Universidade Federal de Pernambuco - UFPE
Centro de Informática - CIN

Pós-graduação em Ciência da Computação

UMA METODOLOGIA DE CLASSIFICAÇÃO DE TRÁFEGO HTTP COM A NETRAMARK

Cezar Augusto Corado Pereira

DISSERTAÇÃO DE MESTRADO PROFISSIONAL

Recife
31 de Outubro de 2012

Universidade Federal de Pernambuco - UFPE
Centro de Informática - CIN

Cezar Augusto Corado Pereira

UMA METODOLOGIA DE CLASSIFICAÇÃO DE TRÁFEGO HTTP COM A NETRAMARK

Trabalho apresentado ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática - CIN da Universidade Federal de Pernambuco - UFPE como requisito parcial para obtenção do grau de Mestre em Ciência da Computação(Mestrado Profissional).

Orientador: *Professor Dr. Paulo André da S. Gonçalves*

Recife
31 de Outubro de 2012

A cada um que lutou, viveu e compartilhou esse sonho comigo. Em especial aos meus filhos João Pedro e Carolina.

AGRADECIMENTOS

Primeiramente ao Instituto Nokia de Tecnologia por ter proporcionado o apoio logístico e financeiro para que eu pudesse iniciar este programa de pós-graduação.

A todos os professores do Centro de Informática da UFPE pela dedicação e por tornar este centro uma referência no Brasil e no mundo.

Em especial ao Prof. Dr. Paulo André da S. Gonçalves por mostrar o caminho da verdadeira contribuição científica. Suas palavras, ensinamentos e generosidade durante o processo de orientação foram de extrema importância. Obrigado por tornar este mestrado uma experiência enriquecedora e única em minha vida acadêmica.

Aos meus amigos verdadeiros de ontem e sempre, Julio Coelho, Geovani Neto, Luis Santos e Rodrigo Botelho. Estaremos sempre juntos.

Aos meus amigos de hoje e sempre, Gleydson Vilanova e José Wallace Junior. Obrigado por agregarem suas experiências e conhecimento durante essa longa caminhada. Vocês são pessoas iluminadas e estaremos sempre juntos.

Ao amigo Laerte Rodrigues e sua família. Obrigado pela acolhida e pelo companheirismo. Serei eternamente grato por isso. Estaremos sempre juntos.

À minha mãe Neide e ao meu tio e padrinho Joaquim Corado. Dar orgulho e felicidade à vocês foi uma das grandes motivações durante esta caminhada.

Ao meu irmão Àlvaro por estender a mão e conceder suas sábias palavras sempre que precisei. Obrigado por acreditar em mim.

Ao meu pai Silvio e também aos meus irmãos Luciana e Silvio Junior.

Aos meus filhos João Pedro e Carolina. Todo esse esforço foi por vocês e posso afirmar que valeu a pena. Espero poder ser um pai melhor a cada dia e tornar o mundo de vocês mais feliz.

À minha noiva Ana Paula, amiga e companheira de todos os momentos. Obrigado por estar incondicionalmente ao meu lado em todos os momentos, não tenho palavras para agradecer por isso.

E por fim ao ser supremo e força maior da minha vida, Deus. Obrigado por tornar tudo isso possível e me ensinar que toda experiência em nossas vidas tem um propósito bom. Obrigado por estar aqui hoje e sempre.

Ninguém ignora tudo. Ninguém sabe tudo.
Todos nós sabemos alguma coisa. Todos nós ignoramos alguma coisa.
Por isso aprendemos sempre.

—PAULO FREIRE (A importância do ato de ler, 1989)

RESUMO

Em [Lee et al., 2011] é apresentada uma ferramenta denominada *NeTraMark*, a qual implementa dez classificadores do estado da arte para a classificação de tráfego de rede. A *NeTraMark* inclui adicionalmente a proposta de um classificador, chamado *Weighted Vote*, que combina o resultado desses dez classificadores em um processo de votação baseado em pesos configuráveis pelo operador da ferramenta. O resultado de *Weighted Vote* é então obtido a partir de duas regras: Votação da maioria baseada nos pesos e o classificador com maior acurácia em relação ao *ground truth*. No entanto, o resultado desse classificador combinado nem sempre é o mais acurado, pois uma atribuição adequada de pesos depende do conhecimento prévio do administrador sobre o que trafega realmente na rede, o que na maioria das vezes inexiste. Este trabalho propõe uma metodologia para uso da *NeTraMark* que descarta a necessidade de conhecimento prévio do tráfego da rede para uma atribuição adequada de pesos. A metodologia inclui a proposta de um classificador combinado para tráfego HTTP. Esse classificador, denominado FBC (*F-Measure Based Classifier*), utiliza pesos fixos obtidos a partir de diversos experimentos com cenários de tráfego camuflado na porta 80 do protocolo TCP (*Transmission Control Protocol*). O classificador proposto identifica tráfego HTTP sem a necessidade de tentativas baseadas em conhecimento prévio para o melhor ajustes dos pesos. A avaliação de acurácia do classificador proposto mostra que obteve-se resultado igual ao melhor classificador da *NeTraMark* com 96,60% de acurácia. Obteve-se ainda melhoria de 26,89% em comparação a acurácia de *Weighted Vote*. Adicionalmente, ao analisar tráfego de rede desconhecido também foi capaz de sugerir um percentual 3,11% maior de tráfego HTTP que o melhor classificador da *NeTraMark* e 8,35% maior que o classificador *Weighted Vote*.

Palavras-chave: Classificação de Tráfego de Rede, NeTraMark, HTTP, Tráfego Camuflado.

ABSTRACT

In [Lee et al., 2011] is presented a tool called *NeTraMark*, which implements ten state of art classifiers for network traffic classification. *NeTraMark* further includes the proposal of a new classifier, namely *Weighted Vote*, which combines the results of those ten classifiers in a voting process based on configurable weights. The result of *Weighted Vote* is then obtained from two rules: Majority votes based on the weights and the best accuracy classifier related to ground truth. However, the result of this classifier is not always more accurate, because a proper assignment of weights depends on prior knowledge of the administrator on what actually goes through the network, which in most cases is nonexistent. This thesis proposes a methodology to use *NeTraMark* that eliminates the need for prior knowledge of network traffic to get an appropriate allocation of weights. The methodology includes the proposal of a combined classifier for HTTP traffic. This classifier, namely FBC (*F-Measure Based Classifier*), uses fixed weights obtained from several experiments with camouflaged traffic scenarios on port 80 of TCP (*Transmission Control Protocol*) protocol. The proposed classifier identifies HTTP traffic without the need of attempts based on prior knowledge for the best fit weights. The assessment of the proposed classifier accuracy shows that was obtained equal results than the best *NeTraMark* classifier with accuracy 96,60%. Was obtained further improvements of 26,89% in comparison of *Weighted Vote* accuracy. In addition, when analyzing unknown network traffic was also able to suggest a greater percentage of 3,11% of HTTP traffic than the best *NeTraMark* classifier and 8,35% greater than *Weighted Vote* classifier.

Keywords: Network Traffic Classification, NeTraMark, HTTP, Camouflaged Traffic.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

GLOSSÁRIO

Acurácia	Grandeza que indica grau de corretude e perfeição, 35
ADSL	Asymmetric Digital Subscriber Line, 34
APC	Acurácia Ponderada do Classificador, 50
Benchmark	Um ponto de referência para uma medição, 17
BLINC	Blind Classification, classificação no escuro, 20
Downstream	Dados enviados de um provedor de serviços de rede à um cliente, 34
DPI	Deep Packet Inspection, inspeção profunda de pacotes, 14
FBC	F-Measure Based Classifier, classificador baseado em F-Measure, 49
FN	Falsos Negativos, 23
FP	Falsos Positivos, 23
HTTP	Hypertext Transfer Protocol, protocolo de transferência de hipertexto, 2, 5
IANA	Internet Assigned Numbers Authority, 7
IP	Internet Protocol, protocolo de internet, 5
ISO	International Standardization Organization, organização internacional para padronização, 5
ML	Machine Learning, aprendizagem de máquina, 14

OSI	Open Systems Interconnection, interconexão de sistemas abertos, 5
P2P	Peer to peer, comunicação ponto a ponto, 11
Precision	Exatidão ou qualidade, 24
Recall	Completude ou quantidade, 24
RFC	Request for Comments, 8
SSH	Secure Shell, protocolo criptográfico de rede para comunicação segura, 11
TCP	Transmission Control Protocol, protocolo de controle de transmissão, 6
Testbed	Plataforma para experimentação e testes, 33
TP	Verdadeiros Positivos, 23
UDP	User Datagram Protocol, protocolo de datagramas de usuário, 6
Upstream	Dados enviados de um cliente à um provedor de serviços de rede, 34
WWW	World Wide Web, rede mundial de computadores ou Internet, 2, 8

CAPÍTULO 1

INTRODUÇÃO

A Internet vem se tornando alvo de ameaças digitais cada vez mais sofisticadas por parte de especialistas em crimes na rede e usuários maliciosos. Isso se deve em grande parte pelo surgimento de vírus, das *botnets*[1], e por necessidades específicas de usuários em utilizar recursos de Internet não autorizados nas organizações. Como consequência, existe um crescente esforço das organizações para a identificação do que se trafega em suas redes e assim encontrar potenciais ameaças. Como parte deste esforço, diversas abordagens foram desenvolvidas para classificação de tráfego de Internet. Pode-se que dizer que atualmente as pesquisas nesta área atingiram maturidade. No entanto, existem lacunas abertas e questões não resolvidas desafiando os pesquisadores e motivando novas pesquisas.

1.1 MOTIVAÇÃO

A Internet foi disseminada ao longo dos anos sempre sustentada pela necessidade de inclusão digital, do acesso à informação e da liberdade de uso, integrando usuários com interesses comuns e fornecendo ferramentas de valor agregado à sociedade. De fato, essas características constituem os alicerces essenciais que sustentaram a rápida disseminação da rede, a integração das comunidades e a grande diversidade de serviços oferecidos através da rede mundial.

Por outro lado, com a evolução tecnológica e o desenvolvimento econômico experimentado pelo segmento de serviços na Internet, segurança é um item constante na agenda de analistas, administradores de rede e gestores de grandes companhias. Este ambiente integrado e livre é também alvo de ameaças cada vez mais sofisticadas por parte de *hackers*, *spammers*, *botnets* e especialistas em crimes na rede.

Neste contexto, o protocolo HTTP (*Hypertext Transport Protocol*) apresenta-se como principal protocolo na camada de aplicação, responsável pelo tráfego de conteúdo através da Internet. Inicialmente utilizado para o acesso de páginas na *World Wide Web* (WWW)

[1]Uma coleção de computadores conectados através da Internet, controlados por uma parte maliciosa para uso em negação de serviços, envio de *spams*, roubo de números seriais de software e informações financeiras, dentre outras finalidades.

o protocolo HTTP hoje é utilizado por um grande conjunto de aplicações de uso conhecido. Essas aplicações vão de TVs inteligentes, aplicações peer-to-peer (P2P), áudio e vídeo *streaming*. Por ser um protocolo amplamente conhecido e utilizado, o protocolo HTTP é considerado na grande maioria dos ambientes uma porta confiável de comunicação na Internet.

Exatamente por esta razão uma grande quantidade de ameaças digitais e os mais variados tipos de tráfego indesejado utilizam o protocolo HTTP. Com as técnicas atuais, os responsáveis por essas pragas virtuais usam este protocolo como um canal aberto para acesso às redes corporativas e para a disseminação de conteúdo malicioso.

Para resolver o problema de conteúdo não permitido em seus ambientes inicialmente os administradores de rede empreendiam esforços no bloqueio de aplicações utilizando filtragem baseada em portas de comunicação. Através de técnicas de camuflagem de tráfego tornou-se possível burlar os dispositivos de segurança existentes nas empresas utilizando o protocolo HTTP. Como exemplo, nos mecanismos de tunelamento o tráfego de dados é encriptado e portanto qualquer técnica baseada em assinaturas ou baseada em portas torna-se ineficiente [Dusi et al., 2008].

As pesquisas sobre classificação e identificação do tráfego de dados na Internet evoluíram através da criação de classificadores capazes de identificar o tráfego com alta confiabilidade [Choi, 2006], [Moore and Zuev, 2005], [Venosa et al., 2008] e [Peng et al., 2009]. A cada conjunto de aplicações desenvolvidas para trafegar na rede novas pesquisas são iniciadas com o objetivo de manter os ambientes cada vez mais seguros [Callado et al., 2009].

Entender o uso dos classificadores atuais, com o objetivo de estudar o comportamento das redes e do tráfego de dados, torna-se essencial na criação de políticas de uso adequadas no dimensionamento e no uso eficiente dos recursos da Internet.

A pesquisa nesta área ainda apresenta várias lacunas que precisam ser preenchidas apresentando soluções de análise de tráfego com menor custo computacional e com maior acurácia.

Num primeiro momento podemos visualizar que identificar o melhor nível de detalhamento dos dados de tráfego ainda é uma oportunidade de desenvolvimento de pesquisa relevante, o problema é encontrar a quantidade mínima de dados necessária para a classificação das aplicações. Criar uma abordagem que possa alcançar os níveis de acurácia e completude para a grande quantidade de aplicações e suas características de complexidade, também é uma questão em aberto.

Reduzir os níveis de tráfego não classificado é um desafio importante pois com as técnicas existentes de classificação ainda existe uma quantidade de tráfego que não é

possível processar e classificar. Isso deve-se ao fato de tratar-se de uma nova aplicação ou por estar encriptada. Mapear automaticamente novas aplicações na rede torna-se um desafio gerado pela diversidade e pelo grau de dinamismo em relação ao domínio de aplicações de Internet.

O desenvolvimento de métricas claras para mapear o uso da rede é importante, assim como, mapear o uso de perfis de usuário e seu comportamento de uso da Internet. Em última instância, definir o local ideal na rede onde realizar medições e processos de classificação podem afetar consideravelmente os resultados das pesquisas [Callado et al., 2009].

A classificação e identificação do tráfego de Internet, especialmente o tráfego que se apresenta camuflado através do protocolo HTTP, pode representar um grande diferencial para os operadores e administradores da rede. O avanço nestas pesquisas pode auxiliar na identificação de conteúdo indesejado (e.g. *spam*, *spim*, trojans, vírus e *worms*), na redução de vazamento de conteúdo confidencial, em sistemas de contabilização de serviços e no melhor uso dos recursos da rede dentro dos ambientes corporativos.

Dentre as abordagens desenvolvidas para classificação de tráfego , [Lee et al., 2011] implementou uma ferramenta , denominada *NeTraMark*, que disponibiliza 10 (dez) classificadores do estado da arte. Implementa ainda um novo classificador, denominado *Weighted Vote*, que tem seu resultado a partir da combinação do resultado dos outros classificadores. Para isso são atribuídos pesos a estes classificadores de acordo com o conhecimento do operador. Há necessidade também da definição de um *ground truth*, escolhido dentre os classificadores disponíveis.

Ao utilizar a ferramenta para classificar tráfego HTTP, em especial em situação de tráfego camuflado na porta TCP 80, o operador depara-se com algumas questões relevantes. A primeira trata da atribuição adequada dos pesos. A segunda, em definir um classificador *ground truth* que seja o mais acurado para o tráfego, de modo a estabelecer um parâmetro de comparação confiável. Ambas as questões só podem ser resolvidas completamente se o administrador possuir conhecimento sobre este tráfego. Sem o conhecimento prévio, qualquer atribuição de pesos ou definição de *ground truth* tende a ser inadequada. Desta maneira o processo de classificação torna-se empírico e o resultado de *Weighted Vote* fica comprometido.

1.2 OBJETIVOS

O objetivo geral deste trabalho é propor uma metodologia para classificar tráfego de Internet, especificamente para o protocolo HTTP, que melhore o classificador *Weighted Vote* na ferramenta *NeTraMark* desenvolvido por [Lee et al., 2011]. Espera-se também

estabelecer um ganho em resultados de classificação com o objetivo de evitar o uso de aplicações não permitidas ou proliferação de conteúdo indesejado trafegando através do protocolo HTTP. Adicionalmente podem ser listados os seguintes objetivos específicos:

- Estudar os classificadores implementados na *NeTraMark* utilizando bases geradas em ambiente controlado. Estas bases devem conter tráfego HTTP real e simular situações de tráfego camuflado pela porta TCP 80;
- Analisar dos resultados de classificação obtidos no ambiente de simulação, sobretudo considerando a acurácia dos classificadores para tráfego HTTP;
- Determinar a configuração de um novo classificador, combinando classificadores acurados a pesos genéricos com base na análise realizada;
- Avaliar o desempenho do classificador proposto em relação aos classificadores implementados na NeTraMark.

1.3 ORGANIZAÇÃO

O conteúdo deste documento está organizado como segue: O Capítulo 2 apresenta conceitos necessários para a compreensão deste trabalho. O mesmo capítulo define o problema de classificação de tráfego de Internet e justifica o problema de camuflagem de tráfego na porta TCP 80. Em seguida, o Capítulo 3 descreve a ferramenta para classificação de tráfego *NeTraMark*. No Capítulo 4 é apresentada a metodologia para uso da *NeTraMark* baseada na análise de resultados de classificação da *NeTraMark* para tráfego HTTP real e tráfego não HTTP camuflado pela porta TCP 80. A partir deste estudo é proposto um novo classificador baseado em pesos genéricos chamado FBC (*F-Measure Based Classifier*). Neste capitulo os resultados do novo classificador são analisados e comparados em relação aos classificadores disponíveis na *NeTraMark*. Por fim, no Capítulo 5 são apresentadas as conclusões obtidas por este trabalho assim como propostas para trabalhos futuros.

CAPÍTULO 2

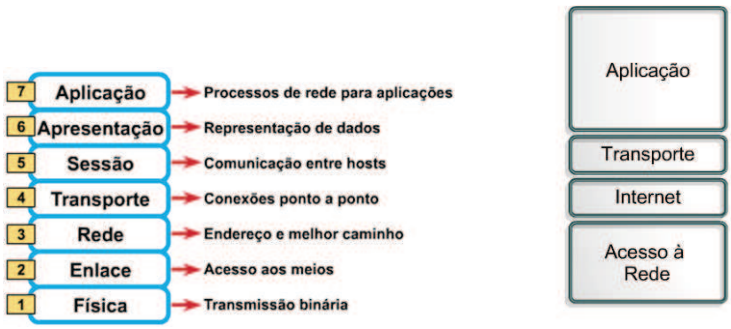
CONCEITOS GERAIS

Este capítulo faz uma contextualização sobre classificação de tráfego de Internet. São definidos conceitos necessário para o desenvolvimento e compreensão deste trabalho. Tratando-se de classificação de tráfego diversas abordagens já foram desenvolvidas. No entanto, em virtude da complexidade do problema faz-se necessário avançar com novas pesquisas nesta área. Especialmente, em relação a técnicas difundidas de camuflagem de tráfego em portas de aplicações conhecidas, como HTTP, a classificação de tráfego torna-se relevante para os administradores de rede.

2.1 REDES DE COMPUTADORES E PROTOCOLOS

Quando as redes de computadores surgiram, as soluções de conexão eram geralmente proprietárias dificultando a interconexão entre diferentes sistemas de computadores. Com isso, houve a necessidade da criação de modelos de referência com o objetivo principal de estabelecer uma arquitetura de referência para que fosse possível conectar várias redes da mesma maneira. Pode-se conceituar os modelos de referência como descrições em camadas de arquiteturas de rede [Tanenbaum, 2003].

Existem dois modelos de referência principais para arquiteturas de rede, o modelo desenvolvido pela ISO (*International Standardization Organization*) chamado OSI (*Open Systems Interconnection*) , e o modelo de referência TCP/IP, desenvolvido pelo Departamento de Defesa dos Estados Unidos. Ambos os modelos são definidos em camadas. Cada camada desempenha uma função específica na comunicação entre sistemas de computadores. O modelo TCP/IP tornou-se o padrão no qual a Internet se desenvolveu. A Figura 2.1 mostra uma comparação entre as camadas do modelo OSI e TCP/IP.



(a) OSI. (b) TCP/IP.

Figura 2.1: Modelos de referência para arquitetura de redes [Perez, 2012][1].

A camada de rede é responsável por controlar a operação da rede de um modo geral. Suas principais funções tratam do roteamento das informações entre origem e destino, o controle de congestionamento e a contabilização de informações trafegadas para fins de tarifação [Peterson and Davie, 2004]. Esta camada lida especificamente com o protocolo IP (*Internet Protocol*) e seu endereçamento. Endereços IP representam uma forma genérica de identificação de um dispositivo na rede.

A camada de transporte é a camada responsável pela transferência dos dados entre a máquina de origem e a máquina de destino, independente do tipo, topologia ou configuração das redes físicas existentes entre elas. Os protocolos mais populares da camada de transporte, dentro do modelo de referência TCP/IP, são os protocolos UDP (*User Datagram Protocol*), sem conexão e garantia na entrega dos pacotes, e o protocolo TCP (*Transmission Control Protocol*) que é orientado à conexão e com garantia na entrega dos pacotes.

Na camada de aplicação ocorre a interação entre o computador e o usuário. A camada de aplicação é responsável por identificar e estabelecer a disponibilidade da aplicação na máquina destinatária e disponibilizar os recursos para que tal comunicação aconteça. Trataremos neste trabalho do protocolo HTTP da camada de aplicação.

2.1.1 Protocolo TCP

Graças ao protocolo TCP as aplicações podem estabelecer a comunicação de forma confiável utilizando as informações das camadas inferiores do modelo de referência TCP/IP. Isto significa que dispositivos de roteamento de pacotes IP têm como único papel o encaminhamento dos dados sob a forma de datagramas ou pacotes. O controle dos dados é

[1]Disponível em http://diaadiaemti.blogspot.com.br/2012/01/comparando-o-modelo-osi-com-o-modelo.html . Acessado em 27/09/2012.

realizado pelo protocolo TCP.

Durante uma comunicação através do protocolo TCP, duas máquinas devem estabelecer uma conexão. A máquina emissora (a que pede a conexão) chama-se cliente, enquanto a máquina receptora se chama servidor. Diz-se então que estamos num ambiente Cliente-Servidor. As máquinas em tal ambiente comunicam-se em modo ligado, ou seja, a comunicação é feita nos dois sentidos conforme demonstra a Figura 2.2.

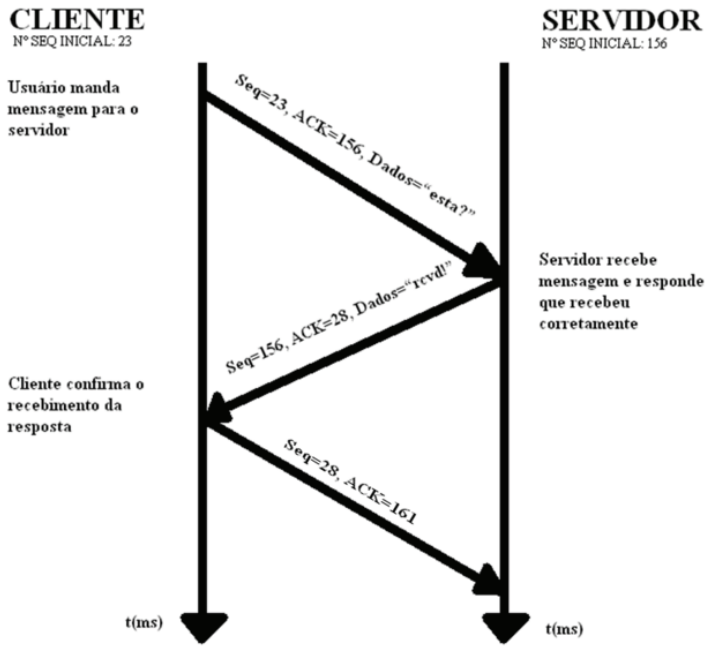


Figura 2.2: Comunicação através do protocolo TCP [Wik, 2012][2].

Para permitir o bom desenrolar da comunicação e de todos os controles que a acompanham os dados de comunicação passam por um processo de encapsulamento. Neste processo, o protocolo TCP junta ao pacote de dados um cabeçalho contendo informações que permitem a sincronização da transmissão, assegurando assim a sua recepção. Na Figura 2.3 é possível verificar a representação deste cabeçalho ou segmento TCP.

[2]Disponível em http://pt.wikiversity.org/wiki/Introducao_Redes_de_Computadores/Protocolo_TCP. Acessado em 27/09/2012.

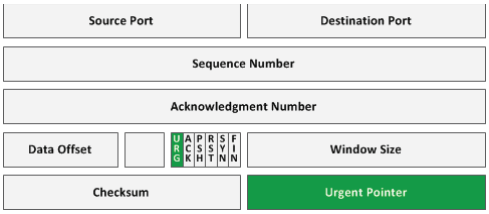


Figura 2.3: Cabeçalho TCP [Stretch, 2011][3].

Um segmento TCP consiste em um cabeçalho fixo com pelo menos 20(vinte) bytes, seguido por zero ou mais bytes de dados [Tanenbaum, 2003]. Especificamente os campos *Source Port* e *Destination Port* identificam os pontos terminais, ou portas locais de conexão. Os números de portas TCP são definidas pela IANA[4], mas vale ressaltar que cada *host* pode utilizar um número de porta da maneira que desejar. Isso possibilita que aplicações utilizem portas conhecidas de outras aplicações. Combinando a porta TCP e o endereço IP de origem à porta TCP e o endereço IP de destino torna-se possível identificar uma conexão entre dois *hosts*.

2.1.2 Protocolo HTTP

De acordo com a RFC 2616[5], o protocolo HTTP (*Hypertext Transfer Protocol*) é um protocolo da camada de aplicação, para distribuição, colaboração e sistemas de informação *hipermídia*. O HTTP é um dos protocolos de transferência mais utilizados pela Internet e vem sendo utilizado pela iniciativa global da *World Wide Web* (WWW) desde 1990.

O protocolo HTTP especifica as mensagens que os clientes podem enviar aos servidores e que respostas eles receberão, estas mensagens são conhecidas como hipertexto. Tais mensagens representam um conjunto de objetos multi-lineares, construindo uma rede através de ligações lógicas (conhecidos como *hyperlinks*) entre dois *hosts*.

Figura 2.4: Exemplo de troca de mensagens por HTTP [Hig, 2011][6].

[3]Disponível em http://packetlife.net/blog/2011/mar/2/tcp-flags-psh-and-urg/. Acessado em 27/09/2012.

[4]Internet Assigned Numbers Authority (IANA). http://www.iana. org/assignments/port-numbers.

[5]Disponível em http://www.w3.org/Protocols/rfc2616/rfc2616.html .

O modo habitual de um navegador de Internet (e.g. *Internet Explorer*, *Chrome*) entrar em contato com um servidor é estabelecer uma conexão TCP para a porta 80 da máquina servidora, embora esse procedimento não seja exigido formalmente. A vantagem de se usar o TCP é que nem os navegadores nem os servidores têm de se preocupar com mensagens perdidas, mensagens duplicadas, mensagens longas ou confirmações [Tanenbaum, 2003]. Todos esses questões são tratadas pela protocolo TCP.

Por ser um protocolo bastante difundido no uso de aplicações Internet, o HTTP torna-se alvo de ameaças de segurança em ambientes de rede através de técnicas de camuflagem. A seção seguinte detalha algumas destas técnicas.

2.1.3 Pacotes de rede

Em uma rede de computadores , um pacote de rede pode ser definido como uma estrutura unitária de transmissão de dados, ou uma sequência de dados transmitida por uma rede ou linha de comunicação que utilize a comutação de pacotes.

Conforme pode-se verificar na Figura 2.5 os pacotes de rede contêm várias camadas de informação visando auxiliar na transmissão dos dados entre dois dispositivos de rede. Cada camada de informação contém *bytes* organizados em uma ordem pré-determinada, especificando parâmetros específicos [Kurose, 2010].

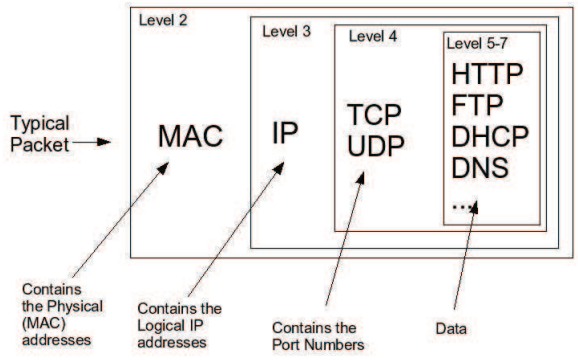


Figura 2.5: Pacote de rede [Dimitrios, 2011][7].

No contexto deste trabalho ao referenciar-se o termo pacote, será considerada na verdade a estrutura unitária que encapsula as camadas de rede, de transporte e de aplicação

[6]Disponível em http://www.highteck.net/EN/Application/Application_Layer_Functionality_and_Protocols.html . Acessado em 27/09/2012.

[7]Disponível em http://tournasdimitrios1.wordpress.com/2011/01/19/the-basics-of-network-packets/ . Acessado em 27/09/2012.

de acordo com o detalhamento abaixo. A esta estrutura pode-se denominar também pacote TCP/IP.

Conforme mostrado na Figura 2.6 dentro deste pacote existem *headers*(cabeçalhos) e o conteúdo. Dentre outras informações os *headers* da camada de rede (protocolo IP) incluem o endereço IP de origem e de destino, enquanto os *headers* da camada de transporte (protocolo TCP) incluem o número de porta TCP de origem e de destino. Ao conteúdo com os dados denomina-se *payload* dos pacotes. O *payload* contém os dados ou a informação útil da aplicação a ser transmitida. Pode-se dizer que o IP se encarrega da entrega dos pacotes, enquanto o TCP se encarrega da verificação de erros, numeração de portas e entrega dos pacotes sem falhas entre os *hosts*.

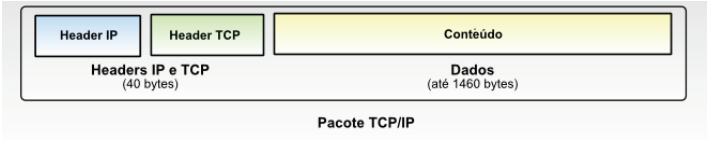


Figura 2.6: Pacote TCP/IP [Vendetta, 2010][8].

Em termos de tamanho, neste pacote existem ao menos 20 *bytes* para os *headers* do protocolo TCP e mais pelo menos 20 *bytes* para os *headers* do protocolo IP, totalizando ao menos 40 *bytes* de *headers* por pacote. Desta forma, existem 1460 *bytes* para dados em um pacote de 1500 *bytes* e 536 *bytes* de dados em um pacote de 576 *bytes*.

A Figura 2.7 mostra uma topologia para uma rede baseada na transmissão de pacotes. Nesta topologia os computadores *Client 1*, *Client 2* e *Client 3* trocam pacotes sucessivos de informação com a máquina servidor.

[8]Disponível em http://codeidol.com/csharp/csharp-network/IP-Programming-Basics/Analyzing-Network-Packets/ . Acessado em 27/09/2012.

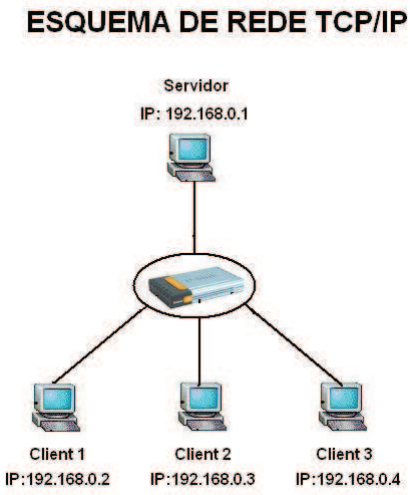


Figura 2.7: Exemplo de rede [UOL, 2007][9].

A este conjunto de pacotes sucessivos denomina-se fluxo de pacotes. Assume-se para este estudo que um fluxo de pacotes é uma sequência de pacotes com o mesmo endereço de origem, endereço de destino, porta de origem, porta de destino e número de protocolo. Pode ser representado por uma lista ordenada ou tupla deste cinco elementos. Um fluxo de pacotes pode ser determinado ou capturado em intervalos de tempo sucessivos.

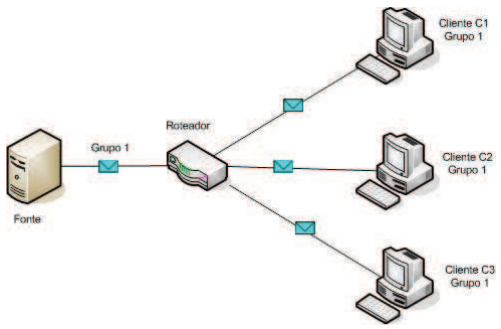


Figura 2.8: Exemplo de fluxo de pacotes [Cyb, 2009][10].

2.2 CAMUFLAGEM DE TRÁFEGO

O termo camuflagem[11] consiste na arte de colocar ou colocar-se sob aparência de outra coisa. Em ambientes de rede, pode-se considerar que a camuflagem de tráfego consiste em

[9]Disponível em http://adrenaline.uol.com.br/forum/internet-redes/161704-esquema-rede-tcp-ip-ajuda.html . Acessado em 27/09/2012.

[10]Disponível em http://cybergav.in/2009/09/25/weblogic-ip-multicast-a-primer/ . Acessado em 27/09/2012.

[11]http://www.dicionarioaurelio.com

colocar os pacotes de uma dada aplicação com as características de outra aplicação. Dentre as técnicas de camuflagem conhecidas, podem-se destacar as técnicas de tunelamento (com uso de criptografia) e o uso de portas conhecidas de aplicações alheias.

Em redes de computadores dentro das organizações, os sistemas de segurança são comumente baseados em aplicações de implementações de *Firewall*[12] e IDS (*Intrusion detection systems*)[13]. Usuários mal intencionados ou especialistas de crimes na Internet podem utilizar diversas técnicas de camuflagem de tráfego para burlar os sistemas de segurança da rede.

A finalidade deste tipo de iniciativa pode incluir obtenção de vantagens em largura de banda, tráfego de aplicações não permitidas na rede e acesso a *hosts* e serviços não autorizados [Peng et al., 2009]. Como exemplo, algumas aplicações como software P2P (peer to peer) podem camuflar o tráfego utilizando portas conhecidas como a porta TCP 80 do protocolo HTTP, então, alguns sistemas de segurança não conseguirão distinguir aplicações não permitidas de aplicações conhecidas [Karagiannis et al., 2004].

O tunelamento consiste no encapsulamento de determinado tráfego, utilizando um protocolo normalmente permitido na rede. O túnel geramente é encriptado e utiliza portas de serviços disponíveis para acesso a serviços não permitidos. O tunelamento pode ser feito a partir de um computador externo ou um computador interno da rede. Dentre as abordagens de tunelamento disponíveis destacam-se aquelas que utilizam o protocolo SSH (*Secure Shell*) e o protocolo HTTP.

O tunelamento a partir do protocolo SSH, conforme demonstrado na Figura 2.9, se caracteriza por duas máquinas ligadas ao mesmo servidor SSH. Este servidor faz apenas o redirecionamento das requisições do computador que está controlado por um *firewall*.

[12]Dispositivo na rede, associados a redes TCP/IP, que tem por objetivo aplicar uma política de segurança a um determinado ponto da rede.

[13]Refere-se a meios técnicos de descobrir em uma rede quando esta está tendo acessos não autorizados que podem indicar a ação de um *cracker* ou até mesmo de funcionários mal intencionados.

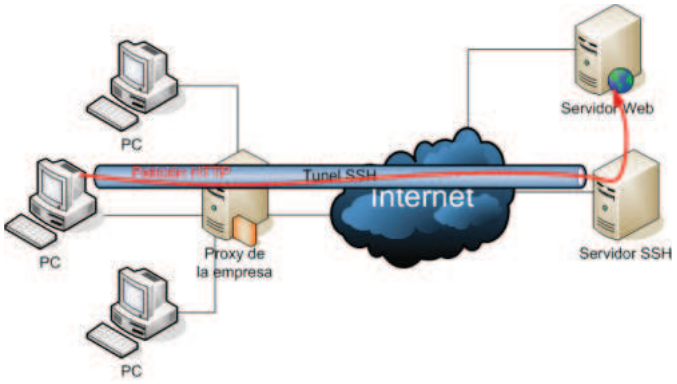


Figura 2.9: Arquitetura de um túnel SSH [Farhat, 2011][14].

Neste exemplo, podemos considerar que somente o serviço SSH (utilizando a porta padrão TCP 22) está disponível na rede da empresa. Um usuário malicioso utilizando o computador **PC** pode estabelecer um túnel a um servidor SSH o qual tenha acesso. Este túnel encapsula os dados de tráfego da aplicação não permitida (e.g. Servidor *Web*). O servidor externo redireciona este tráfego ao Servidor Web e permite que o usuário acesse esta aplicação.

Outra técnica bastante difundida é o tunelamento HTTP. Esta técnica consiste no encapsulamento de tráfego de aplicações de rede utilizando o protocolo HTTP. Da mesma forma que o tunelamento por SSH o tráfego também é encriptado. O canal de comunicação estabelecido para esse tipo de técnica é conhecido como túnel HTTP.

Existem diversas implementações para configuração de túneis HTTP. Resumidamente é composto de um *software* cliente-servidor permitindo que sejam usados em condições de redes restritas.

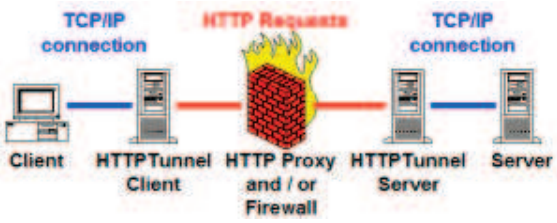


Figura 2.10: Arquitetura de um túnel HTTP [Weber, 2010][15].

A Figura 2.10 mostra uma representação simples de um túnel HTTP. Neste tipo de implementação todo o tráfego que atravessa o sistema de *firewall* é feito utilizando a porta

[14]Disponível em http://h20-platform.blogspot.com.br/2012/12/how-to-access-any-blocked-website-using.html. Acessado em 27/09/2012.

[15]Disponível em http://http-tunnel.sourceforge.net/ . Acessado em 27/09/2012.

TCP 80. Considere que somente tráfego HTTP é permitido nesta rede. O computador **Client** utiliza um *software Client* para estabelecer uma conexão ao servidor de túnel HTTP. O software encripta o pacote de qualquer aplicação (e.g. P2P, jogos) e encaminha ao servidor através do túnel. O servidor de túnel HTTP recebe o pacote, decripta e encaminha para o servidor externo da aplicação encapsulada. A resposta deste servidor de aplicação trafega no sentido inverso, executando-se o mesmo processo de encriptação e decriptação. Para o sistema de *firewall* da rede o pacote trata-se de um pacote HTTP e não irá bloquear este tráfego.

Além de técnicas de tunelamento deve-se considerar que portas conhecidas são um alvo fácil para usuários maliciosos. Como a configuração de portas TCP depende do usuário, qualquer serviço utilizando o protocolo TCP/IP pode ser configurado com uma porta conhecida (e.g. Servidor SSH configurado para aceitar conexões na porta TCP 80). Desta forma, em sistemas básicos de *firewall* baseado na associação de portas um tráfego desconhecido ou não permitido pode trafegar livremente pela rede.

2.3 GROUND TRUTH

O *Ground Truth* é um termo bastante usual para representar a coleção de dados ou informações que possibilitam a construção de uma base referencial de comparação. Em [Ellis et al., 2002] o *Ground Truth* representa o conjunto de dados de treinamento descrevendo a similaridade entre pares para a abstração de uma métrica.

Em termos gerais para problemas de classificação de tráfego, o *ground truth* representa o referencial de comparação para que se possa determinar a acurácia de determinada abordagem. O *Ground Truth* representa também um ponto de referência que serve como base para comparação de diferentes classificadores [Lee et al., 2011]. Estabelece portanto dentro da área a qual se está pesquisando o parâmetro confiável para verificação do resultado de um determinado classificador.

Especificamente em se tratando de problemas relacionados a classificação de tráfego de Internet, a maioria das modelagens, sistemas de firewall e sistemas de detecção de intrusão requerem dados de tráfego com informações suficientes sobre as aplicações e protocolos associados a cada pacote ou fluxo de pacotes [Gringoli et al., 2009]. Esses dados são essenciais para a construção de um *Ground Truth* confiável para avanço nas pesquisas nesta área.

No entanto, a disponibilização de tráfego identificado é escassa e em alguns casos representa alto custo [Estrada and Nakao, 2010]. Isso deve-se ao fato de não haver grande disponibilidade de bases que contenham todas as informações sobre as aplicações e proto-

colos associados ao tráfego. Essa escassez é resultado da necessidade de confidencialidade e da preservação da privacidade dos dados.

2.4 CLASSIFICAÇÃO DE TRÁFEGO

O processo de classificação de tráfego consiste num processo automizado de categorização para tráfego de rede. Esse processo é feito com base em vários parâmetros (e.g. número de porta TCP ou protocolo) para respectivas classes ou categorias de aplicação. O resultado de classificação deve permitir a diferenciação de tráfego gerado pelo usuário da rede.

Esta área de pesquisa tornou-se ao longo do tempo um dos maiores desafios em redes de telecomunicações [Callado et al., 2009]. Classificar o tráfego tornou-se de fundamental importância em atividades relacionadas à redes de computadores. Essas atividades vão de monitoramento e segurança à qualidade de serviços prestados e provisionamento de recursos [Moore and Zuev, 2005]. Num sentido mais amplo, desenvolver estudos neste campo de pesquisa pode auxiliar na otimização das operações de rede e no planejamento de melhorias em arquiteturas de redes futuras [Kim et al., 2008].

Inicialmente, todas as aplicações de rede utilizavam protocolos conhecidos e portas que facilmente permitiam a identificação do tráfego. Contudo, esta situação mudou pela utilização de técnicas de camuflagem de tráfego e utilização de portas efêmeras (dinâmicas) [Gargiulo and Sansone, 2010].

Com isso, um método confiável que possa identificar corretamente uma categoria de aplicação, ou tráfego gerado por um host na rede, ainda precisa ser desenvolvido. Diversas abordagens foram desenvolvidas com o objetivo de tratar o problema de classificação de tráfego de Internet através da otimização de acurácia nos resultados de classificação.

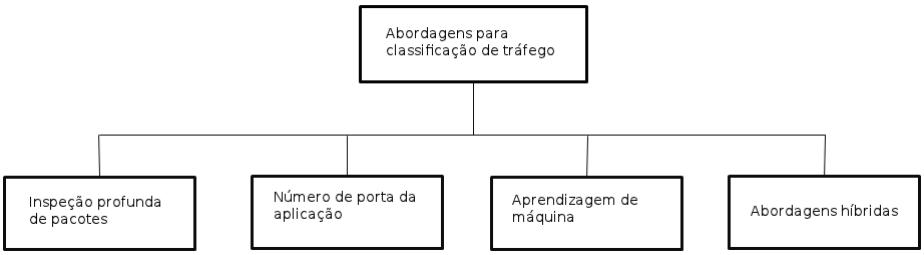


Figura 2.11: Taxonomia para classificação de tráfego de Internet.

A Figura 2.11 apresenta uma proposta de taxonomia para abordagens de classificação de tráfego. Estas abordagens estão baseadas em técnicas baseadas em DPI (inspeção profunda de pacotes), número de porta de aplicação, aprendizagem de máquina (ML) e

abordagens híbridas.

As abordagens baseadas em DPI tratam da classificação baseada na análise do conteúdo dos pacotes (*payload*) [Choi, 2006] e [Venosa et al., 2008]. Tais abordagens utilizam características determinísticas dos pacotes para extrair assinaturas que possam identificar determinado tipo de tráfego.

A classificação baseada em associação de portas utiliza a definição de portas da IANA para determinar o resultado de classificação. O *header* da camada de transporte é analisado e o número de porta é identificado. O classificador então faz a correlação do número da porta com as portas definidas pela IANA.

Os classificadores que utilizam aprendizagem de máquina [Moore and Zuev, 2005], [Auld et al., 2007], [Gu et al., 2010] e [Szabo et al., 2012] são implementações de algoritmos de aprendizagem semi-supervisionados. Estes classificadores constroem modelos a partir de poucos dados de treinamento identificados para classificar uma massa maior de dados não identificada.

Considerando abordagens híbridas diversas técnicas foram desenvolvidas, como a proposta de [Lee et al., 2011] que apresenta um *benchmark* de várias abordagens de classificação e combina os resultados para obtenção de um novo classificador baseado em pesos configuráveis.

RESUMO

Neste capítulo foram apresentados os conceitos gerais para compreensão do problema de classificação de tráfego de Internet destacando técnicas de camuflagem de tráfego. As técnicas de camuflagem utilizam portas de aplicações conhecidas para trafegar conteúdo não permitido na rede. Especialmente a porta TCP 80 do protocolo HTTP é alvo de técnicas de tunelamento e utilização indevida. Uma taxonomia para classificação de tráfego de Internet foi apresentada destacando os principais trabalhos nesta área.

CAPÍTULO 3

NETRAMARK - BENCHMARK PARA CLASSIFICAÇÃO DE TRÁFEGO

Analisando as pesquisas realizadas em classificação de tráfego de Internet verificou-se que diversas abordagens já foram desenvolvidas. Em virtude de aspectos específicos de cada tipo de tráfego, ambientes de rede, métricas e bases de dados disponíveis torna-se um desafio comparar os resultados destas abordagens de maneira objetiva.

Em [Lee et al., 2011] é descrita uma abordagem que propõe um *benchmark* chamado *NeTraMark*. Esta abordagem implementa técnicas do estado da arte em classificação de tráfego em uma única ferramenta. Este capítulo descreve detalhadamente esta ferramenta. Analisa ainda, sua aplicabilidade em cenários de tráfego HTTP real e tráfego não HTTP camuflado, no qual o administrador da rede não possui conhecimento das aplicações que trafegam na rede.

3.1 REQUISITOS PRINCIPAIS DA NETRAMARK

Visando a eficiência da abordagem, [Lee et al., 2011] elencou 6 (seis) requisitos principais que nortearam o desenvolvimento da ferramenta:

Comparabilidade: Permitir que, através do *benchmark*, usuários possam comparar diferentes abordagens, utilizando as mesmas métricas de desempenho e bases de estudo. A definição das categoria de aplicações também deve ser única. Esse requisito visa facilitar a quantificação das melhorias obtidas em uma abordagem sobre a outra;

Reproducibilidade: Permitir que resultados obtidos por diferentes grupos de pesquisa possam ser reproduzidos através de um ambiente único. Esse requisito visa ainda estimular a adoção de algoritmos de classificação mais eficientes para determinado tipo de tráfego, através da reprodução de experimentos;

Eficiência: Permitir maior eficiência em processos de classificação de tráfego. Tal eficiência visa otimizar a produtividade durante a realização de experimentos de classificação;

Extensibilidade: Possibilitar a extensão da técnica. Esse requisito visa permitir que novas técnicas e algoritmos de classificação sejam incorporados à ferramenta aumentando sua abrangência para diferentes tráfegos e cenários de classificação;

Sinergia: Minimizar as fraquezas de determinadas abordagens em relação a determinados tipos de tráfego (e.g. abordagens de análise do *payload* não são eficazes para tráfego encriptado). Através da sinergia entre os classificadores, equilibrando pontos fortes e fracos de cada abordagem, pode-se resultar em classificações mais completas e acuradas. A combinação dos classificadores visa permitir esta sinergia;

Facilidade de uso: Permitir que usuários possam configurar parâmetros relativos aos classificadores de maneira rápida e simples.

Na seção seguinte são apresentadas as características gerais da *NeTraMark*.

3.2 CARACTERÍSTICAS GERAIS

A motivação principal para o desenvolvimento desta abordagem foi auxiliar no processo de identificação de tráfego possibilitando a comparação de desempenho entre diversos classificadores existentes.

Essa comparação continua sendo um desafio na área de classificação de tráfego. Cada abordagem desenvolvida utiliza diferentes métricas de desempenho e categorias de aplicação. Em sua maioria são aplicadas em bases públicas geralmente anonimizadas e/ou sem a presença de *ground truth* para avaliações de acurácia.

A ferramenta *NeTraMark*, traz a implementação dos principais classificadores de tráfego existentes na literatura, podendo ser utilizada para identificar o tráfego pertencente a aplicações como jogos, P2P, SSH, *Web* (tráfego HTTP e HTTPS), entre outras.

Adicionalmente, a *NeTraMark* inclui a proposta e implementação de um classificador de tráfego denominado *Weighted Vote*. Esse classificador combina o resultado dos demais classificadores que ela implementa por um processo de votação baseado em pesos configuráveis, buscando um melhor resultado de classificação.

3.2.1 Configuração do Ground Truth

Buscando resolver esta questão e flexibilizar o processo de utilização da ferramenta a abordagem permite que os usuários configurem qualquer classificador implementado na ferramenta como *ground truth*. Desta maneira a *NeTraMark* mantém como parâmetro de referência um dos classificadores reconhecidamente utilizados pela comunidade cientica

para classificação de tráfego. A *NeTraMark* permite a utilização de um *ground truth* confiável.

A Figura 3.1 mostra a tela de configuração do parâmetro *ground truth* na *NeTraMark*.

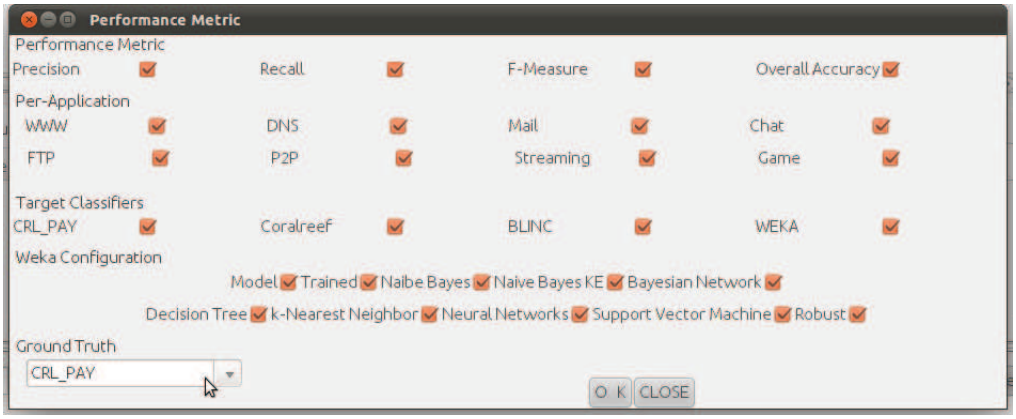


Figura 3.1: Configuração do *ground truth* na *NeTraMark*.

Na próxima seção são apresentadas as categorias de aplicação definidas na ferramenta *NeTraMark*.

3.2.2 Categorias de Aplicação

Visando atender o requisito de comparabilidade a *NeTraMark* padroniza categorias de aplicação para um conjunto de aplicações e protocolos. Esta padronização também permite a comparação de um classificador que identifica apenas uma aplicação específica com outro classificador que identifica uma categoria (grupo de aplicações). Em resumo permite uma comparação em alto nível entre classificadores utilizando grupos de aplicações.

Por padrão a *NeTraMark* utiliza as categorias descritas na Tabela 3.1. Essas categorias podem ser modificadas alterando o código fonte da ferramenta.

Tabela 3.1: Categorias de aplicação da *NeTraMark*.

Categoria	Aplicações e protocolos
Web	HTTP, HTTPS
P2P	FastTrack, eDonkey, BitTorrent, Ares,Gnutella, WinMX, OpenNap,MP2P, SoulSeek, Direct Connect, GoBoogy, Soribada, PeerEnabler, Napster, Blubster, FileBEE, FileGuri, FilePia, IMESH, ROMNET, HotLine, Waste
FTP	FTP
DNS	DNS
Mail/News	BIFF, SMTP, POP, IMAP, IDENTD, NNTP
Streaming	MMS(WMP), Real, Quicktime, Shoutcast, Vbrick Streaming, Logitech Video IM, Backbone Radio, PointCast, ABACast
Network Operation	Netbios, SMB, SNMP, NTP, SpamAssasin, GoToMyPc, RIP, ICMP, BGP, Bootp, Traceroute
Encryption	SSH, SSL, Kerberos, IPSec, ISAKMP
Games	Quake, HalfLife, Age of Empires, DOOM, Battle field Vietnam, WOW, Star Sieze, Everquest, Startcraft, Asherons, HALO
Chat	AIM, IRC, MSN Messenger, Yahoo messenger, IChat, QNext, MS Netmeet, PGPFone, TALK
Attack	Adress scans, Port Scans
Unknown	-

Na próxima seção são apresentados os classificadores que foram implementados na *NeTraMark*.

3.2.3 Classificadores Implementados

A ferramenta *NeTraMark* (*Network Traffic Benchmark*) traz em sua implementação onze classificadores de tráfego de Internet: *Coralreef*, *Crl_pay*, *BLINC*, C4.5 *Decision Tree*, *Naive Bayes*, *Naive Bayes Kernel Estimation*, *Bayesian Networks*, *K-Nearest Neighbor* , *Neural Network* e *Support Vector Machine* (SVM).[1] Além disso, implementa um novo classificador denominado *Weighted Vote*. A seguir é apresentada uma descrição de cada um desses classificadores.

O classificador *Coralreef*[2] estabelece um processo de classificação baseado na correlação entre a aplicação e um número de porta conhecido registrado no IANA[3]. Através destas associações serviços podem ser rapidamente identificados com base no número da porta [Callado et al., 2009]. Para manutenção do classificador é suficiente que novas associações sejam inseridas na base de dados de portas. A despeito da rapidez e facilidade de uso deste classificador vários estudos apontam que o mesmo possui desempenho abaixo

[1]Algoritmos de aprendizagem de máquina(ML)do suite WEKA: Data Mining Software in Java. Acessível em http://www.cs.waikato.ac.nz/ml/weka/

[2]CoralReef. Acessível http://www.caida.org/tools/measurement/coralreef/

[3]Internet Assigned Numbers Authority (IANA). http://www.iana.org/assignments/port-numbers

de 70% em classificar tráfego que não está utilizando portas conhecidas [Lee et al., 2011].

O classificador *Crl_pay* [Choi, 2006] é baseado na abordagem de extração de assinaturas existentes nas aplicações investigando os protocolos e as características determinísticas de seus pacotes. Estas assinaturas são extraídas apartir de caracteres específicos, símbolos, combinações de palavras-chave e até mesmo propriedades estruturais de um pacote. O estudo de [Choi, 2006] apontou uma melhoria de 11% na acurácia de classificação em relação ao método baseado em portas TCP. No entanto, o gerenciamento das assinaturas torna-se complexo pela diversidade de aplicações. O acesso ao conteúdo do *payload* traz à tona questionamentos sobre a privacidade dos dados. Para tráfego encriptado esse classificador é ineficaz.

o classificador *BLINC* [Karagiannis et al., 2005] determina o tipo de tráfego estabelecendo um perfil para um *host* através da sua interação social na rede. Este perfil é comparado com assinaturas *Built-in* no classificador. Em bases anonimizadas esse classificador pode ser utilizado, pois implementa o processo de classificação no escuro e não requer acesso ao *payload* dos pacotes. No entanto, esta abordagem apresenta algumas limitações, como a impossibilidade de identificação de tipos diferentes da mesma aplicação, tráfego encriptado na camada de transporte e de aplicações com uso de NAT (*Network Adress Translation*).

C4.5 Decision Tree [Williams et al., 2006a] implementa um modelo em estrutura de árvore a partir de um conjunto de dados de treinamento usando o conceito de entropia da informação. Cada nó representa o teste de características do fluxo de pacotes. Os galhos da árvore o desenlace do teste e cada folha a identificação de uma classe. Para este propósito uma lista ordenada de classes deve ser fornecida. O classificador varre a árvore de decisão da raiz até a folha. A identificação da folha é o resultado de classificação (categoria de aplicação) do tráfego.

Naive Bayes [Williams et al., 2006b] é baseado no teorema *Bayes*, o qual analiza a correlação entre cada característica do fluxo de pacotes e as categorias de aplicação derivando uma probabilidade condicional entre valores característicos e a classe a qual se deseja obter resultado.

Naive Bayes Kernel Estimation (NBKE) [Williams et al., 2006a] implementa uma generalização do classificador *Naive Bayes* utilizando múltiplas distribuições gaussianas, tornando-se mais acuradas que uma única distribuição para classificação de tráfego.

Bayesian Network [Auld et al., 2007] e [Gu et al., 2010] estabelece um modelo de gráfico acíclico dirigido (DAG) que representa um conjunto de categorias de aplicação como nós e sua relação probabilística como bordas. Considerando os fluxos de pacotes

onde a suposição de independência condicional não é válida a aprendizagem deste modelo pode superar o *Naive Bayes*. Um exemplo para *Bayesian Network*, é a representação probabilística entre doenças e sintomas. Dados os sintomas o classificador pode ser utilizado para computar as probabilidades da presença de várias doenças.

K-Nearest Neighbors (KNN) [Estrada and Nakao, 2010] estima a distância Euclidiana para cada instância de teste em relação ao espaço de características n-dimensional. O classificador atribui uma identificação para o tráfego analisando a votação da maioria entre os vizinhos k-mais próximos a tupla[4] de teste.

O classificador *Neural Network* [Auld et al., 2007] [Williams et al., 2006a] implementa o algoritmo mais comum de redes neurais chamado *Multilayer Perceptron*. Este algoritmo possui uma camada de *input* de *neurons* representando características de tráfego e uma camada de *output* de *neurons* representando classes ou categorias de aplicação. Há ainda uma ou mais camadas escondidas entre estas. Este classificador torna-se proibitivamente lento para grandes quantidades de instâncias de treinamento.

O classificador *Support Vector Machine* (SVM) utiliza um algoritmo mais rápido para treinamento chamado *Sequential Minimal Optimization* (SMO) [Williams et al., 2006a] e [Li et al., 2011]. Utiliza a classificação em pares (*pairwise*) para quebrar um problema multi-classe em um conjunto de subproblemas de 2 (duas) dimensões e elimina a necessidade de otimização numérica. Pelos estudos desenvolvidos com este classificador ele necessita de poucos dados de treinamento para obter bons resultados de acurácia, tornado-o mais prático em classificação de tráfego de Internet já que dados de treinamento são escassos.

O estudo de [Kim et al., 2008] indica que os classificadores *SVM*, *KNN* e *Neural Network* apresentam alta acurácia para tráfego heterogêneo, tais como HTTP, DNS, Mail e P2P. O que interfere em termos de desempenho computacional e acurácia destes classificadores é a quantidade de dados utilizados para treinamento. Foi demonstrado também que *Bayesian Network*, *Naive Bayes Kernel Estimation*, *Naive Bayes*, e *C4.5 Decision Tree* , requerem muito mais instâncias do que estes para atingir níveis aproximados de acurácia.

O classificador *Weighted Vote*, proposto em [Lee et al., 2011], combina o resultado dos dez classificadores anteriormente citados em um processo de votação baseado em pesos configuráveis. Para obtenção de *Weighted Vote* a *NeTraMark* requer a configuração de um *ground truth* selecionado a partir de um dos classificadores implementados e a atribuição manual de pesos para cada classificador.

[4]Conjunto ou lista ordenada de valores.

O algoritmo implementado na *NeTraMark* determina então *Weighted Vote* extraindo a resultante da combinação de duas regras: Votação da maioria baseada em pesos e o classificador com maior acurácia em relação ao *ground truth*. Como configuração padrão na *NetraMark*, os pesos estão configurados em 0,1 para todos os classificadores. Ainda por padrão o *ground truth* é o classificador *Crl_pay*.

Para clarificar o processo de obtenção de *Weighted Vote*, suponha que determinado fluxo de pacotes **A** contem tráfego P2P. Os classificadores *Coralreef*, *Crl_pay* e *BLINC* classificaram o fluxo de pacotes **A** como P2P, *Web* e P2P respectivamente. Os pesos atribuídos para estes classificadores foram 0,1 para P2P, 0,1 para Web e 0,3 para P2P. Considerando o uso do *ground truth* padrão (*Crl_pay*) o resultado para *Weighted Vote* será *Web*. Em resumo, a votação de pesos determinou que o fluxo de pacotes é P2P. No entanto o classificador com maior acurácia foi *Crl_pay*, combinando as duas regras este resultado prevaleceu na classificação.

Na próxima seção são apresentadas as métricas de avaliação de desempenho, implementadas na ferramenta *NeTraMark*.

3.2.4 Métricas de Avaliação de Desempenho

A ferramenta *NeTraMark* disponibiliza métricas padronizadas para avaliação de desempenho dos classificadores implementados. As métricas permitem uma avaliação coesa dos resultados em diferentes bases, tornando-os comparáveis.

Antes, considere as seguintes definições que são utilizadas para o cálculo das métricas de avaliação de desempenho:

- *Verdadeiros Positivos (TP)* - Quantidade de fluxos de pacotes classificados corretamente em relação ao *ground truth*;
- *Falsos Positivos (FP)* - Quantidade de fluxos de pacotes incorretamente classificados como a aplicação desejada. (e.g. tráfego identificado como *Web* porém é uma aplicação não HTTP utilizando a porta TCP 80);
- *Falsos Negativos (FN)* - Quantidade de fluxos de pacotes incorretamente classificados em relação ao *ground truth*.

Considerando a acurácia, as métricas implementadas na *NeTraMark* são:

- *Overall Accuracy*. Representa a acurácia geral de um classificador para toda a base de tráfego, considerando todas as categorias de aplicação. Representa então,

a proporção entre o somatório de TP e o somatório de TP e FP, e é obtida através da seguinte equação: $\left(\frac{TP}{(TP+FP)}\right)$.

- *Precision* - Representa o percentual de fluxos da base de tráfego classificados corretamente para uma dada aplicação. Representa exatidão ou qualidade. Essa métrica é obtida através da seguinte equação: $\left(\frac{TP}{(TP+FP)}\right)$;
- *Recall* - Representa o percentual de uma categoria de aplicação adequadamente identificada na base de tráfego. Representa completude ou quantidade. Essa métrica é calculada através da seguinte equação: $\left(\frac{TP}{(TP+FN)}\right)$;
- *F-Measure* - Uma métrica amplamente utilizada em recuperação da informação e classificação [Witten and Frank, 2005] e [Hripcsak and Rothschild, 2005] que representa a acurácia de um classificador para uma determinada classe de aplicação ao combinar a capacidade média de *Precision* e *Recall* harmonicamente. O valor dessa métrica é obtido através da seguinte equação: $\frac{2(Precision \times Recall)}{(Precision+Recall)}$.

Em termos de desempenho computacional, a *NeTraMark* implementa ainda 2(duas) métricas:

- *Classification Time*. Tempo que o classificador leva para apresentar os resultados de classificação de uma determinada base;
- *Learning time*. Tempo que um algoritmo de aprendizagem de máquina leva para construir uma modelo de classificação a partir dos dados de treinamento.

A seção seguinte descreve as funcionalidades da ferramenta *NeTraMark*.

3.3 FUNCIONALIDADES

Nesta seção são apresentadas as funcionalidades da ferramenta *NeTraMark* na classificação de tráfego. A ordem das seções a seguir pode ser considerada como a descrição de um processo passo a passo para classificação de tráfego na ferramenta.

3.3.1 Configurações Iniciais

O código fonte da *NeTraMark*[5], bem como, o tutorial para instalação do ambiente está disponível para uso de pesquisadores e estudantes que queiram explorar esta abordagem.

[5]Disponível mediante solicitação e somente para propósito de pesquisa. Acessível em http://popeye.snu.ac.kr/∼sclee/NeTraMark

Para iniciar o processo de classificação de determinado tráfego é necessário que algumas configurações sejam realizadas na ferramenta. O objetivo destas configurações é estabelecer os parâmetros que serão utilizados na determinação dos resultados de classificação e na apresentação destes resultados.

Inicialmente, na opção *Visualization/Statistics/Classification configuration* do menu o usuário deve configurar quais classificadores e as respectivas categorias de aplicação as quais serão apresentados resultados de classificação. A Figura 3.2 mostra as opções de configuração.

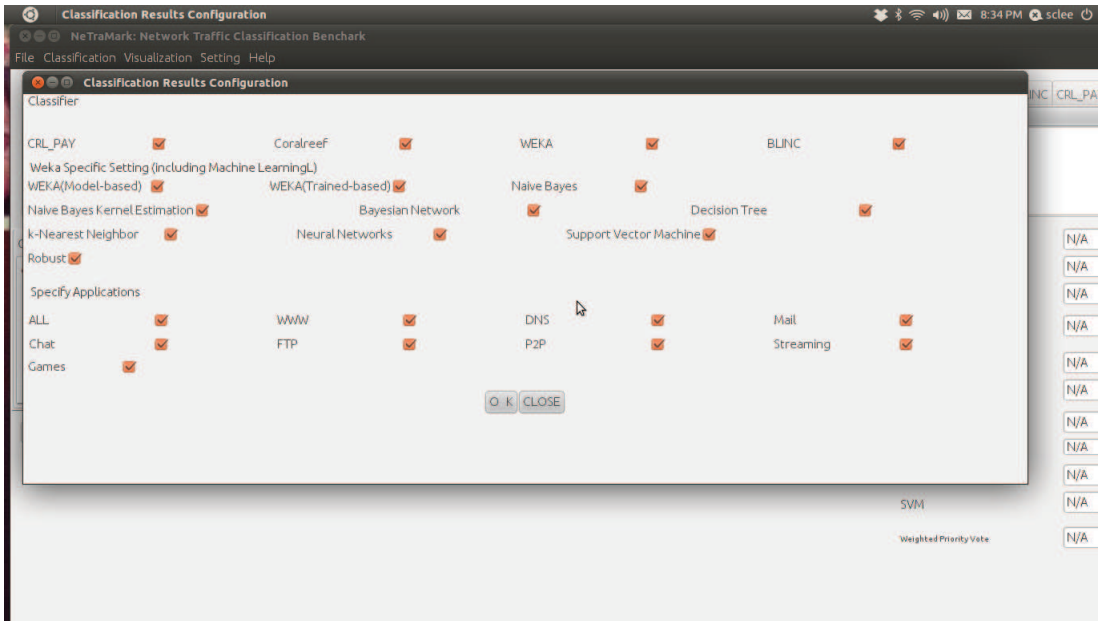


Figura 3.2: Seleção de resultados de classificação.

Em seguida, na opção *Visualization/Benchmark/Benchmark configuration* do menu, o usuário configura as opções de *benchmark* que serão utilizadas. A Figura 3.3 mostra essa configuração.

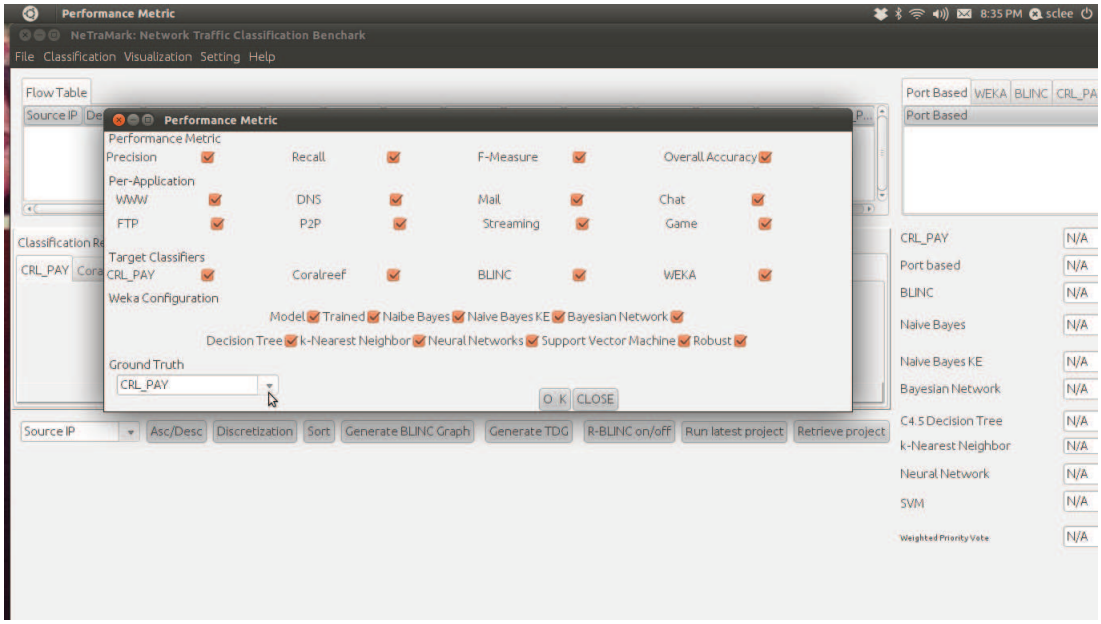


Figura 3.3: Configuração do *Benchmark*.

Nesta opção a ferramenta permite que sejam associados os classificadores que serão comparados às categorias de aplicação e às métricas de desempenho. Pode-se realizar comparação entre classificadores específicos, utilizando uma ou todas as métricas disponíveis na ferramenta. É definido ainda nesta configuração qual classificador será utilizado como *ground truth*. O classificador baseado na análise do *payload*, *Crl_pay*, é o *ground truth* padrão na *NeTraMark*.

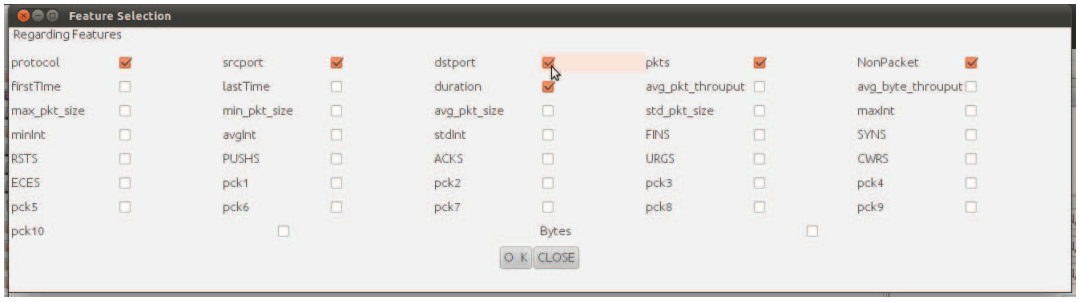
Considerando o uso do classificador *Weighted Vote* faz-se necessário configurar manualmente um peso para cada classificador, associando-o a uma categoria de aplicação. Os valores de peso disponíveis na ferramenta variam de 0,1 até 1. Por exemplo, sabendo-se que o classificador *BLINC* possui boa acurácia para tráfego P2P, pode-se atribuir para este classificador o peso 0,7 para a categoria de aplicação P2P. Este processo é baseado na experiência e conhecimento do operador ou em experimentos prévios realizados na própria ferramenta.

Para realizar a configuração dos pesos, o usuário deve acessar a opção *Setting/Priority values* do menu. A Figura 3.4 mostra a tela de configuração para os pesos.

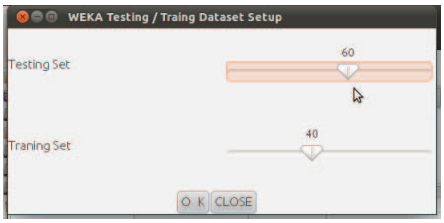
Figura 3.4: Configuração do pesos na *NeTraMark*.

Considerando os classificadores baseados em aprendizagem de máquina (ML), a fer-

ramenta permite a seleção de características dos pacotes, chamadas *features*, a serem utilizadas para o modelo de aprendizagem. Essa configuração está disponível na opção *Classification/WEKA/ARFF file generation/Feature selection*. É possível ainda, determinar o percentual que será utilizado por estes modelos para instâncias de treinamento e de teste. Para isso, o usuário deve acessar a opção *Classification/WEKA/ARFF file generation/ Train/Test dataset configuration*. A Figura 3.5 mostra as opções de configuração disponíveis.



(a)



(b)

Figura 3.5: Opções de configuração para classificadores baseados em ML.

3.3.2 Classificadores

A *NeTraMark* disponibiliza a opção de execução dos classificadores individualmente ou em *batch* executando todos os classificadores de uma única vez.

Para executar individualmente, o usuário deve selecionar o classificador através do menu na opção *Classification*. Este processo torna-se útil em cenários onde não há necessidade de ter os resultados de todos os classificadores, ou se deseja analisar somente um, ou alguns classificadores específicos. Para a execução em *batch* o usuário deve clicar no botão *Run latest project* e selecionar a base com o tráfego que se deseja classificar.

Vale ressaltar que para uso na ferramenta *NeTraMark* a base de tráfego (arquivo de captura) deve estar em formato *pcap*[6].

[6]Pcap (*packet capture*) consiste em uma API (*application programming interface*) para captura de tráfego de rede. Para sistemas operacionais *Unix-like* é implementada através da *library libpcap*. Para sistemas *Windows* através da *WinPcap*.

3.3.3 Resultados de Classificação e Benchmark

Após o processo de classificação são apresentados os resultados de identificação de tráfego para cada classificador. A Figura 3.6 mostra a tela de resultados apresentados pela *NeTraMark*.

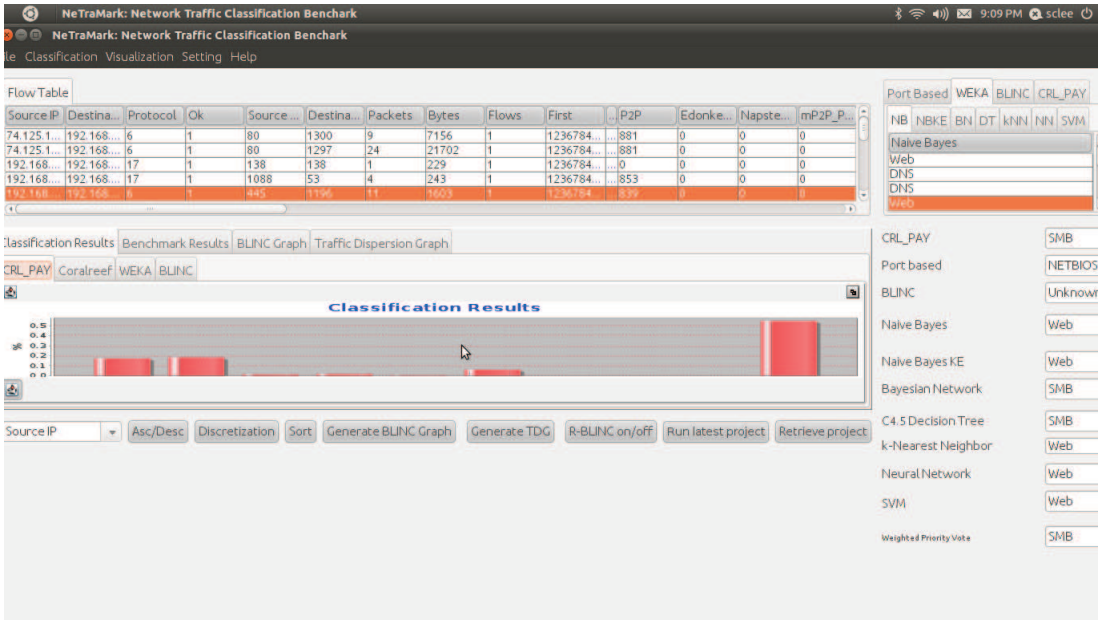


Figura 3.6: Resultados de classificação de tráfego.

De acordo com a figura acima na guia *Flow Table* é possível selecionar um fluxo de pacotes específicos e verificar os resultados para cada classificador. Os resultados individuais de cada um são apresentados no canto inferior direito da tela.

Na guia *Classification Results* a ferramenta apresenta o resultado geral de classificação para cada classificador. Ao selecionar o classificador, são apresentados em gráficos os percentuais de tráfego identificados para cada categoria de aplicação. Cada barra do gráfico representa o percentual de determinada categoria de aplicação identificada.

Na guia *Benchmark Results*, são apresentados os resultados das métricas de desempenho disponibilizadas na ferramenta. Essa guia permite uma análise comparativa dos resultados de classificação entre as diferentes técnicas de classificação disponibilizadas na ferramenta.

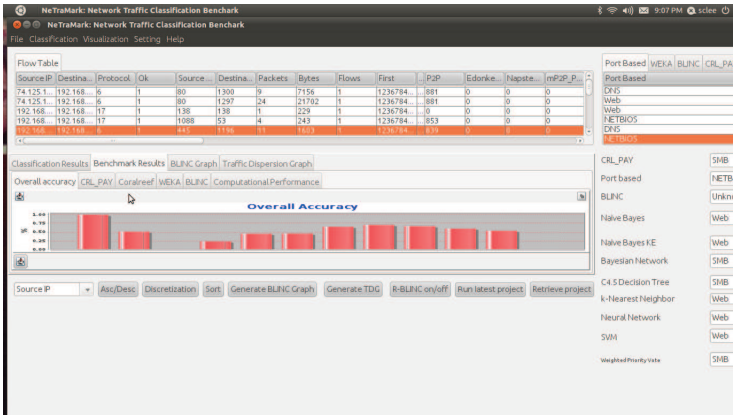


Figura 3.7: *Overall accuracy* dos classificadores.

Na Figura 3.7, é apresentada em forma de gráfico a acurácia geral dos classificadores considerando toda a base. As barras do gráfico indicam em termos percentuais qual a acurácia de cada classificador considerando todo o tráfego contido na base.

Já na Figura 3.8 são apresentadas as métricas de acurácia *Precision*, *Recall* e *F-Measure*. O gráfico indica em termos percentuais a acurácia relativa à cada categoria de aplicação encontrada na base, em comparação ao classificador configurado como *ground truth*.

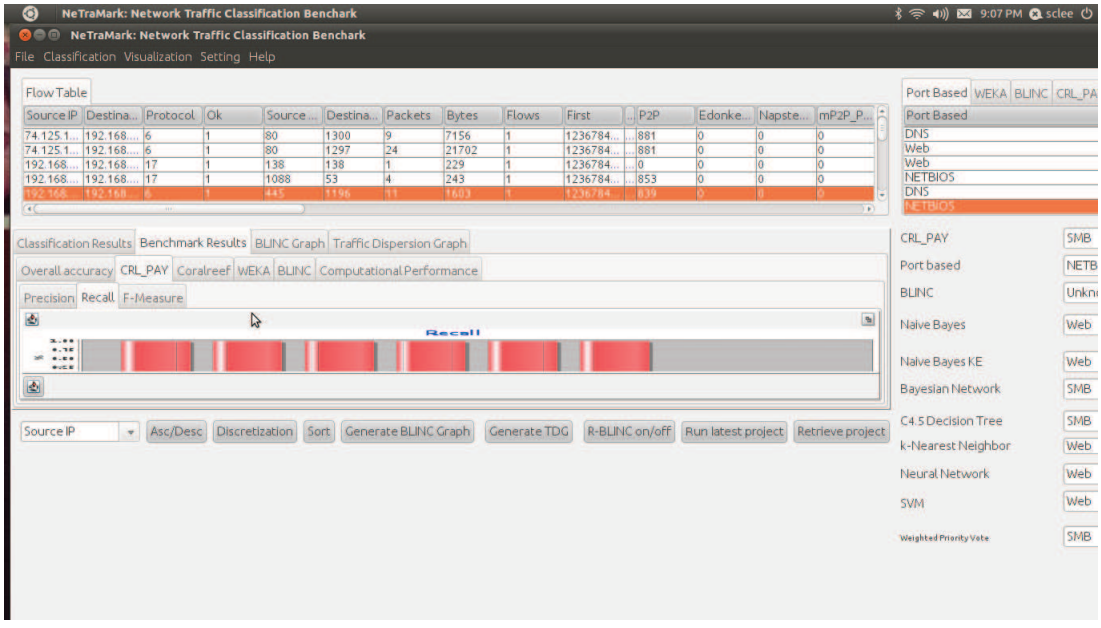


Figura 3.8: Acurácia dos classificadores por categoria de aplicação.

Os resultados de *benchmark* permitem a avaliação de desempenho dos classificadores implementados, determinando melhores classificadores para cada tipo de tráfego anali-

sado.

Na próxima seção são analisados alguns problemas encontrados na ferramenta *NeTraMark*.

3.4 PROBLEMAS ENCONTRADOS

Considerando as pesquisas em classificação de tráfego, a disponibilização de bases com *ground truth* é limitada e escassa [Venosa et al., 2008]. Por outro lado, em ambientes de rede organizacionais, o conhecimento do administrador sobre as aplicações que trafegam na rede é limitado e muitas vezes inexiste.

Considerando a necessidade de se classificar tráfego HTTP utilizando a *NeTraMark*, há ausência de bases suficientes que possibilitem um estudo mais aprofundado de acurácia dos classificadores. Há ainda o fato de que em redes com usuários maliciosos, o tráfego gerado por estes usuários é camuflado e portanto, desconhecido do administrador da rede.

Devido a estes fatores, a atribuição de pesos, bem como a determinação do melhor *ground truth*, torna-se um processo empírico e muitas vezes inadequado. Este processo restringe-se então, ao conhecimento do operador ou ao histórico do desempenho dos classificadores baseado em experimentos prévios na ferramenta. Conclui-se que nestas condições o classificador *Weighted Vote* nem sempre apresentará o melhor resultado de classificação [Lee et al., 2011].

Utilizando técnicas de camuflagem usuários maliciosos podem tráfegar conteúdo de outras aplicações através do protocolo HTTP pela porta TCP 80. Portanto, utilizar os classificadores implementados na *NeTraMark*, para determinar o que de fato é tráfego HTTP real[7] torna-se um desafio para o administrador da rede nestas condições.

RESUMO

Este capítulo apresentou a ferramenta *NeTraMark*. Através da descrição de seus requisitos de desenvolvimento, características gerais e funcionalidades permitiu-se um conhecimento mais aprofundado desta abordagem. Especificamente na seção funcionalidades foi descrito um processo passo a passo para classificação de tráfego na ferramenta.

A ausência de bases *ground truth* que permitam o aprofundamento no estudo dos classificadores e a falta de conhecimento do administrador da rede em cenários de camuflagem de tráfego, tornam o processo de classificação na ferramenta mais complexo e por

[7]Tráfego gerado por aplicações contendo pacotes genuinos do protocolo HTTP por uma porta TCP conhecida.

muitas vezes inadequado.

Especificamente para identificação de tráfego HTTP real, em cenários de camuflagem de tráfego na porta TCP 80, percebe-se a necessidade de melhorar o processo de ajuste de pesos do classificador *Weighted Vote* para obtenção de melhores resultados de classificação.

CAPÍTULO 4

METODOLOGIA DE CLASSIFICAÇÃO DE TRÁFEGO COM A NETRAMARK

O problema descrito no Capítulo 3 está em estabelecer um processo empírico de atribuição de pesos e definição de *ground truth* para classificar tráfego HTTP real na *NeTraMark*. Em condições reais, o administrador da rede geralmente não conhece todas as aplicações que estão trafegando na rede. Para que haja uma maior acurácia do classificador *Weighted Vote* em cenários de camuflagem através da porta TCP 80 esse conhecimento é essencial. Logo, evitar este impasse na classificação de tráfego HTTP na *NeTraMark* torna mais fácil a tarefa do administrador. Esta é a proposta deste trabalho.

Este capítulo descreve detalhadamente a metodologia aplicada incluindo a proposta deste novo classificador, além de fazer uma avaliação seus resultados em relação aos classificadores implementados na *NeTraMark*.

4.1 AMBIENTE DE REDE E CENÁRIOS DE TRÁFEGO

Para o desenvolvimento desta proposta foi necessário estabelecer um *testbed* para simulações de cenários de camuflagem de tráfego de aplicações através da porta TCP 80. Este ambiente controlado foi composto por uma infraestrutura de rede e bases de dados *ground truth* que representam tais cenários. O objetivo de tais simulações consistiu na criação de bases de dados de tráfego de rede em formato *pcap* contendo tráfego HTTP real e simulações de situações de camuflagem na porta TCP 80. Estes cenários possibilitaram uma avaliação de acurácia dos classificadores implementados na ferramenta *NeTraMark*. Esta seção apresenta a infraestrutura de rede utilizada para o estudo proposto e os cenários estudados.

4.1.1 Infraestrutura de Rede

A Figura 4.1 apresenta a infraestrutura de rede utilizada nos estudos aqui realizados. A infraestrutura consiste de uma rede doméstica conectada à Internet através de um enlace ADSL (*Asymmetric Digital Subscriber Line*) com largura de banda de 15 Mbps *downs-*

tream e 1 Mbps *upstream*. Nessa rede há três *hosts* e um roteador ADSL interconectados via *Ethernet* à 100 Mbps. Os *hosts 1* e *2* atuam como geradores de tráfego controlado. O *host 3* atua como o *gateway* padrão da rede e é utilizado para capturar todo o tráfego da rede com o auxílio da ferramenta *Wireshark*.

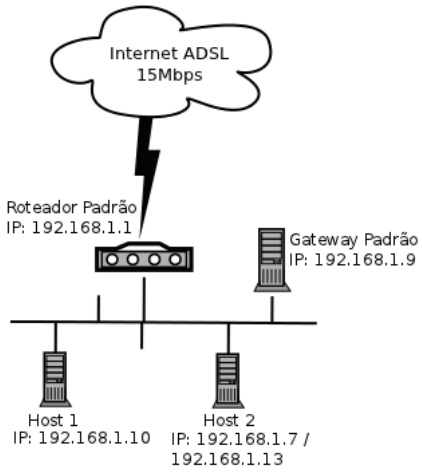


Figura 4.1: Topologia de rede do ambiente controlado.

Abaixo o detalhamento dos recursos utilizados para este ambiente controlado:
Configuração dos hosts

- ***Host 1:*** Sistema operacional UBUNTU *Linux Natty Narwhal* versão 11.04, CPU Intel atom 1.3 GHZ, 2GB de memória RAM, 250GB de espaço em disco e adaptador de rede com velocidade 10/100 mbps;
- ***Host 2:*** *Dual boot*, Sistema operacional UBUNTU *Linux Natty Narwhal* versão 11.04, Sistema operacional *Microsoft Windows* 7, CPU Intel Pentium II 2 GHZ, 3GB de memória RAM, 250GB de espaço em disco e adaptador de rede 10/100/1000 mbps;
- ***host 3:*** Sistema operacional UBUNTU *Linux Natty Narwhal* versão 11.04, CPU Intel Pentium Celeron 1.7 GHZ, 2GB de memória RAM, 500GB de espaço em disco e adaptador de rede 10/100 mbps.

Softwares

- **Geração de tráfego:** WGET, *Gtk-gnutella*, SSH, *World of Warcraft* (WOW), HTTP-TUNNEL, *Google Talk*, *Emule*, *Shareaza*;
- **Captura de tráfego:** Wireshark.

4.1.2 Cenários de Tráfego em Ambiente Controlado

Uma base de dados de tráfego previamente conhecido estabelece um *ground truth* confiável para comparações de acurácia entre classificadores. Estas avaliações, podem determinar abordagens mais adequadas de classificação para categorias de aplicação específicas e cenários em ambientes de rede. Em resumo, a existência de *ground truth* determina se o classificador identificou corretamente ou não o tráfego de rede.

Para o trabalho proposto, foram estudados 10 (dez) cenários representados por bases [1] de dados geradas a partir da captura de tráfego na rede representada pela Figura 4.1. A Tabela 4.1 apresenta um resumo do tipo de tráfego gerado para cada cenário.

Tabela 4.1: Resumo dos Cenários *ground truth* Criados.

Cenário #	**Tráfego gerado**
1	Host 1 acessando sítios de Internet através do protocolo HTTP(porta TCP 80) e HTTPS(porta TCP 443)
2	Host 1 estabelecendo uma conexão SSH com servidor externo (50.57.188.33) pela porta TCP 80 para *upload* de arquivos
3	Host 1 estabelecendo um túnel SSH com servidor externo (50.57.188.33) utilizando a porta TCP 80 para acesso ao serviço de IM *Gtalk*
4	Host 2 acessando rede P2P do *Shareaza* para download de arquivos
5	Host 2 acessando rede P2P do *Emule* através de tunelamento do protocolo HTTP utilizando a porta TCP 80
6	Host 2 acessando servidor de jogo *World of Warcraft* (WOW) através de tunelamento do protocolo HTTP utilizando a porta TCP 80
7	Base 1 e Host 3 estabelecendo uma conexão SSH com servidor externo (50.57.188.33) pela porta TCP 80 para upload de arquivos
8	Base 1 e Host 3 estabelecendo um túnel SSH com servidor externo (50.57.188.33) utilizando a porta TCP 80 para acesso ao serviço de IM *Gtalk*
9	Base 1 e Host 2 acessando servidor de jogo *World of Warcraft* (WOW) através de tunelamento do protocolo HTTP utilizando a porta TCP 80
10	Base 1 e Host 2 acessando rede P2P do Emule através de tunelamento do protocolo HTTP utilizando a porta TCP 80

A escolha destes tipos específicos de tráfego deu-se pela necessidade de analisar a acurácia dos classificadores implementados na *NeTraMark*. Especialmente, para classificar tráfego Web real e tráfego não HTTP quando este está camuflado através da porta TCP 80. A combinação do tráfego (bases 7, 8, 9 e 10) visa dar maior consistência na avaliação de acurácia dos classificadores, ao se deparar com um ambiente heterogêneo de aplicações. Esta heterogeneidade é característica principal de tráfego gerado em redes nas organizações.

[1] As bases estão disponíveis através do endereço http://www.cin.ufpe.br/∼pasg/bases-cacp2/

O cenário 1 contém tráfego gerado apartir do *host 1*, através de requisições de *download* de arquivos nas portas TCP 80 e 443 (HTTP e HTTPS), em diversos sítios da Internet. O cenário 2 possui dados de tráfego gerados pelo *host 1* ao fazer upload de arquivos através de uma conexão à um servidor SSH externo. Este servidor está configurado para receber requisições deste serviço na porta TCP 80. Os dados do cenário 3 contém tráfego encriptado da aplicação *Google Talk* gerados apartir do *host 1*. Este tráfego foi encriptado através do protocolo SSH. Esta conexão é feita através de um túnel SSH configurado na porta TCP 80.

O cenário 4 contém dados gerados pelo *host 2*, ao utilizar o aplicativo *Shareaza* para acessar uma rede P2P. Já o cenário 5, consiste de tráfego gerado pelo *host 2* utilizando o *software Emule* para acessar uma rede P2P. No entanto, nesta base, foi utilizado o aplicativo *HTTP-TUNNEL* para estabelecer um túnel usando o protocolo HTTP pela porta TCP 80. Deste modo, todo o tráfego P2P foi encriptado. O cenário 6 contém tráfego gerado apartir da mesma configuração para acesso ao servidor de jogos do *game World of Warcraft* (WOW).

Nos cenários de 7 a 10 o tráfego foi combinado ao gerar na mesma base pacotes idênticos aos do cenário 1 em conjunto com os dos cenários 2,3, 5 e 6 respectivamente. Vale ressaltar que nestes cenários o tráfego foi gerado em tempo real no ambiente controlado e não foi utilizada nenhuma ferramenta adicional para concatenação (*merge*) deste tráfego.

Ao analisar o tráfego capturado identificou-se grande quantidade de pacotes desconhecidos, provavelmente gerados por outras aplicações rodando em segundo plano nos *hosts*. Simulando a execução dos classificadores em algumas destas bases originais verificou-se que o tempo de execução dos classificadores aumentou consideravelmente. Ainda, não haveria possibilidade de determinar o *ground truth* para estes pacotes. Estes fatores levaram a conclusão que o uso das bases geradas com seu conteúdo original comprometeria o tempo de execução e a avaliação da acurácia dos classificadores no escopo deste trabalho. Para resolver este problema foi aplicado um processo de filtragem em cada base para que apresentasse somente pacotes gerados pelos hosts descritos na Tabela 4.1.

4.2 ACURÁCIA DOS CLASSIFICADORES DA NETRAMARK

Esta seção avalia a acurácia dos 11 (onze) classificadores implementados na T ferramenta *NetTraMark* utilizando os cenários apresentadas na Seção 4.1.2. É avaliado o desempenho de cada classificador considerando as métricas de acurácia descritas no Capítulo 3.

Vale ressaltar que diferentemente da avaliação proposta por [Lee et al., 2011] que inclui a métrica *Overall Accuracy* , a proposta deste trabalho investiga a acurácia dos

classificadores restrita a categoria de aplicação Web (HTTP/HTTPS). Por esta razão, são utilizadas somente métricas que viabilizam a avaliação dos resultados de classificação para uma categoria de aplicação específica [Lee et al., 2011] e [Kim et al., 2008], ou seja, *Precision*, *Recall* e *F-Measure*.

Nesta etapa do estudo o objetivo é investigar os resultados e determinar quais classificadores tem melhor acurácia dados os cenários de tráfego propostos. Apartir destes resultados haverá a possibilidade de estabelecer-se a proposta para um novo classificador.

4.2.1 Avaliação de Acurácia

Seguem os resultados obtidos na classificação das bases e a análise dos resultados. Vale ressaltar, que foi utilizada a configuração padrão de pesos e *ground truth* na ferramenta *NeTraMark*.

De acordo com [Williams et al., 2006a] para um melhor desempenho computacional utilizando algoritmos baseados em aprendizagem de máquina torna-se necessário não utilizar um percentual muito alto de instâncias de treinamento (40% é o recomendado). Alguns algoritmos tornam-se proibitivamente lentos na construção do modelo de classificação com grandes quantidades de dados de treinamento. Portanto, grandes percentuais de dados de treinamento tornam inviável a utilização dos algoritmos de ML da *NeTraMark* em situações corriqueiras nas quais o administrador necessita de uma resposta rápida da análise do tráfego. Além disso o trabalho de [Williams et al., 2006a] mostra que a maioria dos algoritmos de ML, também disponíveis na *NeTraMark*, atingem acurácia maior que 95% com no mínimo de 22 (Vinte e duas) *features* de tráfego selecionadas.

Definiu-se então que neste trabalho, para os algoritmos de aprendizagem de máquina a proporção de 40% dos dados de tráfego seriam utilizados para treinamento e 60% para teste. Foram selecionadas ainda todas as 37 (Trinta e sete) *features* (e.g. *Protocol, srcport, dstport, pkts* e etc...) disponíveis na *NeTraMark* para a construção do modelo de classificação dos cenários estudados.

4.2.1.1 Cenário 1 O cenário 1 consiste em tráfego HTTP real utilizando as portas TCP padrão. A Tabela 4.2 mostra o resultado de acurácia para cada classificador. Verificou-se inicialmente que o *Precision* apresentado foi de 100% para todos os classificadores. Isso indica que não houveram resultados FP para esta base.

Em relação a *Recall* o classificador *BLINC* apresentou a maior incidência de FN. O resultado indica que somente 20,82% de tráfego da categoria *Web* foi adequadamente identificado. Para o classificador *Naive Bayes* a incidência de FN reduziu porém *Recall*

ainda apresentou resultado de 44,98%, ou seja, abaixo de 50%.

É possível notar uma pequena melhoria de 4,09% entre os classificadores *KNN* e *Bayesian Network*. Para os classificadores *NBKE*, *C.5 Decision Tree*, *Neural Network* e *SVM* há uma melhoria gradativa, porém sutil, na redução de FN nesta ordem. *NBKE* apresentou *Recall* de 55,39% e SVM 57,62%.

O classificador *Coralreef* apresentou somente 58,74% de *Web* adequadamente identificado nesta base. Este desempenho é ruim considerando que o tráfego analisado é real e utiliza as portas TCP padrão dos protocolos. A menor incidência de FN foi observada nos classificadores *Crl_pay* e *Weighted Vote*, ou seja, 92,94% do tráfego *Web* corretamente identificado.

Tabela 4.2: Resultados de classificação para o Cenário 1.

Classificador	*Precision*	*Recall*	*F-Measure*
WEIGHTED VOTE	100%	92,94%	96,34%
CRL_PAY	100%	92,94%	96,34%
CORALREEF	100%	58,74%	74,00%
KNN	100%	57,99%	73,41%
SVM	100%	57,62%	73,11%
NEURAL NETWORK	100%	56,13%	71,90%
C4.5 DECISION TREE	100%	55,76%	71,60%
NBKE	100%	55,39%	71,29%
BAYESIAN NETWORK	100%	53,90%	70,05%
NAIVE BAYES	100%	44,98%	62,05%
BLINC	100%	20,82%	34,46%

Em termos de acurácia para a categoria de aplicação analisada, o classificador *BLINC* obteve o pior resultado, 34,46%. *Naive Bayes* melhorou um pouco o desempenho devido a não incidência de FP, e obteve 62,05%. Há uma melhoria de 8% em relação a este classificador ao utilizar o classificador *Bayesian Network*.

Novamente os classificadores *NBKE*, *C.5 Decision Tree*, *Neural Network* e *SVM* obtiveram acurácia semelhante, com 71,29% para *NBKE* e 73,11% para O classificador *SVM*. A acurácia do classificador *KNN* é levemente superior, 73,41%. O desempenho do classificador *Coralreef* obteve uma acurácia de 74,00% do tráfego classificado corretamente. No entanto, os melhores classificadores para este cenário foram *Crl_pay* e *Weighted Vote* que apresentaram a mesma acurácia, 96,34%.

4.2.1.2 Cenário 2 O cenário 2 consiste em tráfego encriptado através do protocolo SSH utilizando a porta TCP 80. Os resultados de classificação são apresentados na

Tabela 4.3.

Os resultados mostram que a maioria dos classificadores obtiveram acurácia máxima de 100%, na classificação do tráfego. Para estes classificadores não houveram incidências de FP e FN. Isso indica que estes classificadores identificaram corretamente que havia tráfego encriptado na porta TCP 80.

Tabela 4.3: Resultados de classificação para o Cenário 2.

Classificador	*Precision*	*Recall*	*F-Measure*
CRL_PAY	100%	100%	100%
WEIGHTED VOTE	100%	100%	100%
NAIVE BAYES	100%	100%	100%
NBKE	100%	100%	100%
BAYESIAN NETWORK	100%	100%	100%
C4.5 DECISION TREE	100%	100%	100%
KNN	100%	100%	100%
NEURAL NETWORK	100%	100%	100%
SVM	100%	100%	100%
CORALREEF	0%	0%	0%
BLINC	0%	0%	0%

A única exceção são os classificadores *Coralreef* e *BLINC*. No caso destes, houve acurácia de 0%. Para o classificador *Coralreef* houve 100% de incidência de FP mostrando a fragilidade do classificador baseado em portas. O classificador BLINC não identificou corretamente nenhum fluxo de pacotes pois houve 100% de incidência de FN.

4.2.1.3 Cenário 3 O tráfego contido no cenário 3 consiste de pacotes gerados com a aplicação *Google Talk*. O detalhe neste caso, é que o tráfego está encriptado através do protocolo SSH, que utiliza a porta TCP 80 para a conexão com o servidor SSH externo. Os resultados de classificação são apresentados na Tabela 4.4.

Em relação a *Precision* a maioria dos classificadores apresentam 100% de pacotes corretamente identificados. No entanto, o classificador *Coralreef* apresentou 65,22% devido a grande incidência de FP para este cenário. Isso se explica pelo fato de associar a porta TCP 80 à tráfego HTTP, no entanto, neste cenário trata-se de outra aplicação. Ainda, o classificador *BLINC* teve *Precision* de 0% por não ter identificado nenhum TP.

Considerando a incidência de FN pode-se verificar pelos resultados que todos os classificadores apresentaram resultados de Recall abaixo de 50%. Isso indica grande incidência de FN para todos os classificadores. Para os classificadores *SVM*, *C4.5 Decision Tree*, e *Bayesian Network*, o resultado obtido foi de 17,39% para *Recall*.

Tabela 4.4: Resultados de classificação para o Cenário 3.

Classificador	*Precision*	*Recall*	*F-Measure*
NBKE	100%	47,83%	64,71%
NAIVE BAYES	100%	43,48%	60,61%
CORALREEF	65,22%	39,47%	49,18%
CRL_PAY	100%	26,09%	41,38%
WEIGHTED VOTE	100%	26,09%	41,38%
KNN	100%	21,74%	35,71%
NEURAL NETWORK	100%	21,74%	35,71%
BAYESIAN NETWORK	100%	17,39%	29,63%
C4.5 DECISION TREE	100%	17,39%	29,63%
SVM	100%	17,39%	29,63%
BLINC	0%	0%	0%

Houveram resultados iguais de *Recall* para os classificadores *KNN* e *Neural Network*, ou seja, somente 21,74% de tráfego classificado corretamente. Também para os classificadores *Crl_pay* e *Weighted Vote* o resultado se repetiu com 26,09%. *Coralreef* obteve melhoria significativa para *Recall*, devido ao aumento de TP e baixa incidência de FN. Este classificador identificou corretamente 39,47% do tráfego. *Naive Bayes* obteve o segundo melhor resultado para a métrica com 43,48%. No entanto, NBKE foi o que apresentou menor incidência de FN, classificando 47,83 do tráfego existente na base.

Avaliando os resultados finais de acurácia, identificou-se uma variação de 64,71% na acurácia entre o pior classificador, BLINC, e o melhor resultado, o classificador *NBKE*. Verifica-se ainda resultados iguais para os classificadores *SVM*, *C4.5 Decision Tree*, e *Bayesian Network*, com acurácia de 29,63%. Para os classificadores *KNN* e *Neural Network* e acurácia de 35,71%, e para os classificadores *Crl_pay* e *Weighted Vote* com acurácia de 41,38%. *Coralreef* teve desempenho levemente superior a estes ao obter acurácia de 49,18%. No entanto a acurácia só foi superior à 50% nos classificadores *Naive Bayes* e *NBKE*.

4.2.1.4 Cenário 4 O cenário 4 consiste em tráfego P2P real gerado pelo uso do aplicativo *Shareaza*. Os resultados de classificação são apresentados na Tabela 4.5.

Os resultados indicam que para este cenário não houveram incidências de FP. Todos os classificadores apresentaram *Precision* de 100%. Por se tratar somente de tráfego P2P real não existe a possibilidade de incidência de FP para tráfego HTTP.

Em relação a métrica *Recall* houve grande variação na incidência de FN nos classificadores. O classificador *Naive Bayes* apresentou maior número de incidências com

Tabela 4.5: Resultados de classificação para o Cenário 4.

Classificador	*Precision*	*Recall*	*F-Measure*
SVM	100%	97,87%	98,92%
KNN	100%	94,03%	96,92%
NEURAL NETWORK	100%	93,88%	96,85%
CRL_PAY	100%	93,46%	96,62%
WEIGHTED VOTE	100%	93,46%	96,62%
C4.5 DECISION TREE	100%	93,17%	96,47%
BAYESIAN NETWORK	100%	88,62%	93,97%
CORALREEF	100%	80,94%	89,47%
NBKE	100%	78,81%	88,15%
BLINC	100%	71,12%	83,13%
NAIVE BAYES	100%	27,31%	42,91%

Recall de 27,31%. Para os classificadores *BLINC* e *NBKE* já houve uma redução significativa de FN, obtendo 71,12% e 78,81% de tráfego P2P corretamente classificado. O classificador *Coralreef* apresentou desempenho de 80,94% ainda com incidências de FN, justificado pelo fato da aplicação P2P utilizar portas dinâmicas. *Bayesian Network* teve desempenho superior ao *Coralreef* atingindo 88,62% de acertos.

O resultado para *Recall* mostra novamente que *Weighted Vote* e *Crl_Pay* tiveram desempenho igual de 93,46%, ou seja, o mesmo número de incidências de FN e TP. O menor número de incidências de FN, no classificador SVM, determinou o melhor resultado com 97,87% de tráfego P2P corretamente classificado.

Em termos de acurácia, verificamos que a maioria dos classificadores apresentou resultado superior a 80%. A exceção ficou por conta do classificador *Naive Bayes*, que apresentou a pior acurácia com somente 42,91%. Verificou-se ainda que a acurácia apresentada por *Crl_pay* e *Weighted Vote* é a mesma e apresenta resultado de 96,62% para esta base. A acurácia do melhor classificador é levemente superior, ou seja, o classificador SVM obteve acurácia de 98,92%.

4.2.1.5 Cenário 5 O cenário 5 consiste em tráfego P2P gerado pelo uso do aplicativo *Emule*. O detalhe é que este tráfego está encriptado, simulando tráfego HTTP real com conexões de entrada e saída na porta TCP 80. Os resultados de classificação são apresentados na Tabela 4.6.

Os resultados indicam que somente os classificadores *BLINC*, *Coralreef* e *Bayesian Network* obtiveram êxito em classificar algum tráfego contido nesta base. Para todos os outros classificadores houve grande incidência de FP e FN, e nenhuma incidência de TP.

Isso indica que para este tipo de tráfego estes classificadores ora apontam tráfego como sendo HTTP real (FP), ora não conseguem classificar o tráfego corretamente (FN).

Tabela 4.6: Resultados de classificação para o Cenário 5.

Classificador	*Precision*	*Recall*	*F-Measure*
BLINC	58,78%	89,14%	70,84%
CORALREEF	17,50%	13,82%	15,45%
BAYESIAN NETWORK	4,47%	2,94%	3,55%
SVM	0%	0%	0%
KNN	0%	0%	0%
NEURAL NETWORK	0%	0%	0%
CRL_PAY	0%	0%	0%
WEIGHTED VOTE	0%	0%	0%
C4.5 DECISION TREE	0%	0%	0%
NBKE	0%	0%	0%
NAIVE BAYES	0%	0%	0%

Levando-se em consideração somente os 3 (três) classificadores que apresentaram resultados de acurácia, identificou-se que, em relação a incidência de FP, o classificador *Bayesian Network* apresentou a maior quantidade de resultados apontando tráfego HTTP, quando de fato tratava-se de P2P. Para este classificador o *Precision* foi de 4,47%. Em relação ao classificador *Coralreef*, essa incidência diminuiu, mas ainda demonstra-se relevante considerando o resultado de 17,50%. O classificador *BLINC* apresentou a menor incidência de FP, classificando corretamente 58,78% do tráfego P2P.

Em relação a FN, os resultados apresentados demonstraram a mesma tendência. O classificador *Bayesian Network* teve alta incidência de FN e *Recall* de 2,94%. Coralreef apresentou também alta incidência de FN. Com *Recall* de 13,82% mostra que a incidência de FN foi ainda superior a de FP. O classificador *BLINC* teve baixa incidência de FN, demonstrando melhor desempenho com 89,14% do tráfego corretamente classificado.

Avaliando a acurácia dos classificadores neste cenário, verificou-se que, em virtude dos resultados analisados acima, o classificador *Coralreef* obteve baixa acurácia com apenas 15,45%. Isso deve-se à grande incidência de FP para esta base. Por fim, o classificador *BLINC* apresentou-se mais adequado para identificar este tipo de tráfego com acurácia de 70,84%.

4.2.1.6 Cenário 6 O cenário 6 consiste em tráfego do jogo *World of Warcraft*. O detalhe é que este tráfego está encriptado, simulando tráfego HTTP real com conexões de entrada e saída na porta TCP 80.

Os resultados de classificação obtidos para esta cenário foram nulos, ou seja, todas as métricas apresentaram valor 0. Isso indica para todos os classificadores utilizados grande incidência de FP e FN, e nenhuma incidência de TP. Nenhum classificador foi capaz de identificar corretamente o tráfego. Todos os classificadores apresentaram grande quantidade de resultados incorretos, apontando tráfego HTTP, quando de fato tratava-se de tráfego do jogo.

4.2.1.7 Cenário 7 Conforme descrito na Seção 4.1.2, a partir do cenário 7, o tráfego HTTP real foi combinado com tráfego não HTTP camuflado através da porta TCP 80. O cenário 7 consiste em tráfego HTTP real utilizando a porta TCP padrão. Ainda nesta cenário, existe tráfego encriptado do protocolo SSH, que utiliza a porta TCP 80 para conexões de entrada e saída. Os resultados de classificação são apresentados na Tabela 4.7.

Tabela 4.7: Resultados de classificação para o Cenário 7.

Classificador	*Precision*	*Recall*	*F-Measure*
CORALREEF	91,76%	100,00%	95,71%
SVM	99,31%	85,21%	91,72%
NBKE	96,69%	70,48%	81,53%
CRL_PAY	100%	69,41%	81,49%
WEIGHTED VOTE	100%	69,41%	81,49%
KNN	100%	67,65%	80,70%
NEURAL NETWORK	99,08%	63,91%	77,70%
C4.5 DECISION TREE	97,20%	62,28%	75,91%
NAIVE BAYES	98,11%	61,90%	75,91%
BAYESIAN NETWORK	100%	55,29%	71,21%
BLINC	100%	25,88%	41,12%

Analisando os resultados de classificação obtidos, verificou-se no que concerne à incidência de FP o classificador *Coralreef* apresentou o pior resultado, com *Precision* em 91,76%. Esse resultado indica a incidência de FP, demonstrando que este classificador falha na identificação de outro tipo de tráfego configurado na porta TCP 80.

Para os classificadores *BLINC*, *Bayesian Network*, *KNN*, *Crl_pay* e *Weighted Vote*, não houveram incidências de FP, demonstrado pelo resultado de 100% para *Precision*. É ainda relevante verificar que, o classificador *SVM* obteve o segundo melhor resultado, identificado corretamente 99,31% do tráfego com baixa incidência de FP.

Em termos de FN, verificou-se que o classificador *Coralreef* obteve o melhor resultado, com 100% para a métrica *Recall*. Isso demonstra que para esse classificador houve somente

incidência de TP ou FP para tráfego HTTP. Não houveram incidências de FN para esse classificador. O pior resultado foi do classificador *BLINC* com *Recall* de 25,88%. O classificador *SVM* obteve o segundo melhor resultado, com 85,21% indicando que houve baixa incidência de FN na classificação.

Avaliando a acurácia dos classificadores, verificou-se que *BLINC* obteve o pior resultado com 41,12% de acurácia. Para esta cenário, assim como os outros até aqui analisados, *Weighted Vote* e *Crl_pay* tiveram a mesma acurácia de 81,94%. *SVM* obteve boa acurácia de 91,72%, perdendo somente para o classificador *Coralreef* (95,71%).

4.2.1.8 Cenário 8 O cenário 8 consiste em tráfego HTTP real utilizando as portas TCP padrão. Ainda nesta cenário existe tráfego do aplicativo *Google Talk*. Este tráfego está encriptado através do protocolo SSH com conexões ao servidor SSH pela porta TCP 80. Os resultados de classificação são apresentados na Tabela 4.8.

Tabela 4.8: Resultados de classificação para o Cenário 8.

Classificador	*Precision*	*Recall*	*F-Measure*
CORALREEF	89,42%	85,32%	87,32%
SVM	94,48%	66,67%	78,17%
CRL_PAY	94,03%	54,31%	68,85%
WEIGHTED VOTE	94,03%	54,31%	68,85%
KNN	94,70%	53,65%	68,49%
NEURAL NETWORK	96,80%	51,27%	67,04%
C4.5 DECISION TREE	96,75%	50,42%	66,30%
BAYESIAN NETWORK	95,87%	49,36%	65,17%
NBKE	96,08%	41,53%	57,99%
BLINC	100%	34,17%	50,93%
NAIVE BAYES	100%	14,58%	25,45%

Para este cenário identificou-se em geral baixa incidência de FP. O pior resultado foi obtido no classificador *Coralreef*, indicando que este classificador determinou grande quantidade de FP para HTTP nos resultados de classificação. Os resultados mostram ainda que os classificadores *BLINC* e *Naive Bayes* não tiveram incidência de FP, obtendo o melhor resultado com 100% de acertos no tráfego que conseguiram identificar. O classificador *Neural Network* obteve o segundo melhor resultado com 96,80%.

Em geral houve grande incidência de FN na maioria dos classificadores. O pior resultado obtido para a métrica *Recall* foi do classificador *Naive Bayes* com 14,58%. Para o classificador *Neural Network* o resultado apresentado é de 51,27%. Novamente para os

classificadores *Crl_pay* e *Weighted Vote* o resultado obtido foi igual entre ambos, indicando *Recall* de 54,31%. Para estes classificadores ainda houve incidência relevante de FN. O classificador *SVM* apresentou o segundo melhor resultado, com baixa incidência de FN na base. O *Recall* para este classificador foi de 66,67%. Analisando a incidência de FN o classificador *Coralreef* obteve o melhor resultado(85,23%) representando poucos casos de FN.

Avaliando a acurácia dos classificadores para esta cenário pode-se observar que, o pior resultado obtido foi para *Naive Bayes* com somente 25,45%. O classificador *Crl_pay*, assim como *Weighted Vote*, tiveram acurácia de 68,85%. Este resultado demonstra que para estes classificadores a incidência de FN é relevante no resultado. Considerando o resultado do classificador *SVM*, conclui-se que este apresentou boa acurácia com 78,17%. Isso deve-se pela menor incidência de FN em relação ao outros classificadores com acurácia inferior. A melhor acurácia obtida para esta cenário foi do classificador *Coralreef* com 87,32%.

4.2.1.9 Cenário 9 O cenário 9 consiste em tráfego HTTP real utilizando a porta TCP padrão. Ainda neste cenário, existe tráfego do jogo *WOW*. Este tráfego está encriptado através do protocolo HTTP com conexões ao servidor externo pela porta TCP 80. Os resultados de classificação são apresentados na Tabela 4.9.

Considerando os resultados apresentados para a métrica *Precision* podemos verificar que há em todos os classificadores grande incidência de FP. Não há classificador com resultado maior que 15%. O pior resultado está nos classificadores *Crl_pay* e *Weighted Vote* que apresentam o mesmo percentual de 9,20%. *SVM* apresenta o segundo melhor resultado com 12,96%. O melhor resultado é obtido para o classificador *Coralreef* com 13,10% de tráfego HTTP corretamente identificado.

Observou-se também que o desempenho dos classificadores melhorou, quando se trata de FN. Há uma menor incidência destes para os classificadores. No entanto, o classificador *Naive Bayes* ainda apresentou baixo desempenho com 0,99%, indicando praticamente todos os resultados como FN. Para os classificadores *Crl_pay* e *Weighted Vote* nota-se uma melhoria acentuada com o resultado de 67,24% para ambos. Este desempenho só foi superado pelo classificador *SVM* com 76,75% e pelo classificador *Coralreef*, que apresentou o melhor resultado para *Recall* com 100% de acertos não havendo incidência de FN.

Considerando a acurácia, verificamos que em geral para este cenário o melhor resultado não superou 30% de acurácia para a base. No geral, os classificadores mostraram-se

Tabela 4.9: Resultados de classificação para o Cenário 9.

Classificador	*Precision*	*Recall*	*F-Measure*
CORALREEF	13,10%	100,00%	23,16%
SVM	12,96%	76,75%	22,18%
KNN	13,02%	64,26%	21,65%
C4.5 DECISION TREE	12,95%	64,64%	21,58%
NEURAL NETWORK	12,97%	64,10%	21,58%
BLINC	12,93%	57,94%	21,15%
NBKE	12,90%	57,05%	21,04%
BAYESIAN NETWORK	12,76%	49,80%	20,32%
CRL_PAY	9,20%	67,24%	16,19%
WEIGHTED VOTE	9,20%	67,24%	16,19%
NAIVE BAYES	11,17%	0,99%	1,82%

ineficazes ao distinguir tráfego HTTP real do tráfego HTTP camuflado. Contudo, os classificadores *SVM* com acurácia de 22,18% e *Coralreef* com 23,16% apresentaram a melhor acurácia para este cenário.

4.2.1.10 Cenário 10 O cenário 10 consiste em tráfego HTTP real utilizando a porta TCP padrão. Ainda nesta base, existe tráfego do P2P gerado pelo aplicativo *Emule*. O detalhe é que este tráfego está tunelado através do protocolo HTTP estabelecendo à conexão pela porta TCP 80. Os resultados de classificação são apresentados na Tabela 4.10.

Para este tráfego, pode-se observar que no geral, os classificadores obtiveram melhores resultados do que no cenário anterior. Verificou-se uma diminuição acentuada na incidência de FP e FN. No que diz respeito à *Precision* os classificadores com melhor resultado foram *Naive Bayes* com 78,43% e *SVM* com 79,02%. *Weighted Vote* e *Crl_pay* apresentaram o pior resultado com alta incidência de FP e resultado de 55,29% para ambos.

Ao analisar a incidência de FN, identificou-se que novamente *Crl_pay* e *Weighted Vote* obtiveram o pior desempenho com 27,98% para *Recall*. Este desempenho já apresenta resultado maior que 50% no classificador *NBKE* (52,69%). No entanto, os melhores resultados são novamente dos classificadores *Naive Bayes*, com 69,39% e *SVM* com 86,57%.

Em termos de acurácia, observou-se que o resultado é razoavelmente bom para a maioria dos classificadores. Ou seja, há acurácia maior que 60% em 8 (oito) classificadores. O pior resultado, no entanto, é novamente dos classificadores *Crl_pay* e *Weighted Vote* que apresentaram acurácia de 37,15%. Os melhores resultados de acurácia obtidos para este cenário estão nos classificadores *Naive Bayes* (73,62%) e *SVM* (82,62%).

Tabela 4.10: Resultados de classificação para o Cenário 10.

Classificador	*Precision*	*Recall*	*F-Measure*
SVM	79,02%	86,57%	82,62%
NAIVE BAYES	78,43%	69,36%	73,62%
NEURAL NETWORK	75,00%	72,29%	73,62%
C4.5 DECISION TREE	72,14%	60,48%	65,80%
BAYESIAN NETWORK	71,68%	60,06%	65,36%
KNN	71,94%	59,88%	65,36%
CORALREEF	71,90%	58,81%	64,70%
BLINC	76,68%	54,96%	64,03%
NBKE	69,29%	52,69%	59,86%
CRL_PAY	55,29%	27,98%	37,15%
WEIGHTED VOTE	55,29%	27,98%	37,15%

4.2.1.11 Conclusões sobre a Avaliação de Acurácia dos Cenários *Ground Truth*. A Tabela 4.11 sumariza os classificadores que apresentaram maior acurácia para os cenários *ground truth* analisadas.

Para o cenário 1, os classificadores *Crl_pay* e *Weighted Vote* apresentaram *Precision* em 100%. Não houve incidência de FP e houve baixa incidência de FN. A acurácia atingiu 96,33%. Os mesmos classificadores apresentaram para o cenário com tráfego SSH camuflado na porta TCP 80, 100% de acurácia. Apesar da grande incidência de FN, o classificador *NBKE* obteve êxito em 64,70% do tráfego de *Chat* encriptado por SSH na porta TCP 80.

Para o tráfego do cenário 4 contendo P2P real, o classificador *SVM* identificou 98,92% dos pacotes corretamente, no entanto, o *BLINC* mostrou-se melhor (acurácia de 70,84%), para este tráfego quando encriptado pelo protocolo HTTP na porta TCP 80.

Não houveram resultados de acurácia para o cenário 6 pois todos os classificadores não apresentaram TP para o tráfego do jogo *WOW*. Nesta base os classificadores identificaram ainda grande quantidade de FP para tráfego HTTP.

Para o cenário 7, os classificadores *Coralreef* (acurácia 95,70%) e *SVM* (91,71%), apresentaram a maior capacidade para classificar o tráfego corretamente, mesmo havendo pacotes de aplicações diferentes. Este desempenho se repete para os cenários 8 e 9.

No cenário 8, o *SVM* é inferior ao *Coralreef* por apresentar maior quantidade de FN (*Recall* de 66,66%). No entanto, teve melhor desempenho na incidência de FP. O classificador *SVM* apontou acurácia de 78,17%.

No cenário 9, a quantidade de FP é significativa para *SVM* e *Coralreef* indicando que apontaram incorretamente pacotes da categoria *Games* como sendo pacotes HTTP.

Tabela 4.11: Melhores Classificadores Para os Cenários *Ground Truth*.

Cenários	Classificador	Categorias	*Precision*	*Recall*	*F-Measure*
Cenário 1	***Crl_pay/ Weighted Vote***	**Web**	**100%**	**92,93%**	**96,33%**
Cenário 2	*Crl_pay/Weighted Vote*	Encryption	100%	100%	100%
Cenário 3	*NBKE*	Chat	100%	47,82%	64,70%
Cenário 4	*SVM*	P2P	100%	97,86%	98,92%
Cenário 5	*Blinc*	P2P	58,77%	89,14%	70,84%
Cenário 6	N/A	Games	0%	0%	0%
Cenário 7	***Coralreef***	**Web/ Encryption**	**91,76%**	**100%**	**95,70%**
Cenário 7	***SVM***	**Web/ Encryption**	**99,31%**	**85,20%**	**91,71%**
Cenário 8	***Coralreef***	**Web/Chat**	**89,42%**	**85,32%**	**87,32%**
Cenário 8	***SVM***	**Web/Chat**	**94.47%**	**66,66%**	**78,17%**
Cenário 9	***Coralreef***	**Web/Games**	**13,09%**	**100%**	**23,16%**
Cenário 9	***SVM***	**Web/Games**	**12,96%**	**76,75%**	**22,17%**
Cenário 10	***Naive Bayes***	**Web/P2P**	**78,43%**	**69,36%**	**73,61%**
Cenário 10	***SVM***	**Web/P2P**	**79,01%**	**86,56%**	**82,62%**

Para este cenário há baixa incidência de FN e a acurácia demonstrada apesar de baixa (respectivamente 23,16% e 22,17%) foi o melhor resultado obtido.

Para o cenário 10, *Naive Bayes* e *SVM* mostraram maior acurácia (respectivamente 73,61% e 82,62%) para identificar tráfego Web e P2P combinado. Este resultado não se repetiu ao classificar P2P isoladamente nos cenários 4 e 5.

O estudo da acurácia dos classificadores nos cenários de tráfego controlado permitiu algumas conclusões sobre os resultados de classificação obtidos:

- O classificador *Weighted Vote* com atribuição padrão de pesos e *ground truth*, apresenta sempre a mesma acurácia do classificador *Crl_pay*;
- Nos cenários onde havia tráfego não HTTP camuflado na porta TCP 80, bases 2, 3, 4, 5 e 6, houveram alguns classificadores com boa acurácia para identificar este tipo de tráfego;
- No entanto, nos cenários 7, 8, 9 e 10, ao combinar este tráfego com tráfego HTTP real, o resultado não se confirmou, e os classificadores *Coralreef*, *SVM* e *Naive Bayes* obtiveram maior acurácia;
- Nos cenários 7, 8 e 9 o classificador *Coralreef* apresentou sempre maior acurácia;

- No entanto, O classificador *Coralreef* identificou grande quantidade de FP nos cenários 7, 8, 9 e 10. Para estes cenários, os classificadores *SVM* e *Naive Bayes* mostraram menor incidência de FP.

4.3 O CLASSIFICADOR PROPOSTO: F-MEASURE BASED CLASSIFIER(FBC)

Considerando os resultados de classificação da Seção 4.2.1, e as conclusões extraídas apartir destes resultados, pode-se formatar a proposta para o classificador FBC. Para determinar os componentes do classificador os critérios adotados foram:

- Classificadores que apresentaram maior acurácia para os cenários que contém tráfego HTTP (Cenários 1, 7, 8, 9 e 10)[2];
- Segundo melhor resultado de acurácia nos cenários que contém tráfego HTTP real e tráfego não HTTP camuflado na porta TCP 80 (Cenários 7, 8, 9 e 10). Este critério, visa maximizar a acurácia através da redução de incidência de FP para tráfego HTTP;
- Pesos baseados no número de incidências de um classificador dados os critérios acima.

Os classificadores selecionados de acordo com estes critérios e seus respectivos pesos estão descritos na Tabela 4.12.

Tabela 4.12: Classificadores selecionados para **FBC**.

Classificador	***Peso***
Crl_pay	1
Naive Bayes	1
Coralreef	3
SVM	4

4.3.1 Acurácia Ponderada (APC)

Com o objetivo de maximizar a influência dos melhores classificadores na acurácia de classificação de FBC, utilizou-se a média ponderada. Para a ponderação, FBC utiliza os classificadores selecionados na seção anterior e seus respectivos pesos genéricos. O valor a ser ponderado é a acurácia média, ou seja, o valor da métrica *F-Measure* médio de

[2]Os resultados dos outros cenários foram descartados pois o escopo do estudo está na classificação de tráfego HTTP.

cada classificador. Este valor é obtido então a partir da média aritmética da métrica *F-Measure* do classificador entre os cenários onde foi selecionado. A *acurácia ponderada* (APC) para FBC é determinada pela Equação (4.1).

$$\frac{((\alpha.\lambda_1) + (\beta.\lambda_2) + (\gamma.\lambda_3) + (\delta.\lambda_4))}{\sum_{i=1}^{n} \lambda_i}, \tag{4.1}$$

Onde:

- α = F-Measure médio do classificador *Crl_pay*;
- β = F-Measure médio do classificador *Naive Bayes*;
- γ = F-Measure médio do classificador *Coralreef*;
- δ = F-Measure médio do classificador *SVM*;
- $\lambda_i(pesos) = \{1, 1, 3, 4\}$;
- $n = \{1, 2, 3, 4\}$.

4.3.2 Descrição do Algoritmo

O algoritmo 1 detalha como é realizada a classificação de tráfego por FBC. Este algoritmo foi implementado e incorporado ao código fonte da ferramenta *NeTraMark*. Tal implementação deu-se através de *scripts* SQL e adequações em Java ao código fonte da ferramenta.

Algorithm 1: Pseudocódigo da implementação de FBC

```
1  .
   Data: ResultadoCrl_pay, ResultadoNaiveBayes, ResultadoCoralreef,
         ResultadoSVM,ResultadoFBC,lim_F,fluxo,APC
   Result: ResultadoFBC
2  α = 0,9633  β = 0,7361  γ = 0,6872  δ = 0,6866;
3  Read lim_F;
4  while Não é o fim do trace do
5      Read fluxo;
6      if ResultadoCrl_pay≠'Web' then α = 0;
7      if ResultadoNaiveBayes≠'Web' then β = 0;
8      if ResultadoCoralreef≠'Web' then γ = 0;
9      if ResultadoSVM≠'Web' then δ = 0;
10     APC=((α+β+(γ.3)+(δ.4))/9);
11     if APC≥lim_F then
12         ResultadoFBC =' Web';
13     else
14         ResultadoFBC =' Unknown';
15     end
16 end
```

Na linha 1 as constantes α, β, γ e δ são declaradas recebendo o valor de acurácia média dos respectivos classificadores selecionados. Na linha 2, o algoritmo parametriza *Limite F-Measure* ($\lim_F$) de acordo com a entrada do operador. O parâmetro ($\lim_F$) representa o valor mínimo de APC para que FBC possa classificar um fluxo de pacotes como *Web*.

É estabelecido um *loop* para que todo o *trace* seja analisado (linhas 3 a 15).Os resultados de classificação são obtidos para cada fluxo de pacotes (linha 4). Nas linhas de 5 a 8, é testado o resultado de classificação para categoria *Web*. Caso o classificador testado não apresente este resultado o valor de acurácia média correspondente recebe o valor 0 (Não identificou o fluxo como sendo *Web*).

Em seguida APC é calculada com base nos valores de acurácia média e os respectivos pesos para cada classificador. O resultado de classificação para FBC é determinado pelo teste condicional descrito na linha 10. Valores de APC maiores ou iguais a $\lim_F$ determinam que o fluxo analisado é *Web*, do contrário o classificador retorna o valor *Unknown* (desconhecido).

Um estudo sobre o impacto de $\lim_F$ sobre a acurácia de FBC é apresentado a seguir na Seção 4.3.3.

4.3.3 Impacto do Parâmetro Limite F-Measure ($\lim_F$)

Nesta avaliação foram utilizadas as bases selecionadas na Seção 4.3. A figura 4.2 mostra o impacto de $\lim_F$ na melhoria dos resultados de classificação de FBC em relação ao *Weighted Vote* com configuração padrão.

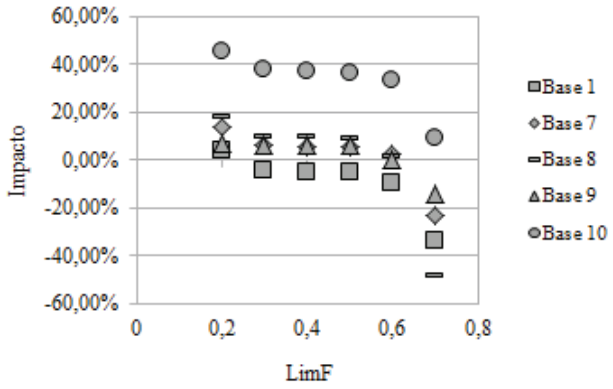


Figura 4.2: Impacto de $\lim_F$ sobre a acurácia de *FBC* X *Weighted Vote*.

Com base nos valores máximo e mínimo de *acurácia ponderada*(APC), foi determinada uma faixa de valores para $\lim_F$ com o objetivo de investigar o impacto da variação de $\lim_F$ na acurácia de FBC. Os valores utilizados estiveram entre 0,2 e 0,7.

A métrica *F-Measure* foi avaliada nos cenários comparando o resultado de FBC em relação à *Weighted Vote* para cada valor de $\lim_F$. Verificou-se que a tendência é de decréscimo na melhoria ao incrementar o valor de $\lim_F$, tornando-o inversamente proporcional a acurácia de FBC nesta análise.

Com $\lim_F = 0,7$ somente o cenário 10 apresenta melhoria de FBC (em todos os outros cenários há uma piora de FBC em relação à *Weighted Vote*). Para $\lim_F = 0,6$ há melhoria em três cenários. O resultado é abaixo de 3% para os cenários 7 e 8, e percentual aproximado de 40% para o cenário 10.

Nos cenários 7, 8, 9 e 10 há melhoria semelhante de FBC para $\lim_F = 0,4$ e $\lim_F = 0,5$. A acurácia é levemente superior para os mesmos cenários para $\lim_F = 0,3$. Para estes valores do parâmetro não há melhoria de FBC para o cenário 1.

Para $\lim_F = 0,2$, há melhoria de FBC em relação ao classificador *Weighted Vote*

para todas os cenários classificados. Portanto, este valor foi utilizado para as avaliações realizadas a seguir.

4.4 A METODOLOGIA PROPOSTA

Sumarizando as etapas descritas anteriormente, pode-se descrever a metodologia proposta para classificação de tráfego HTTP na *NeTraMark* como uma sequência de procedimentos a serem utilizados para obtenção de uma classificação de tráfego sistematizada. Adotando estes procedimentos o administrador da rede não necessitará de conhecimento prévio do tráfego da rede ao utilizar a *NeTraMark* na solução de problemas de classificação ou na identificação de tráfego não autorizado na rede. Estes procedimentos são descritos a seguir:

- **Delimitar o problema de classificação.** Torna-se necessário delimitar qual problema será atacado restringindo-o à aplicações e situações de tráfego específicos na rede;
- **Criar cenários *Ground Truth* em ambiente controlado.** Estes cenários são necessários para que o administrador possa verificar os resultados dos classificadores da *NeTraMark* comparando-os à uma base confiável de resultados de classificação. Os dados de tráfego devem conter pacotes específicos do problema a ser estudado. Devem também ser filtrados para conter somente pacotes relacionados aos *hosts* do ambiente controlado;
- **Selecionar os classificadores com melhores resultados.** O administrador deverá então selecionar os classificadores que apresentaram melhores resultados, ou seja, maior acurácia (métrica *F-Measure*) para cada cenário *Ground Truth* estudado;
- **Determinar pesos para cada classificador selecionado.** Os pesos devem ser determinados com base no número de incidência de cada classificador selecionado considerando o conjunto de cenários *Ground Truth* estudados. Exemplo: Classificador A foi o de maior acurácia em três cenários *Ground Truth* então terá peso 3;
- **Obter o classificador FBC utilizando o algoritmo descrito na Seção 4.3.2.** Para tal deve-se calcular *F-Measure* médio utilizando o resultado de *F-Measure* entre as bases onde o classificador foi selecionado. Deve-se ainda

Calcular APC (Acurácia Ponderada) aplicando os pesos ao *F-Measure* médio respectivo de cada classificador selecionado;

- **Identificar o valor ótimo para $\lim_F$ nos cenários estudados.** O administrador deve realizar um estudo comparativo de acurácia entre FBC e *Weighted Vote* (configuração padrão) para determinar o valor ideal de $\lim_F$. Determina-se este valor identificando o valor de $\lim_F$ onde há a maior quantidade de cenários *Ground Truth* com melhoria na acurácia de FBC em relação à *Weighted Vote*.

Adicionalmente, pode-se realizar estudos de casos para validar os resultados de FBC. Utilizando os classificadores selecionados, seus respectivos pesos e o valor de $\lim_F$, pode-se aplicar FBC para validações da metodologia proposta e para classificação de tráfego desconhecido (sem a presença de *Ground Truth*).

4.5 ANÁLISE COMPARATIVA DOS CLASSIFICADORES

Esta seção analisa a acurácia do classificador proposto em relação aos classificadores com melhor desempenho nos cenários *ground truth* e em cenários não controladas. O objetivo é verificar sua acurácia e aplicabilidade em situações de classificação de tráfego, considerando que não há conhecimento sobre as aplicações que estão gerando tráfego na rede.

4.5.1 Avaliação de FBC para os Cenários Ground Truth

Esta seção apresenta a avaliação de FBC em relação ao classificadores que obtiveram maior acurácia nas bases selecionadas na Seção 4.3.

A Tabela 4.13 mostra os resultados de acurácia para cada cenário de acordo com a métrica *F-Measure*. São descritos os valores de *F-Measure* para o classificador FBC e para o classificador da *NeTraMark* que apresentou a maior acurácia na classificação do tráfego.

Tabela 4.13: Acurácia do Classificador FBC Para os Cenários *Ground Truth*.

Cenário #	*FBC*	*Crl_pay/Weighted Vote*	*Coralreef*	*SVM*
1	100%	96,34%	-	-
7	96,75%	-	95,71%	-
8	87,32%	-	87,32%	-
9	26,08%	-	23,16%	-
10	82,62%	-	-	82,62%

Para o cenário 1, identificou-se que o classificador com maior acurácia foi o classificador *Weighted Vote*. É possível verificar que, neste cenário, o classificador FBC obteve maior acurácia com melhoria de 3,66%.

Considerando os resultados de classificação do cenário 7 verifica-se que FBC apresenta a maior acurácia (96,75%). FBC obteve melhoria de 1,04% em relação ao classificador *Coralreef*, apontado na avaliação realizada na Seção 4.2.1 como melhor classificador para esta base.

No cenário 8, O resultado de FBC se igualou ao classificador *Coralreef*. FBC obteve 87,32% de acurácia. Avaliando os resultados para o cenário 9, o classificador proposto obtém uma melhoria de 2,92% na acurácia em relação ao classificador *Coralreef*.

A acurácia de FBC no cenário 10 é equivalente a acurácia do classificador *SVM* (82,62%).

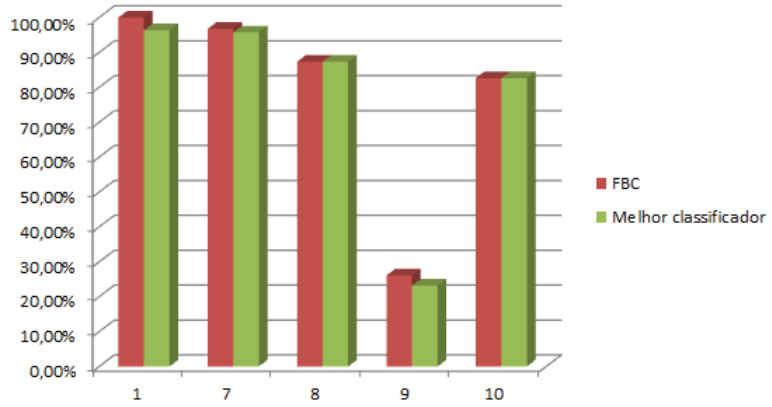


Figura 4.3: Acurácia de FBC em relação aos classificadores da *NeTraMark*

Pode-se observar no Gráfico 4.3 que em relação aos cenários *ground truth* o classificador apresenta uma tendência de melhoria para três bases. Essa tendência foi até 2,92% maior em relação ao classificador de maior acurácia. Para os outros dois cenários a acurácia foi igual.

4.6 AVALIAÇÃO DE FBC(F-MEASURE BASED CLASSIFIER) PARA OUTROS CENÁRIOS

Esta seção apresenta uma avaliação de acurácia do classificador proposto FBC em relação ao classificadores implementados na *NeTraMark* em relação a outros cenários. O resultado de classificação para *Weighted Vote* foi obtido com configuração padrão. Os cenários

utilizados [3] são descritas nas seções correspondentes.

4.6.1 Avaliação de Acurácia para o Cenário HTTP_P2P_GT

Para esta avaliação, o objetivo é analisar a acurácia de FBC para identificar fluxos de pacote da categoria de aplicação *Web* em relação aos outros classificadores da *NeTraMark*. O objetivo é determinar se o classificador proposto pode executar o processo de classificação deste tráfego com bom nível de acurácia descartando a necessidade de uma atribuição empírica de pesos para *Weighted Vote*.

O arquivo de captura que usamos neste avaliação é composto de tráfego com fluxos de pacotes HTTP, P2P e SSH. O tráfego SSH foi camuflado na porta TCP 80. A base de dados de tráfego foi gerada em ambiente controlado o que possibilitou a determinação do *ground truth*.

A captura dos pacotes foi executada através da ferramenta *Wireshark* durante o intervalo de 1(uma) hora em Agosto de 2012. Foram capturados 1048858 pacotes e o tamanho total do arquivo de captura é de 984MB. A base de dados de tráfego foi nomeada **HTTP_P2P_GT**.

Tabela 4.14: Acurácia de classificação para Web no Cenário **HTTP_P2P_GT**.

Classificador	*Precision*	*Recall*	*F-Measure*
Coralreef	**94,98%**	**98,27%**	**96,60%**
FBC	**94,98%**	**98,27%**	**96,60%**
Crl_pay	**100%**	**53,50%**	**69,71%**
Weighted Vote	**100%**	**53,50%**	**69,71%**
Bayesian Network	100%	10,29%	18,66%
Neural Network	100%	9,88%	17,98%
KNN	100%	9,47%	17,29%
SVM	100%	8,64%	15,91%
Naive Bayes	100%	8,23%	15,21%
NBKE	100%	7,82%	14,50%
C4.5 Decision Tree	100%	4,95%	9,44%
BLINC	100%	0,82%	1,63%

Observa-se pelos resultados da Tabela 4.14 que a maioria dos classificadores apresentaram *Precision* de 100%. Isso indica que não identificaram FP para a tráfego HTTP. A quantidade de FN foi determinante para a variação de acurácia de todos os classificado-

[3] As bases de tráfego relacionadas a cada cenário estão disponíveis através do endereço http://www.cin.ufpe.br/~pasg/bases-cacp2/

res. É possivel identificar que o classificador *BLINC* apresentou a maior quantidade de pacotes classificadores incorretamente com *Recall* de 0,86%. Os classificadores baseados em aprendizagem de máquina também tiveram baixo percentual na detecção de tráfego HTTP, obtendo o melhor resultado no classificador *Bayesian Network*. Este classificador apresentou *Recall* de 10,29%.

Neste cenário verificamos que *Weighted Vote* apresenta o mesmo resultado de acurácia (69,71%) que o classificador *Crl_Pay*. Esse resultado corrobora o que já havia sido identificado na análise realizada na Seção 4.2.1.

O classificador *Coralreef* apresenta a maior acurácia para o cenário estudado identificando corretamente 96,60% do tráfego HTTP. Para este classificador a métrica *Precision* indica a incidência de FP ao classificar tráfego camuflado na porta TCP 80. Para o classificador FBC a acurácia obtida é também de 96,60%. A acurácia de FBC neste cenário é 26,89% superior a acurácia de *Weighted Vote*.

4.6.2 Avaliação percentual para o Cenário SAMPLE_ENTERPRISE

Para esta avaliação o objetivo é analisar o capacidade de FBC na detecção de tráfego HTTP em relação aos outros classificadores da *NeTraMark*. O objetivo é determinar se FBC pode detectar um percentual maior de fluxo de pacotes já que não há conhecimento do *ground truth* neste cenário.

Nesta avaliação utilizamos bases de dados de tráfego disponibilizadas no portal *openpacket* [4] contendo amostras de tráfego comumente encontrado em uma rede empresarial[5]. As bases foram concatenadas em uma única base nomeada **SAMPLE_ENTERPRISE** que contém 6661 pacotes com tráfego DNS, HTTP, NTP, SNMP dentre outros.

[4] Acessível em https://www.openpacket.org/

[5] Arquivos de captura example.com-1.pcap, example.com-4.pcap, example.com-5.pcap e example.com-6.pcap

Tabela 4.15: Desempenho dos classificadores para o Cenário **SAMPLE_ENTERPRISE**

Classificador	*Fluxos*	*%*
FBC	**157**	**25,70%**
SVM	138	22,59%
Neural Network	125	20,46%
Bayesian Network	116	18,99%
KNN	111	18,17%
C4.5 Decision Tree	110	18,00%
Crl_pay	106	17,35%
Weighted Vote	106	17,35%
Coralreef	97	15,88%
NBKE	76	12,44%
Naive Bayes	72	11,78%
BLINC	0	0%

O cenário analisado possui no total 611 fluxos de pacotes. O classificador *BLINC* apresentou o pior desempenho não identificando tráfego Web na base. Esse desempenho de *BLINC* deve-se ao fato de que a análise foi feita numa base anonimizada. Em bases anonimizadas as informações relativas ao *host* e suas interações com outros *hosts* na rede não estão disponíveis. Conforme explicado anteriormente este classificador utiliza estas informações para determinar o resultado de classificação. *NBKE* e *Naive Bayes* tiveram desempenho abaixo de 15% identificando somente 12,44% e 11,78% respectivamente.

O classificador *Coralreef* identificou 97 fluxos, representando 15,88%. Novamente o resultado apresentado para *Weighted Vote* se iguala ao do classificador *Crl_pay*, identificando tráfego HTTP em somente 17,35% do total de fluxos de pacotes.

Ainda abaixo de 20% estão os classificadores *C4.5 Decision Tree*, *KNN* e *Bayesian Network*. O classificador *Neural Network* obteve o terceiro melhor desempenho (20,46%) precedido por *SVM* (22,59%).

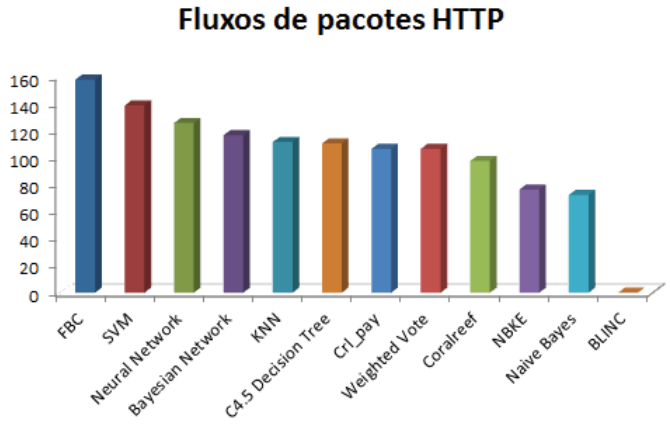


Figura 4.4: Fluxos de pacotes HTTP detectados no cenário SAMPLE_ENTERPRISE

De acordo com o Gráfico 4.4 o classificador proposto FBC obteve o melhor desempenho identificando 157 fluxos de pacotes. Este tráfego representa 25,70% do total de fluxos existentes na base. O resultado mostra que o classificador FBC sugere uma quantidade maior de fluxos HTTP. No entanto, vale ressaltar que como não há *ground truth* não houve possibilidade de avaliar a acurácia destes resultados.

RESUMO

Este capítulo descreveu a metodologia utilizada para definição do classificador proposto FBC. Foi realizada uma avaliação de acurácia dos classificadores implementados na *NeTraMark* em cenários de tráfego HTTP real e tráfego não HTTP camuflado pela porta TCP 80. Esta avaliação indicou que determinados classificadores conseguem identificar tráfego não HTTP com boa acurácia, no entanto, ao se combinar o tráfego com HTTP real o resultado de acurácia obtido é diferente. Além disto, o classificador *Weighted Vote* configurado com pesos e *ground truth* padrão, não mostrou-se com boa acurácia na maioria dos cenários analisados. A partir dos resultados de acurácia e das conclusões extraídas na avaliação realizada, o classificador foi definido com base em pesos genéricos e classificadores que apresentaram melhor desempenho na identificação de tráfego HTTP. O classificador proposto foi aplicado nas cenários *ground truth* e em cenários disponilizados sem o *ground truth*. Os resultados apresentados mostram que o classificador proposto, obtém acurácia igual ou superior a todos os classificadores implementados na *NeTraMark*. Mostram ainda que é capaz de sugerir um percentual maior de fluxos de pacotes HTTP para tráfego desconhecido.

CAPÍTULO 5

CONCLUSÕES

Classificação de tráfego HTTP real é um problema bastante recorrente para administradores de rede. Em virtude da existência de técnicas de camuflagem de tráfego não permitido através da porta TCP 80 torna-se um desafio identificar se determinado *host* está utilizando aplicações permitidas ou burlando as políticas de segurança da rede.

Diversas abordagens de classificação de tráfego foram desenvolvidas pela comunidade científica. Dentre elas, [Lee et al., 2011] desenvolveu um *benchmark* para comparação de técnicas denominado *NeTraMark*. A ferramenta *NeTraMark* disponibiliza 10 (dez) classificadores do estado da arte em classificação de tráfego de rede e um novo classificador chamado *Weighted Vote*. O resultado deste classificador é obtido da resultante de duas regras: Votação de pesos baseados na maioria e a acurácia dos classificadores em relação ao *ground truth*.

Em ambientes de rede, especialmente naqueles onde existem usuários maliciosos, o conhecimento do administrador da rede sobre o que de fato está trafegando inexiste. Desta maneira, qualquer atribuição de pesos e a identificação de um classificador que estabeleça o melhor *ground truth* tende a ser inadequada. Este processo é determinante para a acurácia do classificador *Weighted Vote*.

Este trabalho abordou o problema de classificação no qual a ferramenta *NeTraMark* foi utilizada para identificar tráfego HTTP real em cenários de camuflagem de tráfego através porta TCP 80. Ao explorar estes cenários foram identificadas algumas limitações na abordagem de [Lee et al., 2011].

Com o intuito de melhorar o processo de classificação de tráfego HTTP real na *NeTraMark* em situações de camuflagem através da porta TCP 80, foi apresentada uma metodologia para uso da *NeTraMark* que gerou um novo classificador chamado FBC (*F-Measure Based Classifier*). O classificador proposto descarta a atribuição empírica de pesos para o classificador *Weighted Vote* utilizando uma configuração genérica. O classificador aplica esses pesos considerando somente classificadores que obtiveram melhor acurácia nas bases *ground truth* analisadas. O classificador também permite o ajuste de acurácia ponderada mínima através do parâmetro $\lim_F$. Este parâmetro influencia no resultado do classificador para identificar este tipo de tráfego.

O desempenho do classificador proposto foi comparado aos demais classificadores implementados na *NeTraMark*, utilizando-se bases capturadas em ambiente controlado e em bases externas disponibilizadas através do portal *OpenPacket*. Os resultados de classificação mostraram-se iguais ou superiores ao classificador de maior acurácia nas bases estudadas e até 26,89% superior ao classificador *Weighted Vote* configurado de forma padrão. Os resultados mostram ainda que o classificador proposto sugere um percentual de até 13,92% maior de fluxos de pacotes HTTP do que os outros classificadores na base estudada que não possuia *ground truth* disponível.

O classificador proposto executa o processo de classificação de tráfego HTTP descartando a necessidade de sucessivos ajustes de pesos e o uso do *ground truth* configurado na *NeTraMark*. Utilizando o valor fixo para $\lim_F$, obtido através do estudo do impacto deste parâmetro na acurácia, permite-se um melhor resultado para o classificador proposto. Em resumo o classificador proposto mostrou-se adequado para classificar tráfego HTTP real nos cenários propostos com acurácia igual ou superior aos classificadores implementados na *NeTraMark*.

Como trabalhos futuros podemos listar os seguintes pontos:

- Um estudo mais detalhado de outras técnicas de camuflagem na porta TCP 80. Tal estudo tornaria o classificador mais abrangente e reduziria a incidência de *Falsos Positivos* para tráfego HTTP real;
- Otimização do classificador proposto com o objetivo de classificar também o tráfego não HTTP que esteja camuflado na porta TCP 80;
- Uso desta metodologia para abordar outros problemas de classificação de tráfego possibilitando a implementação de novos classificadores que possam identificar outras categorias de aplicação (e.g. P2P, Games) na *NeTraMark*;
- Experimentos reais em ambiente de rede organizacionais para uma melhor investigação sobre o classificador proposto. Este trabalho utilizou somente simulações de tráfego em ambiente controlado para definição do classificador.

REFERÊNCIAS BIBLIOGRÁFICAS

[Auld et al., 2007] Auld, T., Moore, A. W., and Gull, S. F. (2007). Bayesian neural networks for internet traffic classification. *IEEE Transactions on Neural Networks*, 18(1):223–239.

[Callado et al., 2009] Callado, A., Kamienski, C., Szabo, G., Gero, B., J.Kelner, Fernandes, S., and Sadok, D. (2009). A survey on internet traffic identification. *IEEE communications surveys and tutorials*, 11(3):38–50.

[Choi, 2006] Choi, Y. (2006). On the accuracy of signature-based traffic identification technique in ip networks. *BCN BusinessUnit*, pages 1–3.

[Connolly, 2012] Connolly, D. (2012). Hypertext transfer protocol http/1.1. `http://www.w3.org/Protocols/rfc2616/rfc2616.html`. Accessado: 27/09/2012.

[Dainotti et al., 2012] Dainotti, A., Pescape, A., and Claffy, K. C. (2012). Issues and future directions in traffic classification. *IEEE Network*, pages 35–40.

[Dusi et al., 2007] Dusi, M., Crotti, M., Gringoli, F., and Salgarelli, L. (2007). Detection http tunnels with statistical mechanisms. *ICC 2007 proceedings*, pages 6162–6168.

[Dusi et al., 2008] Dusi, M., Crotti, M., Gringoli, F., and Salgarelli, L. (2008). Detection of encrypted tunnels across network boundaries. *ICC 2008 proceedings*, pages 1738–1744.

[Ellis et al., 2002] Ellis, D., Whitman, B., Berenzweig, A., and Lawrence, S. (2002). The quest for ground truth in musical artist similarity. *2002 IRCAM*.

[Estrada and Nakao, 2010] Estrada, V. and Nakao, A. (2010). A survey on the use of traffic traces to battle internet threats. *3rd International Conference on Knowledge Discovery and Data Mining*, pages 601–604.

[Gargiulo and Sansone, 2010] Gargiulo, F. and Sansone, C. (2010). Improving performance of network traffic classification systems by cleaning training data. *2010 International Conference on Pattern Recognition*, pages 2768–2771.

[Gringoli et al., 2009] Gringoli, F., Salgarelli, L., Dusi, M., Cascarano, N., Risso, F., and Claffy, K. C. (2009). Gt: picking up the truth from the ground for internet traffic.

[Gu et al., 2010] Gu, R., H.Wang, and Ji, Y. (2010). Early traffic identification using bayesian networks. *Proceedings of IC-NIDC2010*, pages 564–568.

[Hripcsak and Rothschild, 2005] Hripcsak, G. and Rothschild, A. S. (2005). Agreement, the f-measure, and reliability in information retrieval. *American Medical Informatics Association*, 12(3):296–299.

[Karagiannis et al., 2004] Karagiannis, T., Broidom, A., Faloutsos, M., and Claffy, K. (2004). Transport layer identification of p2p traffic. *AACM SIGCOMM IMC 2004*, pages 121–134.

[Karagiannis et al., 2005] Karagiannis, T., Papagiannaki, K., and Faloutsos, M. (2005). Blinc: Multilevel traffic classification in the dark. *ACM SIGCOMM '05*, pages 229–240.

[Kim et al., 2008] Kim, H., Claffy, K., and Fomenkov, M. (2008). Internet traffic classification demystified: myths, caveats,and the best practices. *ACM CoNEXT 2008*, pages 2–12.

[Kurose, 2010] Kurose, J. (2010). *Redes de Computadores e a Internet*. Editora Pearson, 5 edition.

[Lee et al., 2011] Lee, S., Kim, H., Barman, D., Lee, S., and Kwon, T. (2011). Netramark:a network traffic classification benchmark. *ACM SIGCOMM Computer Communication Review*, 41(1):22–30.

[Li et al., 2011] Li, X., Qi, F., Xu, D., and Qiu, X. (2011). An internet traffic classification method based on semi-supervised support vector machine. *IEEE ICC 2011 proceedings*, pages 1–5.

[Moore and Zuev, 2005] Moore, A. W. and Zuev, D. (2005). Internet traffic classification using bayesian analysis techniques. *ACM SIGMETRICS'05*, pages 50–60.

[Mu and Wu, 2011] Mu, X. and Wu, W. (2011). A parallelized network traffic classification based on hidden markov model. *2011 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*, pages 107–112.

[Peng et al., 2009] Peng, B., H.Liu, and Wei, H. (2009). Performance improvement over linux layer-7 content filtering. *2009 Fifth International Conference on Mobile Ad-hoc and Sensor Networks*, pages 522–527.

[Peterson and Davie, 2004] Peterson, L. and Davie, B. (2004). *Redes de Computadores (Uma abordagem de sistemas)*. Editora Campus(Elsevier), 3 edition.

[Salgarelli et al., 2007] Salgarelli, L., Gringoli, F., and Karagiannis, T. (2007). Comparing traffic classifiers. *ACM SIGCOMM Computer Communication Review*, 37(3):65–68.

[Stallings, 2005] Stallings, W. (2005). *Data and Computer Communications*. Editora Campus(Elsevier), 5 edition.

[Szabo et al., 2012] Szabo, G., Szule, J., Turanyi, Z., and Pongracz, G. (2012). Multi-level machine learning traffic classification system. *ICN 2012:The 11th International Conference on Networks*, pages 69–77.

[Tanenbaum, 2003] Tanenbaum, A. S. (2003). *Redes de Computadores*. Editora Campus(Elsevier), 4 edition.

[Venosa et al., 2008] Venosa, P., Macia, N., and Marrone, L. (2008). Qos-traffic classification in the internet. *4th International Conference on Networked Computing and Advanced Information Management*, pages 584–590.

[Williams et al., 2006a] Williams, N., Zander, S., and Armitage, G. (2006a). Evaluating machine learning algorithms for automated network application identification. *Technical Report 060401B*, pages 1–13.

[Williams et al., 2006b] Williams, N., Zander, S., and Armitage, G. (2006b). A preliminary performance comparison of five machine learning algorithms for practical ip traffic flow classification. *ACM SIGCOMM CCR*, 36(5):7–15.

[Witten and Frank, 2005] Witten, I. H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2nd edition.