e o índice kappa alcançados no estudo de caso 2.

Tabela 5.24 - Taxa de Acerto dos classificadores no teste do estudo de caso 2

<b>Teste</b>					
<b>Periodo</b>	<b>MLP</b>	<b>MaxVer</b>	<b>SVM</b>	<b>J48_Tree</b>	<b>RBF</b>
	<b>Taxa de Acerto</b>	<b>Taxa de Acerto</b>	<b>Taxa de Acerto</b>	<b>Taxa de Acerto</b>	<b>Taxa de Acerto</b>
P1	62,62%	74,67%	62,57%	59,44%	65,68%
P2	82,92%	82,53%	78,58%	89,68%	64,49%
P3	82,48%	97,71%	65,09%	99,30%	71,57%
P4	61,90%	77,60%	49,23%	65,47%	49,00%
<b>Média</b>	<b>72,48%</b>	<b>83,13%</b>	<b>63,87%</b>	<b>78,47%</b>	<b>62,69%</b>

Tabela 5.25 – Índice Kappa dos classificadores no teste do estudo de caso 2

<b>Teste</b>					
<b>Periodo</b>	<b>MLP</b>	<b>MaxVer</b>	<b>SVM</b>	<b>J48_Tree</b>	<b>RBF</b>
	<b>Índice Kappa</b>	<b>Índice Kappa</b>	<b>Índice Kappa</b>	<b>Índice Kappa</b>	<b>Índice Kappa</b>
P1	0,43	0,61	0,42	0,38	0,47
P2	0,65	0,65	0,54	0,79	0,25
P3	0,73	0,97	0,48	0,99	0,57
P4	0,24	0,55	-0,02	0,31	-0,02
<b>Média</b>	<b>0,51</b>	<b>0,70</b>	<b>0,36</b>	<b>0,62</b>	<b>0,32</b>

Dos cinco classificadores testados com os conjuntos de dados dos 4 períodos de tempo utilizando os novos dados do tráfego coletado no mês de maio/2010, o classificador RBF apresentou o pior índice kappa (0,32).

O resultado dos testes dos subconjuntos de 4 períodos utilizando os classificadores com os dados selecionados aleatoriamente no tempo do mês de maio/2010 mostrou que os resultados com o estudo de caso 1 foram mais satisfatórios.

Foi calculada a matriz de confusão para cada um dos classificadores. As Figuras 5.46 e 5.47 apresentam a matriz de confusão gerada para a rede neural MLP e MaxVer, respectivamente.

Matriz de Confusão - Treino MLP				
Período	Classes	anomaloac	padrao	anomaloab
P1	anomaloac	1094	88	13
	padrao	45	1333	621
	anomaloab	0	633	1235
P2	anomaloac	3182	317	-
	padrão	508	2729	-
P3	anomaloac	1220	139	640
	padrão	98	1902	0
	anomaloab	9	0	1991
P4	anomaloac	1374	635	-
	padrão	266	1760	-

Matriz de Confusão - Teste MLP				
Período	Classes	anomaloac	padrao	anomaloab
P1	anomaloac	1222	88	13
	padrao	2	1201	868
	anomaloab	0	1023	918
P2	anomaloac	3642	339	-
	padrão	867	2215	-
P3	anomaloac	1106	165	811
	padrão	103	1977	0
	anomaloab	9	0	2038
P4	anomaloac	763	1317	-
	padrão	276	1826	-

Figura 5.46 – Matriz de confusão gerada para a MLP do estudo de caso 2

Matriz de Confusão - Treino MaxVer				
Período	Classes	anomaloac	padrao	anomaloab
P1	anomaloac	1117	78	0
	padrao	130	1088	781
	anomaloab	0	293	1575
P2	anomaloac	3469	30	-
	padrão	1147	2090	-
P3	anomaloac	1874	117	8
	padrão	6	1994	0
	anomaloab	6	0	1994
P4	anomaloac	1135	874	-
	padrão	30	1996	-

Matriz de Confusão - Teste MaxVer				
Período	Classes	anomaloac	padrao	anomaloab
P1	anomaloac	1245	78	0
	padrao	12	676	1383
	anomaloab	0	819	1122
P2	anomaloac	3942	39	-
	padrão	1327	1755	-
P3	anomaloac	1896	176	10
	padrão	6	2074	0
	anomaloab	15	0	2032
P4	anomaloac	8	2072	-
	padrão	31	2071	-

Figura 5.47 – Matriz de confusão gerada para o MaxVer do estudo de caso 2

Ao observar as Figuras 5.46 e 5.47 nos 4 períodos de tempo, conclui-se que houve uma maior confusão entre os padrões, pois mais sessões rotuladas como “1” foram erroneamente classificadas como “0” e vice-versa. Também observa-se maior confusão entre os padrões “0” e “-1”. Isto permite inferir que os testes do estudo de caso 1 obteve melhor resultado quando comparado com o estudo de caso 2.

### c) Estudo de caso 3

No estudo de caso 3, o subconjunto de dados utilizados para o treinamento dos classificadores foi o mesmo do estudo de caso 1, gerado através da metodologia TRAFICIN.

Para os testes realizados com os classificadores, adicionou-se nos subconjuntos de testes, sessões de ataques do tipo DoS e Probing gerados no labRedes-DSS, como apresentados na Tabela 5.26.

Tabela 5.26 – Subconjunto de dados acrescido de sessões com ataques

Sub-conjunto	Período do tráfego	No. de sessões	Sessões com ataque
TES1	P00	1018	86
TES2	P07	7357	86
TES3	P13	14176	86
TES4	P19	1816	86

As Tabelas 5.27 e 5.28 apresentam uma comparação entre os classificadores MLP, MaxVer, SVM, J48\_Tree e RBF, destacando a taxa de acerto e o índice kappa alcançados no estudo de caso 3, na fase de teste.

Tabela 5.27 - Taxa de Acerto dos classificadores no teste do estudo de caso 3

Teste com dados de ataque					
Período	MLP	MaxVer	SVM	J48_Tree	RBF
	Taxa de Acerto	Taxa de Acerto	Taxa de Acerto	Taxa de Acerto	Taxa de Acerto
P1	94,20%	86,64%	88,21%	91,26%	76,03%
P2	98,65%	97,19%	92,84%	98,46%	82,12%
P3	96,42%	92,90%	87,30%	98,30%	82,48%
P4	92,58%	93,12%	83,14%	95,62%	60,60%
<b>Média</b>	<b>95,46%</b>	<b>92,46%</b>	<b>87,87%</b>	<b>95,91%</b>	<b>75,31%</b>

Tabela 5.28 – Índice Kappa dos classificadores no teste do estudo de caso 3

Teste com dados de ataque					
Período	MLP	MaxVer	SVM	J48_Tree	RBF
	Índice Kappa	Índice Kappa	Índice Kappa	Índice Kappa	Índice Kappa
P1	0,91	0,80	0,82	0,87	0,64
P2	0,97	0,94	0,85	0,97	0,64
P3	0,95	0,90	0,80	0,97	0,73
P4	0,84	0,86	0,65	0,91	0,07
<b>Média</b>	<b>0,92</b>	<b>0,88</b>	<b>0,78</b>	<b>0,93</b>	<b>0,52</b>

Dos cinco classificadores testados com os conjuntos de dados dos 4 períodos de tempo utilizando os novos dados do tráfego coletado no mês de maio/2010,

o classificador RBF apresentou o pior índice kappa (0,52).

O resultado dos testes dos subconjuntos de 4 períodos utilizando os classificadores com os dados de teste do mês de maio/2010 acrescidos de dados de ataques simulados em laboratório mostrou que os resultados com o estudo de caso 3 foram satisfatórios.

Foi calculada a matriz de confusão para cada um dos classificadores. A Figura 5.48 apresenta as matrizes de confusão geradas para a rede neural MLP e MaxVer.

Matriz de Confusão - Teste MLP - dados anômalos				
Periodo	Classes	anomaloac	padrao	anomaloab
P1	anomaloac	340	17	10
	padrao	29	272	2
	anomaloab	0	1	347
P2	anomaloac	3336	49	0
	padrao	99	3820	0
	anomaloab	0	53	0
P3	anomaloac	1235	143	4
	padrao	15	1612	0
	anomaloab	0	3	1595
P4	anomaloac	2284	196	0
	padrao	231	3197	0
	anomaloab	0	1	11

Matriz de Confusão - Teste MaxVer - dados anômalos				
Periodo	Classes	anomaloac	padrao	anomaloab
P1	anomaloac	277	82	8
	padrao	28	273	2
	anomaloab	1	15	332
P2	anomaloac	3204	179	2
	padrao	50	3865	4
	anomaloab	1	37	15
P13	anomaloac	1127	255	0
	padrao	47	1572	8
	anomaloab	2	15	1581
P19	anomaloac	2203	276	1
	padrao	118	3310	0
	anomaloab	0	11	1

Figura 5.48 – Matriz de Confusão do teste dos classificadores MLP e MaxVer com dados de ataque

Na fase de teste do estudo de caso 3, como mostrado na Figura 4.48, os classificadores MLP e MaxVer convergiram rapidamente (no máximo dois minutos) e os resultados também foram satisfatórios. Porém, observa-se que as sessões com ataques foram mais bem classificadas em um período do que em outro. Por exemplo, das 86 sessões com ataque inseridas nos dados do período P1, 84 delas foram corretamente classificadas pelo classificador MLP e 50 delas foram corretamente classificadas pelo classificador maxVer. Entretanto, no período P3, as mesmas 86 sessões com ataque, 51 foram corretamente detectadas pela MLP e 47 corretamente detectadas pelo MaxVer.

O motivo dos resultados diferentes na detecção de ataques DDoS e Probing nos 4 períodos analisados, mostram que em alguns períodos o conteúdo dos

atributos das sessões com ataque são mais similares aos atributos das sessões padrão.

### 5.6.2. Classificação baseada em Análise Temporal dos dados

A classificação baseada em análise temporal dos dados do tráfego de rede analisa a frequência de ocorrência de sessões com a mesma característica, ou seja, valores dos atributos, através de uma janela de leitura de 10 minutos de dados.

Nesta análise é comporta por duas etapas: análise espacial (conteúdo) da base histórica de dois meses e análise temporal das sessões classificadas como anômalas na parte 1, analisando a frequência que estas sessões anômalas similares ocorrem em uma janela de tempo de 10 minutos. A Tabela 5.29 apresenta a quantidade média de sessões lidas em uma janela de 10 minutos de acordo com o período do dia.

Tabela 5.29 – Quantidade de sessões lidas em janela de 10 minutos e 1 hora

Período do tráfego	Sessões - janela de 10 min	Sessões - janela de 1h
P00	132	811
P07	2.139	13445
P13	2846	17208
P19	71	497

Para esta etapa da metodologia foi desenvolvido um sistema baseado em regras de produção para ler o resultado da análise espacial, e armazenar as sessões anômalas similares que se repetem mais de 10 vezes na janela de tempo.

Como a análise espacial foi apresentada na Seção anterior, aqui descreve-se os resultados da análise temporal. O arquivo de dados contendo o resultado da análise espacial é o subconjunto de teste utilizado na classificação

temporal. As sessões com ataque foram inseridas no final do subconjunto dos 4 períodos de tempos, como apresenta a Tabela 5.30.

Tabela 5.30 – Sessões com ataque inseridas nos 4 períodos para análise temporal

<b>Período</b>	<b>No. sessões</b>	<b>Qtd sessões anômalas</b>	<b>Posição das sessões</b>
P00	1018	86	1019 - 1105
P07	7357	86	7358 - 7443
P13	14176	86	14177 - 14262
P19	1816	86	1817 - 1902

A Tabela 5.31 apresenta o resultado obtido da classificação temporal, armazenado em arquivo texto.

Tabela 5.31 – Sessões com ataque detectadas pelo classificador temporal

<b>Janela</b>	<b>P1</b>		<b>P2</b>		<b>P3</b>		<b>P4</b>	
	<b>pos_início</b>	<b>pos_fim</b>	<b>pos_início</b>	<b>pos_fim</b>	<b>pos_início</b>	<b>pos_fim</b>	<b>pos_início</b>	<b>pos_fim</b>
<b>1</b>	<b>1019</b>	<b>1029</b>	<b>7358</b>	<b>7368</b>	<b>14177</b>	<b>14187</b>	<b>1817</b>	<b>1827</b>
<b>2</b>	<b>1030</b>	<b>1040</b>	<b>7369</b>	<b>7379</b>	<b>14188</b>	<b>14198</b>	<b>1828</b>	<b>1828</b>
<b>3</b>	<b>1041</b>	<b>1051</b>	<b>7340</b>	<b>7350</b>	<b>14199</b>	<b>14209</b>	<b>1839</b>	<b>1949</b>
<b>4</b>	<b>1052</b>	<b>1062</b>	<b>7351</b>	<b>7361</b>	<b>14210</b>	<b>14220</b>	<b>1850</b>	<b>1860</b>
<b>5</b>	<b>1063</b>	<b>1073</b>	<b>7362</b>	<b>7372</b>	<b>14221</b>	<b>14231</b>	<b>1861</b>	<b>1871</b>
<b>6</b>	<b>1074</b>	<b>1084</b>	<b>7373</b>	<b>7383</b>	<b>14232</b>	<b>14242</b>	<b>1872</b>	<b>1882</b>
<b>7</b>	<b>1085</b>	<b>1095</b>	<b>7384</b>	<b>7394</b>	<b>14243</b>	<b>14253</b>	<b>1883</b>	<b>1893</b>

Ao observar a Tabela 5.31 nos 4 períodos, infere-se que a classificação temporal teve resultado satisfatório, pois detectou todas as sessões anômalas com características similares na janela de 10 minutos de tráfego.



## 6 CONCLUSÃO

Para executar o processo de detecção de anomalias em redes de computadores, o comportamento da rede deve ser adequadamente mapeado e constantemente atualizado no sistema de detecção de intrusos. O modelo padrão do tráfego deve ser construído sob medida a partir de informações relevantes da rede monitorada, extraídas de grandes volumes de dados.

Uma vez desenvolvido o modelo do comportamento padrão do tráfego, este é utilizado para treinamento do módulo de detecção de anomalias do SDI. Se este modelo não representar adequadamente o tráfego real, altas taxas de alarmes falsos serão geradas pelo SDI, afetando a precisão de detecção. A escolha de uma heurística inadequada para a redução dos dados pode afetar o tempo de treinamento e classificação do módulo de detecção, reduzindo o desempenho do SDI.

Sendo assim, construir um modelo compacto e significativo do comportamento do tráfego é uma atividade desafiadora. A redução de dados permite eliminar redundâncias, sem perder informações relevantes, melhorando a aprendizagem dos classificadores de maneira a minimizar as taxas de alarmes falsos.

Na elaboração da metodologia proposta, foram consideradas as dificuldades do processo de modelagem do tráfego, tais como: as sutilezas e complexidades do tráfego anômalo que pode facilmente confundir o processo; a variedade de informações existentes nos pacotes que trafegam na rede dificultando a seleção das mais significativas para análise; o fato de que o comportamento da rede é dinâmico; os diferentes serviços geram tráfego com perfis diferentes; e a quantidade volumosa de dados para análise.

Pensou-se em aplicar a metodologia no conjunto de dados fornecido pelo KDD-CUP 1999, composto por 41 atributos (STOLFO et al., 1999), para detectar anomalias existentes neste conjunto, porém verificou-se que apenas um atributo (duration) desta base são equivalentes aos aplicados neste trabalho. Visto que a caracterização do tráfego de rede proposta modela o comportamento do tráfego segundo a análise de nove atributos simultaneamente para criação do modelo do tráfego e detecção de anomalias, a aplicação da TRAF CIN sobre os dados do KDD-CUP 1999 para comparação de resultados de detecção de anomalias não seria adequada.

A metodologia TRAF CIN buscou a aplicação de técnicas de inteligência computacional capazes de tratar grandes volumes de dados espaço-temporais, abstrair informações relevantes para caracterizar o tráfego de rede padrão, no intuito de detectar anomalias em tempo satisfatório, produzindo baixa taxa de alarmes falsos.

Outras contribuições da TRAF CIN foram:

- Pré-processamento de grande volume de dados espaço-temporal;
- Geração de limiares mínimos e máximos temporais baseados no cálculo do *z-score* para cada atributo, de acordo com o dia da semana e período do dia;
- Dois métodos para reduções da base de dados analisada;
- Modelo da caracterização do comportamento padrão do tráfego de rede compacto e adequado ao tráfego real;
- Redução no tempo de treinamento dos SDI;
- Baixa taxa de alarmes falsos obtida na fase de classificação dos SDI.

Para atingir essas finalidades, das várias abordagens do TRAF CIN, algumas foram fundamentais, sendo:

- Definição dos passos da metodologia: pré-processamento de dados, modelagem e mineração de dados, clusterização do tráfego, limiarização e rotulação de dados, e detecção de anomalias.
- Criação de uma heurística para organização da base de conhecimento do tráfego histórico levando em consideração o dia da semana e horário da coleta dos pacotes;
- Aplicação da técnica de séries temporais através do cálculo de PDF e DFA aplicado aos atributos das sessões do tráfego de rede para extração de conhecimento dos dados;
- Redução 1: abordagem de mineração de dados baseada em “Mapa de Kohonen Adaptável” (MKA) para clusterização do conjunto volumoso de rede de computadores, abstraindo os vetores representantes do tráfego (e não o valor médio) dos grupos cujas sessões estão juntas devido à semelhança, dentro de incertezas (limiares empíricos, taxa de desvio e similaridade);
- Redução 2: Seleção dos maiores *clusters* de sessões de tráfego, admitindo a hipótese que os *clusters* mais populosos representam o comportamento padrão do tráfego, pois as sessões destes *clusters* são mais comuns (maior frequência de ocorrência);
- Aplicação da técnica de limiarização baseada em *z-score* nos dados de rede com média e desvio padrão históricos de dois meses de tráfego para definir os limiares mínimos e máximos de *z* para cada atributo, de cada dia de semana e período;
- Aplicação de sistema baseado em regras para rotular os dados do tráfego histórico em um de três padrões: 0 (padrão), -1 (anômalo abaixo da média), 1 (anômalo acima da média).

- Aplicação de nova clusterização dos dados rotulados com padrão (0) para equilibrar a quantidade de sessões das três classes já citadas e evitar que os classificadores saturem, ou seja, aprendem tanto uma classe que não conseguem aprender adequadamente outra;
- Aplicação de cinco diferentes classificadores baseados em aprendizagem de máquina na etapa de detecção de anomalias no tráfego de rede, tais como: rede neural MLP, RBF, SVM, Árvore de Decisão e MaxVer, para classificar as sessões do tráfego HTTP corrente;
- Classificação baseada em análise espacial de cada valor de atributo das sessões correntes para classificá-la como anômala ou padrão;
- Classificação baseada em análise temporal dos dados pela frequência de ocorrência de sessões com a mesma característica, através de uma janela de leitura de 10 minutos de dados.

Resultados satisfatórios de detecção de anomalias foram obtidos através da aplicação do TRAF CIN, pois o modelo da caracterização do tráfego padrão gerado do grande volume de dados de rede, foi compacto e representativo do comportamento das sessões nos 4 períodos de tempo. Esse modelo foi utilizado no módulo de treinamento dos SDI e validado na classificação de anomalias quando se observou taxas pequenas de alarmes falsos.

Desta forma, entende-se que a TRAF CIN contribuiu no aprimoramento das metodologias de detecção de anomalias no tráfego HTTP de rede TCP/IP, com o fornecimento de modelo compacto e significativo de representação do comportamento adequado do tráfego de rede real ao longo do tempo. Através do conjunto de procedimentos, técnicas e abordagens relativamente simples, foi possível desenvolver e testar uma nova metodologia que possa se tornar

uma referência para atividades de detecção de anomalias em ambientes de redes operacionais, por se tratar de um problema atual.

A metodologia TRAF CIN é adaptável, podendo ser adequada não somente para caracterização e análise do comportamento do tráfego HTTP, mas para mapeamento de outros tipos de tráfego de rede, tais como SMTP, DNS, DHCP. A aplicação da metodologia em cenários de Cloud Computing (Computação em Nuvem) requer o estudo de protocolos próprios e a utilização de sensores de SDI distribuídos pela rede, uma vez que o acesso a programas, serviços e arquivos neste contexto, é realizado de forma remota, através da Internet (DIAS, 2011).

Com este trabalho, pode-se concluir que técnicas estatísticas são muito úteis para o entendimento do tráfego de rede, principalmente se combinada com técnicas de outras áreas do conhecimento como análise temporal e inteligência computacional, para extrair informações relevantes dos dados analisados e classificá-las.

Durante a execução do TRAF CIN, pode-se constatar que, para experimentos envolvendo dados de tráfego de rede real, são necessários recursos tecnológicos e infra-estrutura adequada para tratamento do grande volume de dados gerado nesse tipo de sistema. O fornecimento desses recursos pela Divisão DSS e pelo projeto CAPREDES no INPE viabilizaram as experiências realizadas nesse trabalho.

Os produtos obtidos através do TRAF CIN tiveram aplicação no ambiente operacional do grupo de redes da DSS e nas atividades de pesquisa e desenvolvimento do LabRedes-DSS, incluindo contribuição na pesquisa acadêmica realizada neste Laboratório. Como contribuição para esse ambiente de rede, a TRAF CIN deixou resultados expressivos na análise, entendimento e caracterização do comportamento do tráfego padrão dessa rede através dos

conjuntos de dados de sessões do tráfego HTTP, de rede coletados durante 3 meses em 4 períodos do dia, todos os dias da semana, visto que o comportamento do tráfego varia ao longo do tempo.

Como trabalho futuro, pretende-se dar continuidade ao desenvolvimento da metodologia TRAF CIN, com as seguintes propostas:

- Estudo dos atributos do tráfego de rede, para definir a combinação destes ou de outros atributos que melhor contribuiriam para caracterizar o tráfego de rede padrão e detectar anomalias;
- Ajustes no TRAF CIN para caracterizar e analisar o comportamento do tráfego de outros protocolos de aplicação.
- Integração dos módulos do TRAF CIN e desenvolvimento de interfaces com usabilidade;
- Melhoria do TRAF CIN para atualização automática do modelo de comportamento do tráfego de rede caracterizado;
- Estudar o comportamento de outros tipos de ataques para análise temporal das sessões do tráfego de rede;
- Capturar o tráfego de rede com janela de tempo menor;
- Comparação do TRAF CIN com outras metodologias de caracterização do comportamento padrão do tráfego e detecção de anomalias de mesmo escopo;

O avanço tecnológico nas comunicações em rede vem sendo registrado e potencializado através do crescimento e evolução das redes de comunicação de dados, possibilitando o compartilhamento de informações e do próprio meio físico computacional, e ressaltando o impacto na economia, na pesquisa e na sociedade. Em contrapartida, a segurança dos ambientes de rede tornou-se

crítica e os controles de segurança precisam ser adequados às novas tecnologias empregadas.

A tecnologia muda rapidamente e cada vez mais, a criação de ambientes com ferramentas integradas para o monitoramento das redes de comunicação de dados, se faz necessária. Muito há que ser feito. Cada nova abordagem pesquisada, cada nova metodologia aplicada ou processo validado que possa ser agregado aos sistemas de detecção de intrusão constituirá um passo na consolidação do conhecimento específico para se proteger a integridade, privacidade e autenticidade dos dados na comunicação em rede.





## REFERÊNCIAS BIBLIOGRÁFICAS

ALVARADO, P.V., ROSALES, C.V., ROMAN, T., HERRERA, A.M. Detecting Anomalies in Network Traffic Using the Method of Remaining Elements. **IEEE Communications Letters**, v. 13, n. 6, June, 2009, p. 462 – 464.

APILETTI, D., BARALIS, E., CERQUITELLI, T., D'ELIA, V. Characterizing network traffic by means of the NetMine framework. In: **The International Journal of Computer and Telecommunications Networking**, v. 53, n. 6. New Yoork, April 2009, p. 774 – 789.

ARAÚJO, A., SANTANA, D., BRADÃO, R., FREITAS, A. Um Sistema de Detecção de Intrusos baseado em Mapas de Kohonen. **Revista Cientifico**, v.2, ano 2, p.12-14, 2005.

AZZINI, A., DAMIANI, E., GIANINI, G., MARRARA, S. Detection of traffic volume anomalies by evolution of negative classifiers in artificial immune systems. In: IEEE INTERNATIONAL CONFERENCE ON DIGITAL ECOSYSTEMS AND TECHNOLOGIES, 2., 2008, Phitsanulok. **Proceedings...**, IEEE, 2008. p. 270-273.

BERBECK, K., TEHRANI, S.N. Adaptative real-time anomaly detection with incremental clustering. **Journal Information Security Tech. Report**, v. 12, p. 56-67, Mar. 2007.

BOUZIDA, Y., MARGIN, C. A Framework for Detecting Anomalies in VoIP Networks. INTERNATIONAL CONFERENCE ON AVAILABILITY, RELIABILITY AND SECURETY, 3., 2008, Barcelona. **Proceedings...** Barcelona: [s.n], Mar 2008. p. 204-211.

BREIMAN, L., FRIEDMAN, J. H., OSSHEN, R. A. **Classification and regression tress**. Belmont: Cahpman&Hall, 1984.

BURBECK, K, NADJIM-T.S. Adaptive real-time anomaly detection with improved index and ability to forget. IEEE INTERNATIONAL CONFERENCE ON DISTRIBUTED COMPUTING SYSTEMS WORKSHOPS, 25., 2005, Sweden. **Proceedings...** , June 2005. p. 195-202.

BURGESS, M. Probabilistic anomaly detection in distributed computer networks. **Journal Science of Computer Programming**, vol. 60, issue 1, Amsterdam, March 2006, pp. 1-32.

CAO, L. **Support vector machines experts for time series forecasting**. Elsevier Neurocomputing, n.51, February 2002, pp. 321-339.

CASWELL, B., BEALE, J., FOSTER, J.C., POSLUNS, J. **Snort 2 – sistema de detecção de intruso open source**. Rio de Janeiro: Editora Alta Books, 2003.

CELENK, M., CONLEY, T., WILLIS, J. GRAHAM, J. Predictive Network Anomaly Detection and Visualization. **IEEE Transactions on Information Forensics and Security**, v. 5, n. 2, p. 288-299, June, 2010.

CHAVES, M.H.P.C. **Análise de estado de tráfego de redes TCP/IP para aplicação em detecção de intrusão**. Dissertação de Mestrado do Curso de Computação Aplicada. Orientador: Dr. Antonio Montes, INPE, São José dos Campos, SP, 2002.

CISCO SECURE SDI. Disponível em <http://www.cisco.com>, Acesso em: 20 jan. 2011

COMER, D.E. **Internetworking with TCP/IP**. 4a. ed. New Jersey: Prentice Hall, 2000. Vol.1: principles, protocols, and architectures.

DAINOTTI, A., PESCAPÉ, A., VENTRE, G. A packet-level characterization of network traffic. In: INTERNATIONAL WORKSHOP ON COMPUTER-AIDED MODELING, ANALYSIS AND DESIGN OF COMMUNICATION LINKS AND NETWORK, 11., 2006, Trento – Italy. **Proceedings...**Trento:[s.n], June 2006. p. 38-45.

DEPREN, O., TOPALLAR, M., ANARIM, E., CILIZ, M.K. An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks. **Expert Systems with Applications**, v. 29, n. 4, p. 713-722, Nov. 2005.

DIAS, M. **Cloud computing**: entenda como funciona este novo modelo de computação. Disponível em: <http://www.bradonetworks.com.br>. Acesso em 12 fev 2011.

ENTERASYS, Security Network. **How Dragon DSCC addresses regulatory compliance requirements**. Disponível em; <http://www.enterasys.com>, Acesso em: 20 jan 2011.

DUDA, R.O, HART, P.E. **Pattern classification and scene analysis**. 1. Ed. New York: Wiley-Interscience, USA, 1973.

FAGUNDES, L. L. **Metodologia para avaliação de sistemas de detecção de intrusão**. Trabalho de Conclusão de Curso. (Graduação em Informática) - Universidade do Vale do Rio dos Sinos. Orientador: Dr. Luciano Paschoal Gaspar, 2002.

FARID, D.Md., RAHMAN, M.Z. Anomaly Network Intrusion Detection Based on Improved Self Adaptive Bayesian Algorithm. **Journal of Computers**. v. 5, n. 1, p. 23-31, Jan. 2010,

FAUSETT, L.V. **Fundamentals of neural networks**: architectures, algorithms, and applications. New Jersey: Prentice Hill, 1994.

FIELDING, R.; GETTYS, J.; MOGUL, J.; FRYSTYK, H.; MASINTER, L.; LEACH, P.; BERNERS-LEE, T. **Hypertext transfer protocol** – HTTP/1.1. Request for Comments RFC 2616. [online]. <<http://www.rfc-editor.org/rfc/rfc2616.txt>>. Fevereiro 2010.

FONTUGNE, R., TOSHIO, H., KENSUKE, F. A visualization tool for exploring multi-scale network traffic anomalies. In: INTERNATIONAL CONFERENCE ON SYMPOSIUM ON PERFORMANCE EVALUATION OF COMPUTER & TELECOMMUNICATION SYSTEMS, (SPECT'09), 12., 2009, Piscataway, NJ. **Proceedings...** Piscataway, NJ-USA: IEEE Press, 2009. p. 274-281.

FORTUNA, C., FORTUNA, M., MOHORCIC, M. Anomaly detection in computer networks using linear SVMs. In: CONFERENCE ON DATA MINING AND DATA WAREHOUSES, 2007, Ljubljana – Slovenia. **Proceedings...** Ljubljana: [s.n], October 2007.

FOSTER, J.C.; BEALE, J.; POSLUNS, J.; CASWELL, B. **Snort 2** - sistema de detecção de intrusos.1. ed. Rio de Janeiro: Alta Books, 2003.

FURLONG, T. **Tools, data, and flow attributes for understanding network traffic without payload**. M.C.S. Thesis. Carleton University, April 2007.

GUIMARÃES, R. C., SARSFIELD J.A.C. **Estatística**. Lisboa: McGraw-Hill, 1997.

HAYKIN, S., **Redes neurais princípios e práticas**. 2. Ed. Porto Alegre: Bookman, 2001. ISBN: 8573077182.

HUBBALLI, N., BISWAS, S., NANDI, S. Fuzzy mega cluster based anomaly network intrusion detection. In: INTERNATIONAL CONFERENCE ON NETWORK AND SERVICE SECURITY 2009 (N2S'09), 2009, Paris, France, June 2009. p. 1-5.

JUN, L. An Architectural Framework for Accurate Characterization of Network Traffic. **IEEE Transactions on Parallel and Distributed System**, Piscataway – United States, 2009. p. 111-123.

KERBY, F. **The shadow konows**. Disponível em < [http://www.open-mag.com/features/Vol\\_18](http://www.open-mag.com/features/Vol_18)>, Acesso em: 20 jan. 2011

KOHONEN, T. The self-organizing map. In: **Proceedings of the IEEE**, v. 78, n. 9, 1990. p. 1464-1480.

KUNDU, S.R., PAL, S., BASU, K., DAS, S.K. An architectural framework for accurate characterization of network traffic. **IEEE Transactions on Parallel and Distributed System**, Piscataway – United States, p. 111-123, 2009.

LAKHINA, A., CROVELLA, M., DIOT, C. Characterization of network-wide anomalies in traffic flows. In: ACM SIGCOMM INTERNET MEASUREMENT CONFERENCE (IMC'04), 2004, Taormina, Italy. **Proceeding...** Taormina: [s.n], Oct. 2004. p. 201-206.

LEE, W.; STOLFO, S.J. Data mining approaches for intrusion detection. In: USENIX Security Symposium – KDDCup99, 1998, Texas. **Proceedings...** Texas: [s.n], 1998.

LI, L., LEE, G. D. DoS Attack Detection and Wavelets. **Advance Technologies in Communications e Networks**, v. 28, n. 3-4, 2005, p. 435-451.

LIU, H., LI, X. A novel traffic classification algorithm using machine learning. In: IEEE INTERNACIONAL CONFERENCE ON BROADBAND NETWORK & MULTIMEDIA TECHNOLOGY - IC-BNMT '09, 2., 2009, Trento – Italy. **Proceedings...** Trento: IEEE, Oct. 2009. p. 340-344.

LIU, J., CHEN, B. Internet traffic characterization based on active network measurement. In: INTERNATIONAL CONFERENCE ON WIRELESS COMMUNICATIONS, NETWORKING AND MOBILE COMPUTING (WICOM 2010), 6., 2010, Chengdu – China. **Proceedings...** Chengdu:[s.n], Sept., 2010. p. 1-11.

LIU, J., HUANG, J. Broadband Network Traffic Analysis and Study in Various Types of Applications. In: INTERNATIONAL CONFERENCE ON INTELLIGENT CONTROL AND INFORMATION PROCESSING, 2010, Dalian – China. **Proceedings...** Dalian: [s.n], Aug. 2010. p. 13 – 15.

LUNA I., BALLINI R., SOARES S. Técnicas de identificação de modelos lineares e não lineares de séries temporais. **Revista da SBA (Sociedade Brasileira de Automática)**, Dez., 2005.

MAHMOOD, A.N., LECKIE, C., UDAYA, P. An efficient clustering scheme to exploit hierarchical data in network traffic analysis. **Journal IEEE Transactions on Knowledge and Data Engineering**, v. 20, n. 6, NJ-USA, June 2008. p. 752-767.

MAHONEY, M.V., CHAN, P.K. An analysis of the 1999 DARPA/Lincoln Laboratory evaluation data for network anomaly detection. In: RECENT ADVANCES IN INTRUSION DETECTION (RAID'03), 6., 2003. **Lecture Notes in Computer Science**. Pittsburgh, PA: Springer Verlag, 2003. v. 2820. p. 220-237.

MCHUGH, J. Testing intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory. **ACM Transactions on Information and System Security**, 2000, p. 262-294.

MILONE, G. **Estatística geral e aplicada**. São Paulo: Pioneira Thomson Learning, 2004.

Minnesota INtrusion Detection System (MINDS). Disponível em: <<http://www.cs.umn.edu/research/MINDS/MINDS.htm>> . Acessado em 12/01/2011.

MUELDER, C., MA, K-L, BARTOLETTI, T. A visualization methodology for characterization of networks scans . In: IEEE WORKSHOP ON VISUALIZATION FOR COMPUTER SECURITY 2005 (VIZSEC 05), 2005, Minneapolis – United States. **Proceedings...** Minneapolis: IEEE, Oct. 2005, p. 29-38.

MUKKAMALA, S.; SUNG, A.H. Detecting denial of service attacks using support vector machines. department of computer science, New Mexico Tech; Source: **IEEE International Conference on Fuzzy Systems**, v.. 2, p. 1231-1236, 2003.

MÜLLER, N.C. **Representação visual de bases de documentos**. Dissertação de Mestrado em Ciência da Computação – PUC. Porto Alegre, Agosto 2002.

MURALEEDHARAN, N. Analysis of TCP flow data for traffic anomaly and scan detection. In: IEEE INTERNATIONAL CONFERENCE OF NETWORKS (ICON 2008), 16., New Delhi. **Proceedings...** New Delhi: IEEE, Dec. 2008. p. 1- 4.

NORTHCUTT, S., ZELTSER, L., WINTERS, S. [et al]. **Desvendando segurança em redes: o guia definitivo para fortificação de perímetros de redes usando firewalls, VPNs, roteadores e sistemas de detecção de invasores**. Rio de Janeiro: Campus, 2002.

PATCHA, A., PARK, J. An overview of anomaly detection techniques: Existing solutions and latest technological trends. **Elsevier Computer Network** , v. 51, n. 12, NJ-USA, Aug, p. 3448-3470, 2007.

PATCHA, A., PARK, J. An overview of anomaly detection techniques: Existing solutions and latest technological trends. **Elsevier Computer Network** , v. 51, n. 12, NJ-USA, Aug., p. 3448-3470, 2007.

PENG, C. Mosaic organization of DNA nucleotides. **Physical Review**, v. 9, n. 2, feb. 1994.

PINA, A.P.B. **Investigação e estatística**. Disponível em: <  
<http://www.saudepublica.web.pt/03-investigacao/031-epiinfoinvestiga/quadrado.htm> >. Acessado em 20/01/2011.

POSTEL, J. **Internet control message protocol**: DARPA Internet program, protocol specification. Request for Comments RFC 792. Disponível em: .  
<<http://www.rfc-editor.org/rfc/rfc792.txt>>. Acesso em: July 2002.

RUSSEL, S., NORVIG, P. **Inteligência artificial**. 2. ed. Rio de Janeiro: Campus-Elsevier, 2004.

SANTOS, A.C.F., SILVA, L.S. SILVA, J.D.S, SENE, M.P.S. Computational intelligence based method for network traffic characterization. **Expert Systems with Applications (ESWA)**, 2010a. Submitted.

SANTOS A. C. F, SILVA J. D. S., SILVA L. S., SENE, M. P. C. Análise de séries temporais para caracterizar o tráfego de rede utilizando mineração de dados por clusterização. In: WORKSHOP DOS CURSOS DE COMPUTAÇÃO APLICADA DO INPE (WORCAP 2010), 10., São José dos Campos, Brasil. **Anais...** São José dos Campos: INPE, 2010b.

SANTOS, A. C. F, SILVA, L.S., SILVA, J.D.S., ROSA, R.R, SENE, M.P.C. Application of statistics and computational intelligence techniques for analyzing network traffic. **Journal of Computational Interdisciplinary Sciences**, ISSN 1983-8409. Paper considered for publication in 22 november 2010c.

SANTOS, A. C. F.; SILVA, L.S.; SILVA, J.D.S.; ROSA, R.R. Aplicação de técnicas de análise de séries temporais em dados de tráfego de rede. In: WORKSHOP DOS CURSOS DE COMPUTAÇÃO APLICADA, 2009, São José dos Campos. **Anais...** São José dos Campos, SP: INPE, 2009.

SCHMERT, S., VOGEL, M., KÖNIG, H. Using model checking to identify erros in intrusion detection signatures. **International Journal on Software Tools for Technology Transfer (STTT)**, Germany, July, p. 89-106, 2010.

SHON, T., MOON, J. A hybrid learning approach to network anomaly detection . **Information Sciences: an International Journal**, v. 177, n. 18, New York, USA, Sept, p. 3799-3821, 2007.

SILVA, L.S. **Uma metodologia para detecção de ataques no tráfego de redes baseada em redes neurais**. Tese de Doutorado do Curso de Pós-Graduação em Computação Aplicada, orientada pelo Dr. Antonio Montes e Dr. José Demisio S. da Silva, INPE, São Jose dos Campos, SP, 2007.

SILVA, L.S.; SANTOS, A.C.F.; SILVA, J.D.S.; MONTES, A. A neural network application for attack detection in computer networks. In: INTERNATIONAL JOINT CONFERENCE IN NEURAL NETWORKS (IJCNN'2004), 2004, Budapest, Hungria. **Proceedings...** Budapest, 2004.

SILVA, L.S., SANTOS, A.C.F., SILVA, J.D.S., MONTES, A. ANNIDA: Artificial Neural Network for Intrusion Detection Application – aplicação da hamming net para detecção por assinatura. In: CONGRESSO BRASILEIRO DE REDES NEURAIIS, CBRN'2005, 7., 2005, Natal, RN. **Proceedings...** Natal, 2005a.

SILVA, L.S., SANTOS, A.C.F., SILVA, J.D.S., MONTES, A. Estudo do uso da Hamming Net para detecção de intrusão. In: SIMPÓSIO DE SEGURANÇA EM INFORMÁTICA, INSTITUTO TECNOLÓGICO DE AERONÁUTICA (ITA) - SSI'2005, 7., 2005, São José dos Campos, SP. **Anais...** São José dos Campos: ITA, 2005b.

SILVA, L.S., SANTOS, A.C.F., SILVA, J.D.S., MONTES, A. Hamming Net and LVQ neural networks for classification of computer network attacks: a comparative analysis. In: BRAZILIAN NEURAL NETWORKS SYMPOSIUM (SBRN'2006), 9., 2006, Ribeirão Preto, SP. **Anais...** Ribeirão Preto, 2006a.

SILVA, L.S., SANTOS, A.C.F., MANCILHA, T.D., SILVA, J.D.S., MONTES, A. detecting attack signatures in the real network traffic with ANNIDA. **Journal Expert Systems With Applications**. Oct., p. 2326-2333. 2006b.

SINGH, M.P., RAJAMENAKSHI, S.N. Visualization for Flow Data Based on Clustering Technique for Identifying Network Anomalies. In: IEEE SYMPOSIUM ON INDUSTRIAL ELECTRONIC AND APPLICATIONS (ISIEA 2009), 2009, Kuala Lumpur – Malaysia. **Proceedings...** Kuala Lumpur: IEEE, Oct. 2009. p. 973-978.

SOUZA, M. **Readaptação do modelo ACME! para detecção de novas técnicas de intrusão**. São José do Rio Preto: Unesp, 2002. Disponível em <<http://www.acmesecurity.org/publicacoes/monografias/folder.2005-12-27.0965990514/acme-pf-2002-marcelosousa.pdf>>. Acesso em: 02/11/2006.

SOYSAL, M., SCHMIDT, E.G. Machine learning algorithms for accurate flow-based network traffic classification: Evaluation and comparison. **Journal Performance Evaluation**, v. 67, n. 6, Amsterdam, June, p. 451-467, 2010.

STEVENS, W.R. **TCP/IP illustrated: the protocols**. Boston: Addison-Ewsley, 1994. Vol. 1, 576 pp. 32-39.

STOLFO, S.J., FAN, W., LEE W., Prodromidis, A. Cost-based Modeling and Evaluation for Data Mining With Application to Fraud and Intrusion Detection: Results from the JAM Project. In: DARPA INFORMATION SURVIVABILITY CONFERENCE AND EXPOSITION, 2000, South Carolina. **Proceedings...** South Carolina: IEEE, p. 130-144.

SWIFT, D.K., DAGLI, C.H. A study on the network traffic of connexion by boing: modeling with artificial neural networks. **Engineering Applications of Artificial**, v. 21, n. 8. Dec. p. 1113-1129, 2008.

TERDIK, G., GYIRES, T. Lévy flights and fractal modeling of internet traffic. **IEEE/ACM Transactions on Networking**, v. 17, n. 1, NJ-USA, Feb. p. 120-129, 2009.

TSANG, C.H. KWONG, S., WANG. H. Genetic-fuzzy rule mining approach and evaluation of feature selection technique for anomaly intrusion detection. **Journal Pattern Recognition**, v. 40, n. 9, NY-USA, Sep. p. 2373-2391, 2007.

VAPNIK, V.N. **Statistical learning theory** . New York: Wiley-Interscience, 1998.

VERONESE, T.B., ROSA, R.R., BOLZAN, MJA, ROCHA FERNANDES, F.C., SAWANT H.S., KARLICKY, M. Fluctuation analysis of solar radio bursts associated with geoeffective X-class flares. **Journal of Atmospheric and Solar – Terrestrial Physics**, doi: 10.1016/j.jastp.2010.09.030.

VIEIRA, P. R. **Desenvolvimento de classificadores de máxima verossimilhança e ICM para imagens SAR**. 1996. 251 p. (INPE-6124-TDI/585). Dissertação (Mestrado em Sensoriamento Remoto) - Instituto Nacional de Pesquisas Espaciais, Sao Jose dos Campos, 1996. Disponível em: <<http://urlib.net/sid.inpe.br/iris@1912/2005/07.20.06.47.40>>. Acesso em: 30 maio 2011.

WEBB,A.R. **Statistic pattern recognition**. 2. ed. [S.I.]: John WilSons, 2002.

WEI, W., SYLVAIN, G. Efficient Detection of DDoS Attacks with Important Attributos. In: INTERNATIONAL JOINT CONFERENCE ON COMPUTATIONAL SCIENCE AND OPTIMIZATION (CSO 2010), 3., 2010, Huangshan, Anhui– China. **Proceedings...** Huangshan, Anhui– China, May 2010, p. 456-460.

WIRESHARK. Disponível em < <http://www.wireshark.org>>, Acesso em 27 jan. 2011.



YU, X., DONG, X., YU, G., QIN, Y., DEJUN, Y. Data-Adaptive Clustering Analysis for Online Botnet Detection. **Neural Networks, IEEE Transactions**, v.16, p. 645-678, 2005.

XU, R., WUNSCH, D. Survey of clustering algorithms. **Neural Networks, IEEE Transactions on**, v.16, p. 645-678, 2005.

ZARPELÃO, B.B., MENDES L.S., ABRÃO, T., SAMPAIO, D.H., LIMA, M.F, PROENÇA, M.L. Detecção de Anomalias em Redes de Computadores. In: SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES (SBRT 2009), 27., 2009, Blumenau, Sc. **Anais...** Blumenau, 2009.



## ANEXO A – PROTOCOLOS TCP/IP

### A.1 Tecnologia TCP/IP

Os protocolos TCP/IP são regras de comunicação utilizadas por computadores para compartilhamento e transmissão de dados. O protocolo tem a função de especificar como um programa deve preparar os dados para que possam ser enviados ao destino de modo que o receptor venha a entender a mensagem, proporcionando, assim, a comunicação entre as máquinas.

Os protocolos são organizados em pilha de quatro níveis (camadas) conceituais, construídas sobre uma quinta camada, que corresponde ao nível físico (hardware), como ilustra a Figura A.1.1.

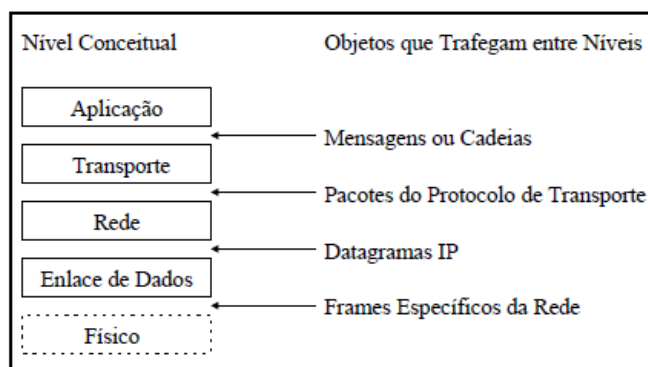


FIGURA A.1.1 – Os 4 níveis conceituais da tecnologia TCP/IP e os objetos que trafegam entre os níveis.

FONTE: adaptada de Comer (2000, p.184)

As camadas da pilha de protocolos têm funções específicas e são as seguintes:

- **Camada de Aplicação:** é a camada de nível mais alto na pilha de camadas TCP/IP, onde usuários executam aplicações, através da utilização de serviços disponíveis em uma rede TCP/IP. Para enviar ou receber dados, uma aplicação interage com os protocolos da camada de transporte. Cada aplicação escolhe o tipo de transporte, que pode ser uma cadeia contínua de bytes ou uma seqüência de mensagens individuais;
- **Camada de Transporte:** provê a comunicação entre aplicações, comumente denominadas comunicação fim-a-fim. Essa camada é responsável pelo estabelecimento e controle do fluxo de dados entre dois *hosts*. Também pode prover transporte confiável, garantindo que as informações sejam entregues sem erros e na seqüência correta. A cadeia de dados sendo transmitida é dividida em pacotes, que são passados para a camada de rede;
- **Camada de Rede:** trata da comunicação entre *hosts*. Esta camada aceita uma requisição de envio de pacotes vinda da camada de transporte, com a identificação do *host* para onde o pacote deve ser transmitido. Encapsula o pacote em um datagrama IP e preenche o cabeçalho do datagrama com os endereços lógicos de origem e destino, dentre outros dados. Utiliza um algoritmo de roteamento para determinar se o datagrama deve ser entregue diretamente, ou enviado para um gateway. Finalmente, o datagrama é passado para a interface de rede apropriada, para que este possa ser transmitido.
- **Camada de Enlace de Dados:** corresponde a camada de nível mais baixo na pilha de camadas da tecnologia TCP/IP. É responsável por aceitar os datagramas IP, encapsulá-los em frames, preencher o cabeçalho de cada frame com os endereços físicos de origem e destino,

dentre outros dados, e transmiti-los para uma rede específica. Nesta camada de enlace de dados está o *device-driver* da interface de rede, por onde é feita a comunicação com a camada física.

- **Camada Física:** corresponde ao nível de hardware, ou meio físico, que trata dos sinais eletrônicos. Esta recebe os frames da camada de enlace, convertidos em sinais eletrônicos compatíveis com o meio físico, e os conduz até a próxima interface de rede, que pode ser a do *host* ou a do gateway da rede, caso esta não pertença à rede local.

Como apresentado na Figura A.1.2, estas camadas da tecnologia TCP/IP podem ser subdivididas em vários protocolos, tais como: protocolos ARP (*Address Resolution Protocol*), RARP (*Reverse Address Resolution Protocol*), IP (*Internet Protocol*), ICMP (*Internet Control Message Protocol*), UDP (*User Datagram Protocol*), TCP (*Transmission Control Protocol*), entre outros.

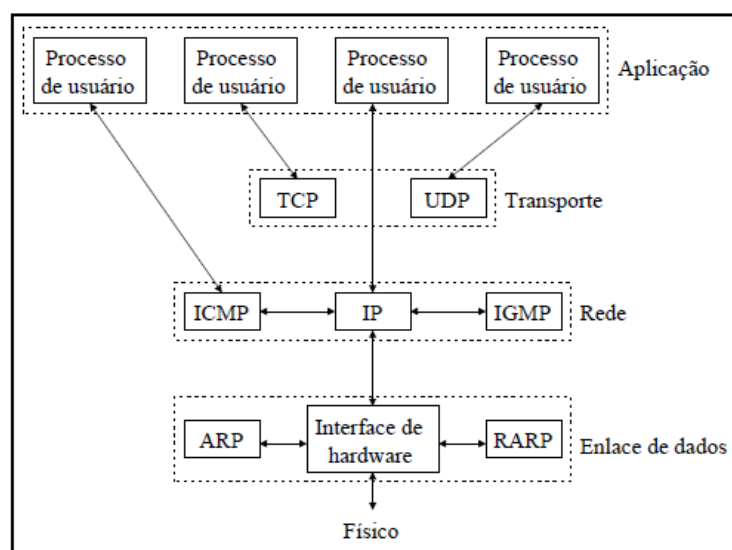


FIGURA A.1.2 – Vários protocolos nas diferentes camadas da tecnologia TCP/IP.

FONTE: adaptada de Stevens (1994, p.6).

Existem vários protocolos que atuam no ambiente de rede. As descrições dos principais protocolos necessários para o entendimento deste trabalho são apresentadas a seguir.

O protocolo IP (*Internet Protocol*) é um protocolo da camada de rede que contém informações de endereçamento e algumas informações de controle. É a base para os outros protocolos da pilha TCP/IP (POSTEL, 2002), tais como ICMP, UDP e TCP, que são transmitidos em datagramas IP, como ilustrado na Figura A.1.2.

Os datagramas IP podem ser definidos como blocos de dados transmitidos de uma determinada origem para um destino, onde origens e destinos são *hosts* identificados por endereços lógicos de tamanho fixo.

Este protocolo pode fragmentar e remontar os datagramas para que possam ser transmitidos entre redes que suportem diferentes tamanhos por bloco de dados. Em redes interconectadas, é projetado para prover as funções necessárias para entregar datagramas IP de uma origem para um destino determinado.

O protocolo IP também incorpora a função de roteamento, ou seja, determina se o datagrama deve ser entregue diretamente a seu destino, caso origem e destino pertençam à mesma rede, ou entregue ao gateway da rede contendo a origem, para o caso contrário. Se o mecanismo de roteamento decide que o datagrama deve ser enviado ao gateway, então este passa a ser responsável por enviar o datagrama para o seu destino.

A figura A.1.3 ilustra o cabeçalho de um datagrama IP, e uma breve descrição de alguns de seus campos baseada em Postel (2002) e Stevens (1994), é apresentada a seguir.

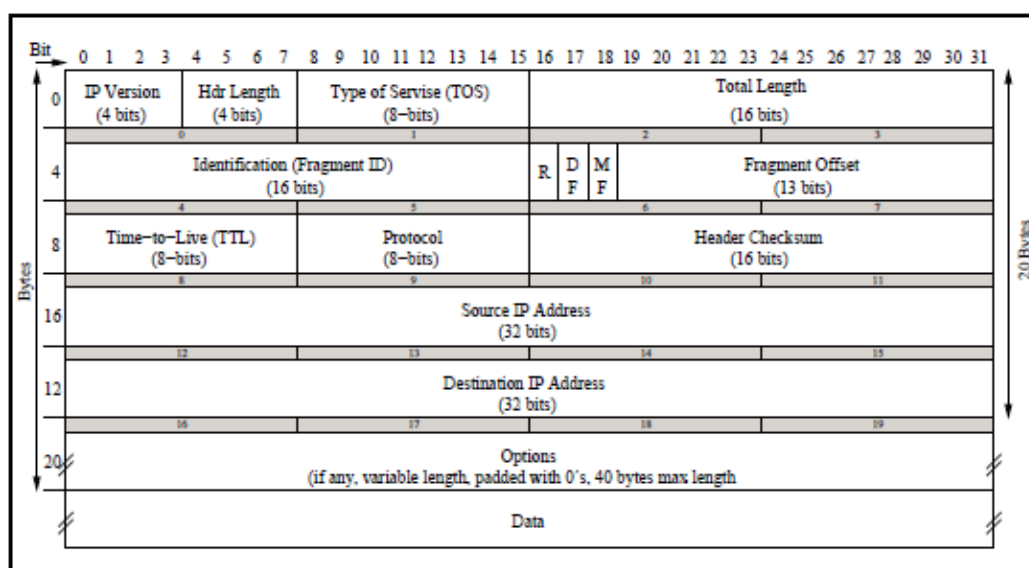


FIGURA A.1.3 – Datagrama IP, com os campos do cabeçalho IP.

FONTE: adaptada de Stevens (1994, p.34).

O campo version indica a versão do protocolo usado e o formato do datagrama, que para este caso corresponde ao IP versão 4 (também conhecido como IPv4). *Hdr length* contém o tamanho do cabeçalho IP, incluindo opções, caso estas ocorram. *Type of Service* provê uma indicação de parâmetros, relacionados à qualidade de serviço desejada, embora a maioria das implementações TCP/IP não suportem esta característica. *Total Length* contém o tamanho total do datagrama, incluindo o cabeçalho IP e os dados propriamente ditos. Identification é utilizado para identificar univocamente cada datagrama enviado por *host*. O campo *flags* é dividido em três partes: R, DF e MF, sendo R o bit reservado e que normalmente tem o valor 0, DF o bit responsável por dizer que o datagrama não deve ser fragmentado, e o MF o bit responsável por informar que o datagrama está fragmentado e existem mais fragmentos. Offset indica o posicionamento do fragmento em um dado datagrama a ser remontado. Time to Live contém um contador que indica o máximo de roteadores pelos quais um datagrama pode passar. O campo *Protocol* diz qual é o protocolo utilizado pelo próximo nível, ou seja, pelo nível

ou camada de transporte. E finalmente, os campos *Source IP Address* e *Destination IP address* contém os endereços lógicos dos *hosts* de origem e destino, respectivamente.

O tamanho do cabeçalho IP é variável, de modo que um datagrama IP sem o campo options tem 20 *bytes*. Caso opções estejam presentes no datagrama, este pode ter até 60 *bytes*. Detalhes sobre o campo options podem ser visto em Postel (2002).

O protocolo TCP (Transmission Control Protocol) é um protocolo de comunicação que provê conexões (circuitos virtuais) entre máquinas, de forma confiável, isto é, é um protocolo orientado à conexão (POSTEL, 2002).

O protocolo é confiável porque quando um *host* envia a outro, o primeiro requer o reconhecimento relativo à chegada dos dados. Além disso, os dados são seqüenciados, de modo que um número de seqüência ("*sequence number*", ilustrado na Figura A.1.3) é associado a todo pacote transmitido. Isto permite que dados sejam reordenados caso sejam recebidos fora de ordem, e descartados caso sejam duplicações de dados já recebidos (STEVENS, 1994).

O formato do cabeçalho de uma mensagem TCP contém um grande número de campos utilizados no controle da comunicação e transmissão de dados entre *hosts*, conforme ilustra a Figura A.1.4. Uma breve descrição de alguns de seus campos baseada em Postel (2002) e Stevens (1994) é apresentada a seguir, na Figura A.1.4.



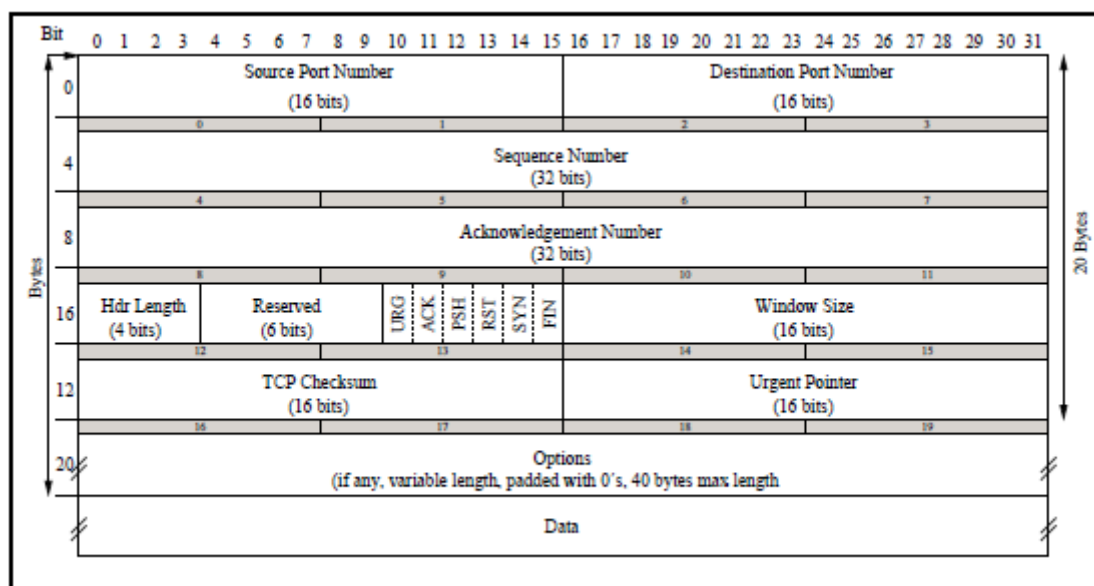


FIGURA A.1.4 – Formato da mensagem TCP.

FONTE: adaptada de Stevens (1994, p.225)

Os campos “*Source Port Number*” e “*Destination Port Number*” são utilizados na identificação das aplicações de envio e recebimento de dados, respectivamente. Quando combinados aos campos “*Source IP Address*” e “*Destination IP Address*” do cabeçalho IP identificam univocamente cada conexão. “*Sequence Number*” informa o número de seqüência do segmento, associado ao primeiro byte da cadeia de dados sendo transmitida. Para o segmento que representa o início de uma conexão TCP, este campo recebe um valor atribuído pelo sistema operacional, conhecido como *Initial Sequence Number* (ISN). Para cada segmento subsequente, este valor é incrementado do número de bytes transmitidos pelo segmento anterior. “*Acknowledgement Number*” contém o próximo número de seqüência (ou *sequence number*) esperado. O campo “*Hdr Length*” informa o tamanho do cabeçalho da mensagem TCP, que varia de 20 a 60 bytes. Se o cabeçalho tem 20 bytes, então não há ocorrência sobre o campo options. “*Window Size*” é responsável por prover o controle de fluxo implementado no protocolo TCP. Este informa

qual é o número máximo de *bytes* que podem ser recebidos por um *host*. O campo “*Urgent Pointer*” contém um valor positivo a ser adicionado ao número de seqüência do segmento, informando que os dados a partir do *byte* por este apontado têm maior prioridade.

As aplicações ou serviços de rede constituem um conjunto de protocolos encontrado no mais alto nível na pilha de protocolos TCP/IP associado aos parâmetros: número de protocolo, número da porta, identificadores, dentre outros.

Alguns dos principais serviços utilizados na Internet, bem como os respectivos números de portas são apresentados na Tabela A.1.1.

TABELA A.1.1: Exemplos de serviços de rede e portas associadas

Serviço	Porta
FTP (file transfer) [data]	20/tcp
FTP (file transfer) [control]	21/tcp
SSH (secure Shell)	22/tcp
TELNET (remote login)	23/tcp
SMTP (eletronic mail)	25/tcp
DNS (domain name system)	53/udp
FINGER (user information)	79/tcp
HTTP (the World Wide Web)	80/tcp
POP3 (post office protocol – version 3)	110/tcp
Sun RPC (remote procedure Call)	111/tcp
NTP (network time protocol)	123/udp
IMAP (Internet message access protocol)	143/tcp
SNMP (simple network management protocol)	161/udp
NFS (network file system)	2049/udp

O serviço *Web* é descrito segundo o protocolo HTTP (*Hypertext Transfer Protocol*) da camada de aplicação também conhecido como *www* (*World-Wide Web*). Este protocolo conduz a realização de várias tarefas, além da convencional transferência de hipertexto, tais como servidores de nomes e sistemas de gerenciamento de objetos distribuídos, através da extensão de seus métodos de requisição, códigos de erro e cabeçalhos (FIELDING et al., 2010).

O crescimento das redes TCP/IP e uso da Internet estimulou o desenvolvimento de técnicas e ferramentas de ataques aos serviços HTTP, por serem estes os mais amplamente utilizados no mundo.