CLEBER KIEL OLIVO

AVALIAÇÃO DE CARACTERÍSTICAS PARA DETECÇÃO DE PHISHING DE EMAIL

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

CURITIBA

2010

CLEBER KIEL OLIVO

AVALIAÇÃO DE CARACTERÍSTICAS PARA DETECÇÃO DE PHISHING DE EMAIL

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

Área de Concentração: *Ciência da Computação*

Orientador: Prof. Dr. Altair Olivo Santin
Co-orientador: Prof. Dr. Luiz Eduardo S. Oliveira

CURITIBA
2010

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

O49a 2010

Olivo, Cleber Kiel
Avaliação de características para detecção de phishing de email / Cleber Kiel Olivo ; orientador, Altair Olívo Santin ; co-orientador, Luiz Eduardo S. Oliveira. – 2010.
xi, 65 f. : il. ; 30 cm

Dissertação (mestrado) – Pontifícia Universidade Católica do Paraná, Curitiba, 2010
Bibliografia: f. 62-65

1. Phishing. 2. Fraude na Internet. 3. Correio eletrônico. 4. Redes de computação - Medidas de segurança. 5. Informática. I. Santin, Altair Olivo. II. Oliveira, Luiz Eduardo Soares de. III. Pontifícia Universidade Católica do Paraná. Programa de Pós-Graduação em Informática. IV. Título.

CDD 20. ed. – 004

Pontifícia Universidade Católica do Paraná
Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Informática

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

DEFESA DE DISSERTAÇÃO Nº 01/2010

Aos 23 dias do mês de fevereiro de 2010 realizou-se a sessão pública de Defesa da Dissertação **"Avaliação de Características para Detecção de Phishing de e-mail,"** apresentada pelo aluno **Cleber Kiel Olivo** como requisito parcial para a obtenção do título de Mestre em Informática, perante uma Banca Examinadora composta pelos seguintes membros:

Prof. Dr. Altair Olivo Santin PUCPR (Orientador)	(assinatura)	Aprov. (aprov/reprov.)
Prof. Dr. Carlos Alberto Maziero PUCPR		REPROVADO
Prof. Dr. Elias Procópio Duarte Junior UFPR		APROVADO
Prof. Dr. Luiz Eduardo Soares de Oliveira UFPR		APROVADO

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

Programa de Pós-Graduação em Informática PPGIa PUCPR

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

Agradecimentos

Agradeço aos meus pais, pois o que estou colhendo hoje são frutos da educação que recebi. Sem eles teria sido muito mais difícil chegar até aqui e saber que ainda posso ir mais longe.

Um agradecimento especial ao Prof. Dr. Altair Santin, por ter me apoiado incondicionalmente em todos os altos e baixos da minha vida de mestrando. Sei que muitas vezes você foi além do seu papel de orientador para me apoiar.

Ao Prof. Dr. Luiz Eduardo S. Oliveira (UFPR) que, com sua perícia em Reconhecimento de Padrões, facilitou a realização do nosso trabalho, tornando-o cada vez mais interessante.

Ao Prof. Dr. Alcides Calsavara, por ceder gentilmente um espaço em suas aulas, permitindo que eu realizasse meu estágio de docência.

À Renata, pelo apoio, tolerância e compreensão nos últimos anos. Sei que logo terei a minha vez de retribuir.

À CAPES, pelo apoio financeiro, e aos responsáveis pelo funcionamento dessa instituição. Sem a bolsa eu não teria começado meu mestrado, nem descoberto o quanto é interessante fazer pesquisa científica.

Aos demais professores, funcionários e alunos da PUC-PR que, direta ou indiretamente contribuíram de alguma forma.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas

AC	*Autoridade Certificadora*
ASCII	*American Standard Code for Information Interchange*
AUC	*Area Under the Curve*
BSVM	*Biased Support Vector Machines*
C1, C2, C*n*	*Característica 1, Característica 2, Característica n*
DNS	*Domain Name System*
FP	*False Positives*
HTML	*HyperText Markup Language*
HTTP	*HyperText Transfer Protocol*
HTTPS	*HyperText Transfer Protocol Secure*
IP	*Internet Protocol*
LIBSVM	*Library for Support Vector Machines*
MX	*Mail Exchanger*
MTA	*Mail Transfer Agent*
MUA	*Mail User Agent*
PUC-PR	*Pontifícia Universidade Católica do Paraná*
ROC	*Receiver Operating Characteristic*
RR	*Random Result*
SMTP	*Simple Mail Transfer Protocol*
SOM	*Self-Organizing Maps*
SVM	*Support Vector Machines*
TI	*Tecnologia da Informação*
TLS	*Transport Layer Security*

TP	*True Positives*
URI	*Uniform Resource Identifier*
URL	*Uniform Resource Locator*

Resumo

Os trabalhos da literatura técnica para detecção de p*hishing* se baseiam somente na taxa de acerto do classificador para justificar a sua eficácia. Aspectos como a confiança dos resultados (verificada pela taxa de falsos positivos), custo computacional para extração de dados e relevância do conjunto combinado de características sendo avaliadas não são considerados. Esta proposta desenvolve um procedimento que permite obter um conjunto mínimo de características relevantes, o modelo do adversário (*adversarial model).* O *modelo do adversário* resultante da avaliação do conjunto de características provê confiança nos resultados, bom desempenho e flexibilidade à proposta. Os resultados experimentais demonstram a viabilidade da proposta.

Palavras-Chave: 1. Modelo do adversário 2. detecção de phishing de email 3. classificador baseado em SVM (*Support Vector Machines*) 4. Curvas ROC (*Receiver Operating Characteristic*).

Abstract

The proposals of the technical literature for detecting phishing are based only on the success rate of the classifier to justify its effectiveness. Aspects such as reliance of the results (evaluated by the false positive rate), computational effort to extract data and relevance o the combination set of characteristics being evaluated are not considered. This proposal develops develops a procedure that allows to obtain the minimum set of relevant characteristics, the adversarial model. The adversarial model obtained from the characteristics set evaluation provides reliance on the results, good performance and flexibility to the proposal. The experimental results show the proposal feasibility.

Keywords: *1. Adversarial model 2. Email phishing detection 3. SVM-based (Support Vector Machines) classifier 4. ROC (Receiver Operating Characteristic) curves.*

Capítulo 1

Introdução

Phishing é uma técnica que usa engenharia social para fazer vítimas, enganando-as com o objetivo de obter suas informações pessoais (geralmente de cunho financeiro) e depois causar-lhes prejuízos. Na internet, o *phishing* pode chegar ao usuário (vítima) de várias maneiras, através de uma janela *pop-up* no navegador (browser), de mensagens instantâneas ou de emails. Geralmente, a vítima é convencida a executar um clique de mouse, que descarregará e instalará algum *malware* (código malicioso) ou acessará um site fraudulento.

Os *malwares*, por exemplo, podem conter *spywares/keyloggers* – software que copia telas (screenshots) ou copia as teclas pressionadas no teclado para um arquivo e depois as envia a um site específico. O *malware* também pode ser um software que compromete a máquina deixando-a zumbi numa *botnet* – máquina que será comandada remotamente para executar atividades que o seu controlador deseja. Com o *phishing*, a vítima também pode ser induzida a acessar um site clonado (sem ter ciência disto) e deixar lá suas informações pessoais.

O email é o serviço de Internet mais utilizado atualmente [Radicati 2009], sendo portanto o principal canal utilizado para *phishing*. O email tem limitações de projeto, por exemplo, que permitem que o remetente seja forjado. Além disto, a maioria dos clientes de email suporta HTML nativamente, assim todos os recursos da linguagem HTML podem ser utilizados numa mensagem de email. Como o email suporta HTML e o *hyperlink* (link usado em

hipertexto, onde se pode associar a um texto visível uma URL "invisível"), ambos se tornaram uma poderosa ferramenta para os malfeitores (*phishers*).

A técnica de disseminação de *phishing* através de emails é muito semelhante à do *spam*. Isto faz o *phishing* ser considerado como uma subcategoria de *spam*, ou até mesmo ser confundido com o mesmo. Porém, os efeitos negativos do *phishing* geralmente são prejuízos financeiros à vítima, decorrente do roubo de informações bancárias etc., enquanto o *spam* em geral apenas envia email com propagandas ao destinatário sem o consentimento do mesmo.

1.1. Motivação

Diante da ameaça que o *phishing* representa, várias abordagens da literatura técnica propuseram algum modo de detecção que atingisse um alto percentual de acerto no classificador de mensagens. Entretanto, todas elas possuem algumas limitações que impedem que essa ameaça seja eliminada com mais eficiência.

Filtros de email em geral são preparados para detecção de *spam*, o que pode não ser tão eficiente especificamente para a problemática de *phishing*. Os filtros Bayesianos, que fazem a classificação do conteúdo do email com base na ocorrência de determinadas palavras, por exemplo, podem avaliar incorretamente palavras que aparecem em emails não classificados anteriormente como *spam*.

Muitas ferramentas de emails, assim como a maioria das ferramentas de navegador, utilizam listas de origens/remetentes "bons" (*whitelist*) e "maus" (*blacklist*). Normalmente, as *blacklist*s bloqueiam o endereço IP do servidor de email (SMTP) de origem da mensagem, ou o domínio de origem da mensagem, ou ainda o próprio endereço de email do remetente. O bloqueio do endereço IP/domínio pode causar problemas quando o remetente utiliza o servidor de SMTP de algum provedor (e.g. Yahoo, Gmail, etc), pois acaba por bloquear todos os remetentes que o utilizam. Já o bloqueio do email do remetente pode ser ineficiente, visto que o mesmo pode ter sido forjado.

No caso do *phishing*, a origem da mensagem, o endereço IP do *phisher*, a URL alvo do *phishing*, etc., costumam mudar constantemente para não serem rastreados. Além disso, a dificuldade de administração de listas (*whitelist*/*blacklist*) pode ser muito complexa, pois o

fluxo de mensagens pode ser muito alto no servidor SMTP onde a filtragem é feita. Assim, esta abordagem geralmente é ineficiente.

Na literatura técnica, muitos trabalhos consideram várias características do email para detectar o *phishing* (ex: uso de codificação HTML, endereços IP como URL, etc), porém, em geral não é feita uma avaliação da relevância de uma característica quando combinada com as demais. Além disto, o número de características utilizadas na detecção impacta diretamente no tempo de processamento. Neste caso, dependendo do número de características a serem avaliadas, o sistema de detecção de *phishing* pode se tornar o gargalo do sistema de email.

Alguns dos trabalhos relacionados também comentam pouco sobre a taxa de falsos positivos ("sensor de confiabilidade" dos resultados fornecidos pelo classificador). A taxa de falsos positivos é muito importante porque sabemos que, se o usuário do sistema de detecção de *phishing* começar a receber muitos alertas falsos, passará a ignorá-los. Assim, em geral é melhor um sistema que só gere alertas confiáveis (precisos), do que um sistema que emita muitos alertas, mas com pouca credibilidade.

Embora já tenha sido muito estudado, o *phishing* continua sendo uma ameaça digna de preocupação. De acordo com dados estatísticos da MessageLabs, em Fevereiro de 2008, 1% de um total de mais de 1 bilhão de emails trocados diariamente eram *phishing* [MessageLabs 2008]. Além disso, todas as abordagens criadas/apresentadas até o momento possuem algum aspecto negativo, o que inviabiliza o seu uso em determinadas ocasiões. Isso mostra o quanto esse problema ainda pode ser explorado, buscando métodos alternativos de detecção e melhorando algumas das abordagens já existentes.

1.2. Proposta

Este trabalho tem o objetivo de avaliar melhor as características encontradas em *phishing* de emails, buscando fundamentar a importância de cada uma delas em relação ao problema estudado, tanto de forma individual como em combinação com as demais. Entende-se por característica qualquer medida que pode ser extraída de um objeto (Ex: tamanho, cor, formato, etc). Utilizando um método de análise por força bruta, junto a técnica de aprendizado

de máquina, se avalia as combinações de características para encontrar a mais eficiente na classificação de *phishing* de emails.

Após rotular um conjunto de mensagens e separá-las em dois grupos, base de treinamento e teste, usando SVM (técnica de aprendizagem de máquina para resolver problemas de classificação de duas classes), estuda-se o classificador que melhor identifica *phishing* de email a partir de 11 características. A avaliação dos campos do email e *hyperlinks* considerados para a classificação leva em conta o email apenas como um arquivo e, portanto, sem demandar informações adicionais obtidas externamente a esse. A avaliação dos classificadores leva em consideração não só a taxa de acerto do mesmo, mas também a taxa de falsos negativos (obtida a partir das curvas ROC) e a área sob a curva AUC para que o classificador fosse escolhido.

O intuito desta abordagem é encontrar um conjunto com o menor número de características que ofereça confiabilidade nos resultados próxima da obtida com todas as 11 características juntas. Encontrado o conjunto mínimo de características com boa taxa de acerto do classificador e confiabilidade, se obtém o chamado *modelo do adversário (adversarial model).* O *modelo do adversário* representa o perfil de *phishing* no momento da avaliação.

Como as características de *phishing* mudam ao longo do tempo, a abordagem sendo proposta considera a reconfiguração do *modelo do adversário*, sem necessitar reavaliar todas as características novamente. Este processo todo resultou num método que sistematiza a obtenção do modelo do adversário para *phishing* de email.

1.3. Organização

Este trabalho está organizado da seguinte forma:

- Capítulo 2: Serão apresentadas as principais características de *phishing*, incluindo também alguns conceitos relacionados à aprendizagem de máquina, curvas ROC e AUC. As características apresentadas neste capítulo foram reunidas a partir de trabalhos relacionados e, adicionalmente, novas características identificadas no decorrer do trabalho.

- Capítulo 3: Este capítulo apresenta os principais trabalhos relacionados encontrados na literatura técnica. Para cada um deles, suas características de *phishing* utilizadas são apresentadas respeitando o seus respectivos pontos de vista. Adicionalmente, são apresentadas considerações sobre esses trabalhos em geral, e uma consolidação de todas as características utilizadas por cada trabalho.
- Capítulo 4: O capítulo 4 apresenta detalhes da proposta e do trabalho realizado, tais como a preparação das bases de emails lícitos e *phishing* – para treinamento do classificador e teste, procedimento de "força bruta" utilizado, análise das curvas ROC e da AUC, obtenção do *modelo do adversário*, e, finalmente, os resultados do teste de desempenho. Adicionalmente, também é apresentado o teste realizado para demonstrar a real importância do *modelo do adversário* e detalhes da implementação do protótipo.
- Capítulo 5: O quinto e último capítulo apresenta as conclusões gerais sobre este trabalho.

Capítulo 2

Phishing de email: Características e Técnicas de Detecção Baseadas em Reconhecimento de Padrões

Phishing é uma forma de estelionato que usa engenharia social para fazer vítimas, enganando-as geralmente com o objetivo de obter suas informações pessoais (geralmente de cunho financeiro) e depois causar-lhes prejuízos. De acordo com o Código Penal Brasileiro, estelionato é "obter, para si ou para outrem, vantagem ilícita, em prejuízo alheio, induzindo ou mantendo alguém em erro, mediante artifício, ardil, ou qualquer outro meio fraudulento" [Código Penal Brasileiro, Título II, Cap. VI, Art. 171].

O email é o serviço de Internet mais utilizado para disseminação de phishing. Logo, o problema foi considerado como sendo de Reconhecimento de Padrões, visto que o intuito da proposta foi de encontrar atributos essenciais contidos nos emails, de forma que fosse possível a identificação de mensagens de *phishing* que utilizem tal serviço. Para isso, esta proposta assume o *phishing* de email como um problema de Reconhecimento de Padrões de duas classes, ou seja, diversas características (qualquer medida possa ser extraída de um objeto) são extraídas dos emails para que um modelo seja obtido com a finalidade de classificar as mensagens em duas classes distintas: *phishing* e não-*phishing*. Características podem ser simbólicas (e.g. cor, forma geométrica, etc) ou numéricas (e.g tamanho, posição, etc), sendo que as numéricas podem ser contínuas ou binárias. As características contínuas podem assumir vários valores (e.g 1, 500, 3.1415, 19.2, etc.) enquanto as binárias assumem somente dois (-1: falso ou 1: verdadeiro). Este trabalho utiliza características contínuas e binárias.

Inicialmente, algumas características de *phishing* de email são escolhidas de acordo com a sua justificativa em relação ao problema. Em seguida, um extrator de características as procura em uma base de emails. Esta extração irá gerar um vetor de características para cada email, o que servirá de base para que o classificador execute a etapa de treinamento/aprendizagem. Ao final do processo, espera-se que o classificador possa distinguir mensagens de email que são *phishing* das não-*phishing.*

Em geral, as técnicas para detecção de *phishing* baseadas em características oferecem vantagens em relação às outras. Por exemplo, os filtros bayesianos, que são treinados a partir das palavras e termos ortográficos contidos nas mensagens, são ineficientes no bloqueio de emails ilícitos escritos em idiomas estrangeiros quando não há exemplares dos mesmos na base de treinamento. Os *phishing* de email também costumam mudar muito o contexto da mensagem, sendo geralmente bem polêmico e utilizado para chamar a atenção do usuário, o que consequentemente acarreta uma mudança de termos ortográficos utilizados, necessitando de um novo treinamento. Visto que as características estão presentes independentemente do idioma utilizado na mensagem ou o seu contexto, as técnicas de detecção de *phishing* baseadas em características são menos suscetíveis a tais falhas.

Após a extração de características do email, as informações são repassadas a um classificador, que basicamente é um algoritmo que irá enquadrar a mensagem em uma das duas categorias (*phishing* ou não-*phishing*).

Este capítulo apresenta, com uma nomenclatura padronizada, algumas das características utilizadas nos trabalhos relacionados e outras que foram identificadas durante o desenvolvimento do projeto. Também são apresentados alguns conceitos de aprendizado de máquina, curvas ROC (*Receiver Operating Characteristic)* e área sob a curva ROC (AUC - *Area Under the Curve*).

2.1. Características de Phishing

As estratégias utilizadas pelos *phishers* para ludibriar o usuário do sistema de email estão muito relacionadas ao uso de subterfúgios técnicos que geralmente não são do conhecimento das vítimas. Algumas técnicas de detecção de *phishing* estão baseadas na identificação de um conjunto de características envolvendo principalmente o cabeçalho

(*header*) e o corpo (*body*) do email. Há também características de *phishing* que podem envolver a consulta a recursos externos ao serviço de email para tentar identificá-lo.

A seguir, serão apresentadas características de *phishing* que foram consideradas importantes para a realização deste trabalho.

- **C1: Hyperlink com texto visível em formato de URL, mas apontando para uma URL diferente do texto visível**

É utilizado um hyperlink HTML com textos visíveis (legíveis) imitando uma URL. Um exemplo desta codificação em HTML pode ser:

* http://www.*pay*pal.com/login.php *

O texto visível mostra o *hyperlink http://www.*paypal*.com/login.php*, porém a URL que será carregada se houver um clique no *hyperlink* será *http://*play*pal*.com.

O texto visível no formato de uma URL ao invés de imagens com link ou textos no estilo "clique aqui", podem causar uma falsa impressão de segurança ao usuário desavisado que não sabe discernir se a URL que aparece em tela é a mesma que será carregada após o clique do mouse.

Esta característica binária também foi utilizada em [Chen e Guo 2006], [Cook, Gurban e Daniluk 2008] e [Fete, Norman e Anthony 2007].

- **C2: Hyperlink com um texto visível qualquer, mas apontando diretamente para um endereço IP como URL**

Este recurso é bastante utilizado pelos *phishers*, pois não precisam expor seus dados cadastrais num registro DNS, uma vez que não será necessário resolver o nome DNS, pois o endereço IP do site malicioso é explicitamente especificado no *hyperlink*.

Um exemplo dessa situação em HTML é:

* Clique Aqui *

Esta característica binária também foi utilizada em [Chen e Guo 2006], [Cook, Gurban e Daniluk 2008], [Fete, Norman e Anthony 2007] e [Basnet, Mukkamala e Sung 2008].

- **C3: Email com o corpo (body) codificado em formato HTML**

O principal meio de disseminação de *phishing* é a codificação HTML suportada na maioria dos clientes de email, pois o *hyperlink* quando mostrado para o usuário "esconde" a URL sob o texto visível. Ou seja, é escondido o endereço do site que será carregado quando ocorrer um clique de mouse no *hyperlink*. E.g:

Content-Type: text/html

O formato HTML permite ainda o uso de outros recursos, como, por exemplo, a inserção de formulários. Por esta razão, o formato HTML está ligado à maioria das características que exploram subterfúgios técnicos para facilitar o ataque em *phishing* de email.

Esta característica binária também foi utilizada nos trabalhos [Fete, Norman e Anthony 2007] e [Basnet, Mukkamala e Sung 2008].

- **C4: URL muito extensa**

A visualização de uma URL com texto longo geralmente confunde o usuário inexperiente com a representação de URI (*Uniform Resource Identifier*), pois o domínio real pode ser camuflado pela utilização excessiva de "subdomínios" (separados por pontos) ou subdiretórios. Um exemplo dessa situação pode ser a seguinte URL:

http://security.update.playpal.com/login/paypal.com/login.php

Esta característica contínua também foi utilizada em [Fete, Norman e Anthony 2007].

- **C5: Domínio do remetente do email diferente do domínio de alguma URL no corpo da mensagem**

Os *phishers* falsificam o remetente da mensagem para tentar alcançar alguns servidores de email, mesmo que estes usem filtros baseados em *blacklist* (listagem de URL/domínio conhecidos como origens de *phishing/spam*). Porém, no corpo da mensagem se encontra a URL maliciosa que geralmente não é avaliada quando a *blacklist* é verificada. Como o emissor de um email pode ser facilmente forjado, os *phishers* tentam imitar domínios de remetentes bastante conhecidos ou confiáveis aos olhos do usuário do email, para no corpo do email colocar a URL que efetivamente o levará a ser vítima de golpe.

Se o domínio fraudulento for *badsite.com*, para fazer vítimas, é preferível que o remetente do email seja forjado (Ex: seguranca@microsoft.com) ao invés de revelar o verdadeiro endereço (Ex: seguranca@badsite.com). No corpo do email, mesmo que a URL fraudulenta não esteja explícita, estará escondida atrás de um texto âncora (visível) ou uma imagem. Com isso, quase sempre o domínio do remetente da mensagem (forjado) será diferente de pelo menos uma URL do corpo da mensagem (endereço fraudulento).

Esta característica binária também foi utilizada em [Chen e Guo 2006] e [Basnet, Mukkamala e Sung 2008].

- **C6: Imagem carregada a partir de domínio externo diferente das URLs do corpo da mensagem**

Nesta abordagem o corpo do email contém imagens (e.g. logomarcas) que são carregadas a partir dos sites autênticos, porém a URL que enganará a vítima é fraudulenta, e obviamente, diferente do domínio da imagem. Um exemplo pode ser:

<img src=http://www.policiafederal.gov.br/imagens/logo.jpg> Você está intimado a comparecer em nossa delegacia! Clique aqui para saber o motivo .

Neste caso o usuário verá o logo verdadeiro da polícia federal e a mensagem "*Você está intimado a comparecer em nossa delegacia!".* No *hyperlink* HTML o texto visível é "*Clique aqui para saber o motivo".* Se este *hyperlink* for clicado, fará o download de malware.exe a partir de *badsite.com*.

Esta característica binária não foi utilizada em nenhum dos trabalhos relacionados do Capítulo 3.

- **C7: Descarga de uma imagem a partir de um endereço IP**

Este é um caso típico do uso de endereço IP ao invés de uma URL registrada num domínio DNS para descarga de imagens. Um exemplo disso pode ser:

*<img src="http://*200.218.145.21*/images_paypal/logo.gif">*

Neste caso a imagem falsa é carregada do próprio site do *phisher*, e portanto qualquer tipo de adulteração pode ser esperada.

Casos assim tornam a mensagem suspeita pois entidades autênticas, quando carregam imagens de fontes externas, não possuem um motivo para utilizar um endereço IP visto que dispõem de um domínio registrado no serviço de DNS.

Esta característica binaria não foi utilizada em nenhum dos trabalhos relacionados do Capítulo 3.

- **C8: Número de domínios da URL**

Nesta abordagem os *phishers* embutem mais de um domínio numa mesma URL, sendo um deles geralmente bem conhecido. A idéia é confundir o usuário que em geral sabe que o domínio fica mais próximo ao fim da URL. No exemplo abaixo, uma pasta do servidor HTTP recebe um nome no formato de uma URL de uma entidade bem conhecida:

*http://www.badsite.com/login/*bancodobrasil.com.br/login.php

Nesta situação o usuário também pode ter uma falsa sensação de segurança, imaginando que está acessando o verdadeiro site do Banco do Brasil, quando na verdade está sendo direcionado para *badsite.com*.

Esta característica, considerada contínua, também foi utilizada em [Basnet, Mukkamala e Sung 2008].

- **C9: Número de subdomínios da URL**

De forma similar à característica C8, alguns *phishers* adicionam subdomínios para dar uma aparência mais confiável à URL utilizando nomes de entidades autênticas e bem conhecidas como no exemplo abaixo:

*http://recadastro.*receitafederaldobrasil*.badsite.com*

Em casos assim, o *phisher* espera que os subdomínios "*recadastro"* e "*receitafederaldobrasil"* chamem mais atenção do usuário do que o próprio domínio *badsite.com*.

Esta característica contínua também foi utilizada em [Basnet, Mukkamala e Sung 2008].

- **C10: Hyperlink com imagem ao invés de texto visível e URL da imagem baseada em endereço IP**

Neste caso, o *hyperlink* HTML é uma imagem ao invés de um texto visível e a URL associada é um endereço IP. O *Phisher* espera que o usuário clique na imagem que irá levá-lo ao site fraudulento. Um exemplo em HTML dessa situação pode ser o seguinte:

* <img src=logo.jpg> *

Assim como na característica C2 o uso de um endereço IP serve para o anonimato do *phisher*, pois não é preciso cadastrar um domínio num serviço de nomes DNS. Casos assim tornam o email suspeito, pois além de utilizar um endereço IP de forma não necessária para entidades autênticas, o mesmo se disfarça usando uma imagem como âncora para o *hyperlink.*

Esta característica binária não foi utilizada em nenhum dos trabalhos relacionados do Capítulo 3.

- **C11: Texto âncora do *hyperlink* não fornece informações sobre o seu destino**

A URL fraudulenta também pode ser escondida atrás de um texto âncora qualquer que não seja uma URL, ou seja, de forma diferente da característica C1 o texto visível (âncora) se apresenta com um texto e não com uma URL. Por exemplo:

** Termo de Compromisso **

Esta é uma característica binária e também foi utilizada em [Chen e Guo 2006].

Quase todas as características relacionadas acima foram retiradas dos trabalhos relacionados (Cap. 3), sendo que estão destacadas nesta seção porque possuem uma justificativa convincente em relação ao *phishing* de email. Além disto, estes conjuntos de características compõem uma espécie de conjunto de consenso entre as características mais frequentemente referenciadas e utilizadas para o fim de detecção de *phishing*. Algumas características como C6 e C7 foram identificadas a partir de um estudo feito no conjunto de mensagens de email que passam num gateway SMTP de uma grande organização governamental e da coleta feita na universidade (detalhes sobre esta base serão apresentados na seção 4.2)

2.2 Aprendizado de Máquina

Em ciência da computação, Aprendizado de Máquina é uma subárea da Inteligência Artificial que tem como objetivo estudar e produzir técnicas ou sistemas computacionais capazes de adquirir conhecimento de forma automática [Resende 2003].

Técnicas de aprendizagem de máquina podem ser utilizadas para o aperfeiçoamento de determinadas atividades computacionais. Seu uso vai desde o auxílio em diagnósticos médicos até o reconhecimento de escrita e fala, robótica, etc. Neste trabalho o aprendizado de máquina foi utilizado para a detecção de *phishing* de email.

Como citado anteriormente, o problema de detecção de *phishing* pode ser visto como um problema de classificação de duas classes. Nesse contexto o algoritmo de aprendizagem de máquina mais adequado é o SVM (*Support Vector Machines* – Máquinas de Vetor de Suporte), pois foi desenvolvido originalmente para resolver esse tipo de problema [Vapnik 1995]. A literatura técnica mostra que o SVM tem sido aplicado com bastante sucesso em diversos domínios de aplicação, inclusive na detecção de *phishing* [Basnet, Mukkamala e Sung 2008], [Chandrasekaran, Narayanan e Upadhyaya 2006], [Pan e Ding 2006] e [Kim, Jang, Cho e Park 2006].

Basicamente o funcionamento de uma SVM pode ser descrito da seguinte forma: dadas duas classes e um conjunto de pontos que pertencem a essas classes, uma SVM determina o hiperplano que separa os pontos de forma a colocar o maior número de pontos da mesma classe no mesmo lado, enquanto maximiza a distância de cada classe a esse hiperplano (figura 2.1). A distância de uma classe a um hiperplano é a menor distância entre ele e os pontos dessa classe e é chamada de margem de separação. O hiperplano gerado pelo SVM é determinado por um subconjunto dos pontos das duas classes, chamados vetores de suporte [Chaves 2006].

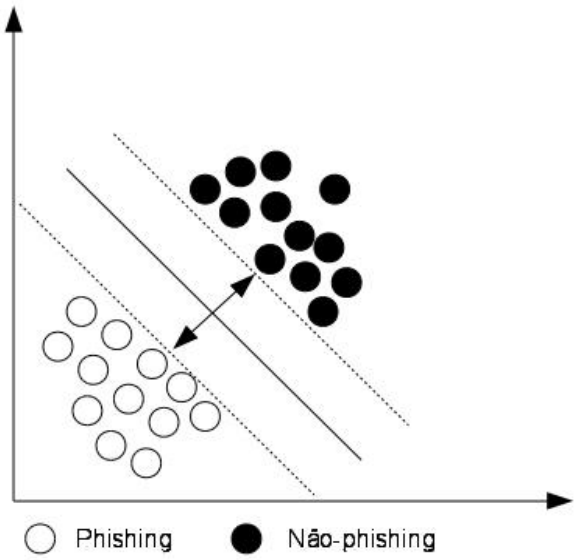

Figura 2.1: Separação de classes pelo hiperplano

O treinamento do SVM pode ser resumido como sendo a detecção dos vetores de suporte dentre as amostras de treinamento. Após isso a função de decisão $f(x) = \sum_i \alpha_i \gamma_i K(x , x_i) + b$ pode ser utilizada para fornecer a classe de uma amostra não rotulada.

Os parâmetros α_i e b são encontrados através de um algoritmo de programação quadrática, onde x é a amostra não rotulada e x_i é o vetor de suporte. A função $K(x , x_i)$ é conhecida como função de *kernel* e mapeia o espaço das amostras dimensões mais altas, onde as amostras tornam-se linearmente separáveis.

Existem diferentes tipos de *kernels* que podem ser utilizados, entre eles, Linear, Polinomial, Gaussiano e Tangente Hiperbólica. Nesse trabalho será utilizado o *kernel* Gaussiano pois é o mais utilizado, e os parâmetros do *gamma* e C serão definidos através de um *grid search.*

2.3 Curvas ROC e AUCs

Dado um classificador designado a classificar instâncias de um problema de reconhecimento de padrões de duas classes (e.g. *phishing* ou não *phishing*), cada instância pode resultar em quatro tipos de situações. Se a instância for positiva (não *phising*) e classificada como positiva, ela é contabilizada como um verdadeiro positivo (*true positive*).

Porém, se esta mesma instância positiva for classificada como negativa, ela é contabilizada como um falso negativo (*false negative*). Da mesma forma, se uma instância negativa for classificada como negativa, ela é contabilizada como um verdadeiro negativo (*true negative*), e, se for classificada como positiva é contabilizada como um falso positivo (*false positive*). A partir do classificador e do conjunto de instâncias (a base de testes), essas quatro situações podem ser representadas através de uma Matriz de Confusão (Figura 2.2).

		Classe Verdadeira	
		P	**N**
Avaliação do classificador	**P**	Verdadeiros Positivos	Falsos Positivos
	N	Falsos Negativos	Verdadeiros Negativos

Figura 2.2: Matriz de Confusão

A partir desses valores, é possível obter as taxas de falsos positivos (FP – razão entre o total de falsos positivos e totais de instâncias negativas) e de verdadeiros positivos (TP – razão entre o total de verdadeiros positivos e o total de instâncias positivas), que podem ser avaliadas de forma mais prática através de curvas ROC.

Uma curva ROC (*Receiver Operating Characteristic*) é um gráfico que mostra a relação entre a sensibilidade e a especificidade de um classificador. A sensibilidade pode ser definida como a probabilidade de classificar corretamente uma amostra rotulada como positiva, enquanto a especificidade pode ser definida como a probabilidade de classificar corretamente uma amostra cujo rótulo seja negativo. Em outras palavras, uma curva ROC mostra a relação entre os Verdadeiros Positivos e os Falsos Positivos. A Figura 2.3 ilustra as taxas de TP x FP de cinco classificadores diferentes. Em geral, quanto mais próximo do canto superior esquerdo, melhor é o classificador (maior taxa de TP e menor taxa FP possíveis).

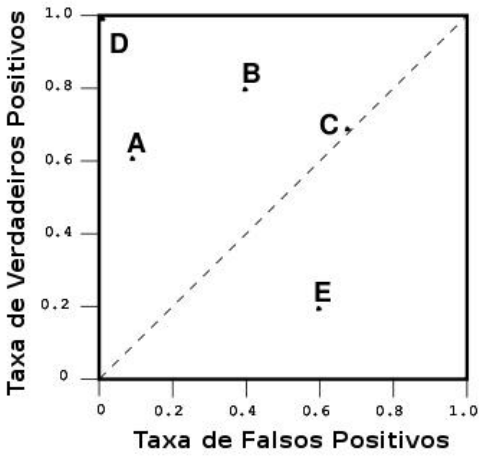

Figura 2.3: ROC básica mostrando cinco classificadores distintos [Fawcett 2006]

Uma das principais vantagens do uso de ROC na avaliação de classificadores está no fato de que a ROC não é sensível a mudança na distribuição de classes. Se a proporção entre amostras positivas e negativas na base de testes for diferente da relação encontrada na base de treinamento, as curvas ROC permanecem as mesmas [Fawcett 2006].

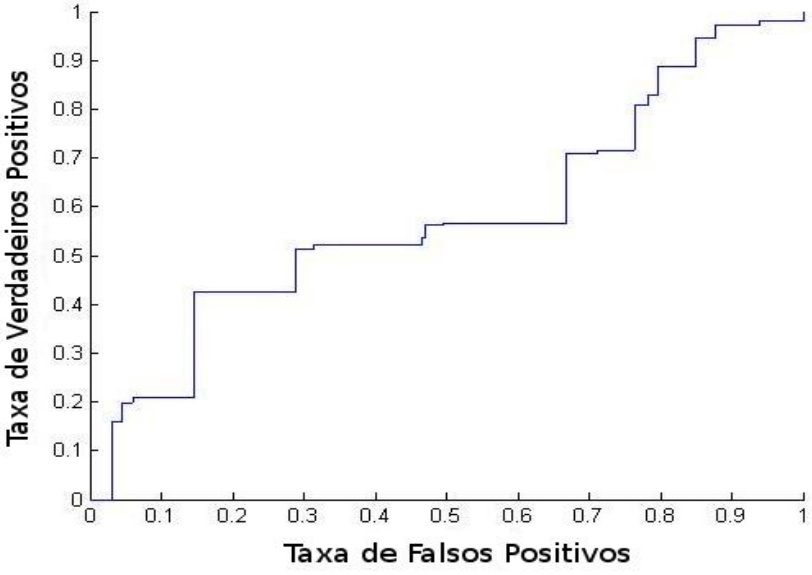

Figura 2.4: Curva ROC com as variações de TP x FP de um classificador

O uso de curvas ROC pode ser útil para visualizar melhor e selecionar classificadores de acordo com seus resultados. Os gráficos são bidimensionais e são representadas em forma

de gráficos onde o eixo Y representa a taxa de verdadeiros positivos (TP) e o eixo X representa os falsos positivos (FP) (figura 2.3).

No caso do SVM, cada instância recebe um valor que representa o "percentual de certeza" com o qual ela foi classificada como positiva ou negativa. Pode-se dizer também que este valor é a probabilidade de uma determinada mensagem pertencer ou não a uma determinada classe. Com isso é possível definir um limiar (*threshold*) para essa probabilidade, permitindo que um único classificador possua várias relações TP x FP ajustáveis. Estas situações também podem ser identificadas e representadas por curvas formadas no espaço da ROC (figura 2.4). Após a visualização em forma de gráfico, é possível escolher a melhor configuração de acordo com as necessidades do sistema.

Outra forma de representação da eficiência de um classificador é através da AUC (*Area Under the Curve*). Diferentemente das curvas ROCs, que avaliam os classificadores com base em dois valores (TP x FP), as AUCs fornecem um único valor que é basicamente a área calculada de um espaço formado abaixo da ROC. Sendo apenas um valor, pode tornar mais fácil a escolha do classificador. Em geral, quanto melhor o classificador, mais próximo de 1 é o valor da AUC. A Figura 2.5 mostra dois exemplos de AUC, uma abaixo da curva ROC A e outra abaixo da curva ROC B.

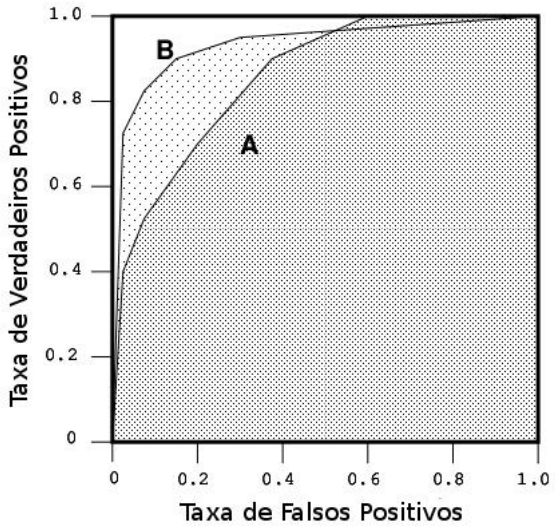

Figura 2.5: Área abaixo das curvas (AUC) ROC A e ROC B [Fawcett 2006]

A análise de curvas ROC e AUCs pode oferecer bons subsídios na escolha de classificadores, sendo que cada técnica pode ser mais apropriada dependendo da situação. Por exemplo, se a necessidade do sistema é um classificador que atinja o máximo de verdadeiros positivos sem importar a quantidade de falsos positivos, a análise da curva ROC pode ser útil, e nesse caso a AUC não é necessariamente a melhor. Já em casos em que é desejável o maior número de verdadeiros positivos com o mínimo de falsos positivos, tanto as ROCs quanto as AUCs poderiam ser utilizadas. Em uma terceira situação, por exemplo, em que são obtidas ROCs muito parecidas, a análise poderia ser feita em combinação com as AUCs, utilizando o valor da área como critério de desempate.

2.4 Considerações

Phishing é uma ameaça que costuma mudar constantemente suas técnicas utilizadas para enganar as vítimas. Isto torna-o um objeto de estudo com grande amplitude na Ciência da Computação, justificando a necessidade do desenvolvimento de novas técnicas que visam a sua detecção.

Técnicas de detecção baseadas em características possuem vantagens em relação à outras abordagens. Por exemplo, as mesmas características podem ser extraídas independentemente do idioma da mensagem. O uso dessas características, aliado ao uso de uma técnica de aprendizagem de máquina como o SVM (que se destaca na resolução de problemas de Reconhecimento de Padrões de duas classes), pode resultar em um modelo de classificação com boa taxa de acerto e confiabilidade. Adicionalmente, apesar de haverem várias propostas para solucionar o problema, nota-se que o uso de técnicas como a análise das curvas ROC e AUCs não é realizado em boa parte dos trabalhos técnicos.

Além disso, muitas abordagens ainda utilizam um número muito alto de características, sem pensar na questão do desempenho durante a filtragem das mensagens em um ambiente de correio eletrônico em produção. Escolher as características que melhor representam o *phishing* e possuem uma justificativa para serem assim consideradas pode contribuir significativamente na obtenção de um modelo de classificação enxuto e com boa taxa de acerto.

Capítulo 3

Trabalhos Relacionados

A seguir, serão abordados alguns dos trabalhos que visam o bloqueio de *phishing* de email. Quando foi possível, foram destacadas quais características de *phishing*, enumeradas no Capítulo 2, são consideradas em cada trabalho. Quando isso não é mencionado explicitamente significa que a característica considerada no trabalho não tem uma explicação intuitiva para caracterizar *phishing* ou não foi devidamente justificada para ser assim considerada.

As características serão consideradas de acordo com o ponto de vista de cada abordagem. A seção 3.1 apresenta uma listagem de características que não foram apresentadas no Capítulo 2 por não serem consenso ou por não possuírem relevância significativa para serem definidas como caraterísticas importantes de *phishing*.

3.1. Características complementares de *phishing*

Antes de iniciarmos a seção de trabalhos relacionados enumeraremos algumas características que são utilizadas em alguns trabalhos para facilitar seu entendimento. Será utilizado a mesma denominação sequencial do Capítulo 2 (C1, C2, …, Cn) para facilitar a citação das características nos trabalhos relacionados. Assim, as características a seguir serão enumeradas em continuação a listagem apresentada no Capítulo 2, isto é da característica C12 até a C19.

Esta estratégia de organização do assunto também foi adotada para facilitar o seu entendimento, dado que é comum nos trabalhos relacionados encontrarmos a mesma característica descrita de maneira diferente.

- **C12: Uso de Javascript no corpo do email**

A linguagem Javascript pode ser utilizada para vários fins. Como exemplo pode-se citar seu uso para mudar as informações da barra de status do cliente de email ou de um navegador de Internet. Assim, o Javascript se torna uma ferramenta muito utilizada pelos *phishers* para interagir com a interface gráfica do usuário e facilitar as manobras de manipulação do usuário, no sentido de induzi-lo a acreditar no *phishing* [Fette, Norman e Anthony 2007] e [Basnet, Mukkamala, e Sung 2008].

- **C13: Corpo do email contendo formulário HTML**

A linguagem HTML permite a inserção de formulário, que os falsários (*phishers*) utilizam em nome da comodidade para o cliente. Evidentemente, junto com o formulário, o *phisher* encaminha uma mensagem tentando enganar o usuário no sentido de parecer que o email é verdadeiramente da entidade que foi personificada.

O conteúdo do formulário geralmente solicita informações pessoais para atualização de cadastro, etc. As informações do formulário sempre são enviadas para alguma URL/IP onde serão armazenadas e utilizadas a posteriori, para causar algum dano a sua vítima [Basnet, Mukkamala, e Sung 2008].

- **C14: Número de hyperlinks contidos no corpo da mensagem**

Para maximizar as chances de obter um clique do mouse do usuário em um *hyperlink* de um site fraudulento, o *phisher* aumenta o número desses no corpo da mensagem.

Assim, quando o número de *hyperlinks* contidos no corpo da mensagem for repetido muitas vezes este passa a ser um indicativo de que a mensagem é suspeita e deve ser considerada como uma característica de *phishing*, pois combinada com as outras pode mesmo identificar tal ação [Fette, Norman e Anthony 2007] e [Basnet, Mukkamala, e Sung 2008].

- **C15: URL com caracteres codificados em ASCII**

Ao invés de utilizar caracteres ASCII legíveis, os *phishers* usam uma cadeia de caracteres representada pelo valor numérico (expresso no sistema de numeração hexadecimal) dos códigos da tabela ASCII correspondente a cada caractere legível. Desta forma, é possível mascarar uma URL de modo que o usuário não consiga identificá-la − por desconhecimento desta forma de representação de URLs e, portanto o usuário acaba clicando no *hyperlink* e sendo pego pelo golpe.

Um exemplo de código ASCII para mascarar um endereço fraudulento pode ser:

* Clique aqui *

Neste exemplo, o navegador interpretaria o código ASCII e direcionaria o *hyperlink* para a URL http://badsite.com [Chen e Guo 2006].

- **C16: Redirecionamento a partir de uma URL, exploração de vulnerabilidades**

Esta abordagem está bastante ligada a presença de vulnerabilidades em versões dos softwares cliente de email ou navegador de Internet. A partir da exploração de uma vulnerabilidade de um destes *softwares* – mais frequentemente o navegador, o usuário é redirecionado de um site legítimo para um malicioso [Chen e Guo 2006]**.**

- **C17: Busca por palavras-chave no corpo da mensagem**

O *phisher* normalmente tenta se passar por uma entidade de credibilidade (e.g. Instituição financeira ou orgão do governo). A mensagem do email tenta atrair a atenção do usuário (vítima) normalmente dizendo que sua conta foi bloqueada, ou que há uma atualização urgente de segurança que deve ser instalada, etc.

Tal estratégia define uma tendência ao uso de determinadas palavras, que normalmente são empregadas neste tipo de *phishing* e podem ser detectadas por uma ferramenta com este propósito. Esta técnica é similar a usada na detecção de SPAM, que via de regra se comporta de maneira similar para alguns tipos de produtos [Fette, Norman e Anthony 2007] e [Basnet, Mukkamala, e Sung 2008].

- **C18: Uso de ferramenta anti-spam para auxiliar na classificação**

As ferramentas anti-spam são bastante estudadas e desenvolvidas, podendo auxiliar na detecção de *phishing*, principalmente na busca de palavras-chave no corpo do email. Além da identificação de palavras no corpo do email estas ferramentas possuem recursos para cálculos que resultam em métricas, o que pode fornecer informações sobre o conjunto de palavras utilizadas e, portanto facilitar a identificação de *phishing* [Fette, Norman e Anthony 2007].

- **C19: Consulta a serviço externo ao sistema de email**

Vários serviços externos podem ser consultados para obter informações adicionais sobre uma URI que é origem de um email, por exemplo.

Alguns destes serviços podem ser: consulta a um site de busca por uma URL/assunto suspeita(o) a fim de confirmar a fraude, consulta ao serviço de DNS (por exemplo, para confrontar o endereço IP do *MX* do domínio do remetente com o endereço IP que enviou a mensagem), consulta ao serviço de *whois* para descobrir a idade de um domínio contido no email, pois domínios com data de criação muito recentes podem ser considerados suspeitos. Estas informações adicionais podem prover indícios mais fortes de que uma mensagem é *phishing*, porém possuem a desvantagem de demandar consultas constantes a serviços Internet – o que pode impactar no desempenho ou mesmo no funcionamento no sistema de email [Cook, Gurbani, e Daniluk 2008], [Fette, Norman e Anthony 2007] e [Basnet, Mukkamala, e Sung 2008].

3.2. Online Detection and Prevention of Phishing Attacks

Chen e Guo criaram uma abordagem para o cliente de email que se baseia em 6 características principais (C1, C2, C5, C11, C15 e C16) [Chen e Guo 2006]. A análise da mensagem é feita principalmente através dos *links* contidos no corpo do email.

Inicialmente, o algoritmo verifica se as características C1, C2, e C15 são verdadeiras e toma algumas ações. Caso o hyperlink não possua a forma de uma URL (nesse caso, o hyperlink pode ser um texto ou imagem qualquer), o domínio do remetente é consultado em listas de bons e maus remetentes (*whitelist/blacklist*), o que determina se a mensagem é ou não fraudulenta.

O algoritmo dessa abordagem é apresentado a seguir:

```
/*  v_link: visual link;
    a_link: actual_link;
    v_dns: visual DNS name;
    a_dns: actual DNS name;
    sender_dns: sender's DNS name. */

int LinkGuard(v_link, a_link} {
  v_dns = GetDNSName(v_link);
  a_dns = GetDNSName(a_link);
  if ((v_dns and a_dns are not
    empty) and (v_dns != a_dns))
       return PHISHING;
  if (a_dns is dotted decimal)
       return POSSIBLE_PHISHING;
  if(a_link or v_link is encoded)
  {
    v_link2 = decode (v_link);
    a_link2 = decode (a_link);
    return LinkGuard(v_link2, a_link2);
  }
 /* analyze the domain name for
    possible phishing */
  if(v_dns is NULL)
    return AnalyzeDNS(a_link);
  }

int AnalyzeDNS (actual_link) {
 /* Analyze the actual DNS name
according
    to the blacklist and whitelist*/
  if (actual_dns in blacklist)
      return PHISHING;
  if (actual_dns in whitelist)
      return NOTPHISHING;
  return PatternMatching(actual_link);
}
int PatternMatching(actual_link){
 if (sender_dns and actual_dns are
different)
      return POSSIBLE_PHISHING;
 for (each item prev_dns in seed_set)
 {
  bv = Similarity(prev_dns, actual_link);
  if (bv == true)
       return POSSIBLE_PHISHING;
 }
 return NO_PHISHING;
}
float Similarity (str, actual_link) {
 if (str is part of actual_link)
     return true;
 int maxlen = the maximum string
    lengths of str and actual_dns;
 int minchange = the minimum number of
     changes needed to transform str
     to actual_dns (or vice verse);
 if (thresh<(maxlen-minchange)/maxlen<1)
     return true
 return false;
}
```

Se os valores consultados não constarem em nenhuma lista, é verificado se a característica C5 é verdadeira e, se assim for, a mensagem é classificada como suspeita. Uma última análise compara as URLs do email com aquelas que o usuário já visitou (existe um banco de dados que possui essas URLs). Se houver alguma URL similar às do email, esse é classificado como suspeito. Essa similaridade é analisada por um módulo específico, que busca similaridades na escrita da URL (e.g. microsoft.com e micr0s0ft.com).

Se não houver nenhuma URL similar a outras do banco de dados, a mensagem é classificada como não-*phishing* e o algoritmo é encerrado.

De acordo com os autores, a técnica de detecção proposta atingiu um percentual de acerto de 96%, embora esse valor tenha sido obtido somente com uma base de *phishing* com 203 mensagens. O ideal seria realizar testes também com mensagens legítimas para poder avaliar a taxa de falsos positivos.

Uma vantagem da abordagem é a não-necessidade de uma etapa de aprendizagem. Entretanto, se as características de *phishing* mudarem, a fórmula utilizada na detecção pode ser inviabilizada. Nota-se também que as características são consideradas de forma isolada, ou seja, nenhuma combinação de características foi estudada. A característica C16, apesar de aparecer no sumário das características, não possui evidências claras de seu uso no algoritmo.

3.3. Phishwish: A Stateless Phishing Filter Using Minimal Rules

Cook e seus colegas propuseram uma técnica que atingiu 95,72% de taxa de acerto em uma base contendo 81 phishing de email e 36 emails legítimos, utilizando um classificador com 11 características (C1, C2, 4 características derivadas de C19 e 5 características não documentadas) [Cook, Gurbani, e Daniluk 2008]. A proposta está baseada nas seguintes regras:

- Regra 1: a partir dos resultados de um mecanismo de busca (E.g: Google) é verificado se o email está tentando direcionar o usuário a um endereço diferente da página de *login* atual da companhia a qual a mensagem supostamente foi originada (C19 da seção 3.1).
- Regra 2: em emails no formato HTML é verificado se a URL visível utiliza TLS, então essa é comparada a URL que consta na URI. Se esta última não utilizar TLS a característica é indicativa de *phishing* (característica não documentada).
- Regra 3: é verificado se é feito uso de endereço IP na URL de *login* ao invés de um nome de domínio (C2 da seção 2.1).
- Regra 4: verifica se o nome da entidade que aparece na URL está fora da porção correspondente ao nome do domínio (característica não documentada).
- Regra 5: em emails no formato HTML é verificado se o texto âncora do link tem o formato de uma URL e então comparado com o endereço para o qual o mesmo aponta (C1 da seção 2.1).

- Regra 6: no cabeçalho do email, os campos “Received” são checados a fim de averiguar se o caminho pelo qual a mensagem passou inclui um servidor de email da entidade em nome da qual o email se identifica (C19 da seção 3.1).
- Regra 7: uma busca no corpo da mensagem é feita para encontrar divergências entre o domínio da URL de *login* com o domínio das demais URLs da mensagem (exceto URLs de imagens - característica não documentada). Se alguma divergência for encontrada esta característica é classificada como verdadeira para *phishing*.
- Regra 8: são obtidas informações sobre o domínio da página de *login* através de uma consulta ao serviço de *whois* (C19 da seção 3.1). A mesma busca é feita para os domínios das demais URLs (exceto URLs de imagens). Se houver divergências entre as informações do domínio da página de *login* com o domínio das demais URLs da mensagem, esta característica é verdadeira.
- Regra 9: de maneira semelhante à Regra 7 é feita uma busca por inconsistências entre os domínios das URLs das imagens. Se isto ocorrer esta característica é verdadeira (característica não documentada).
- Regra 10: de maneira semelhante à Regra 8 é feita uma busca por inconsistências entre as informações da consulta ao serviço de *whois* dos domínios das URL's de imagens (C19 da seção 3.1). Se houver alguma inconsistência esta característica é verdadeira para *phishing*.
- Regra 11: esta regra é considerada como verdadeira quando a página web está inacessível. Em caso contrário, esta regra não é aplicada (característica não documentada).

A classificação é feita com base na fórmula $S = \frac{\sum W_i P_i}{\sum W_i X_i}$ onde cada característica possui um peso ajustável W_i e um valor P_i que vai de 0.0 a 1.0, sendo 0.0 aplicado nos casos em que a característica não é aplicável. O *threshold* não é pré-estabelecido (pode ser definido à critério do administrador) e o resultado final (S) é a probabilidade da mensagem ser ou não *phishing*.

Apesar de resultados reportarem uma alta taxa de acerto de 95,72%, algumas características foram mal explicadas/documentadas ou não possuem uma justificativa da sua relevância em relação ao *phishing*. O tamanho da base utilizada para os testes foi muito

limitado, com 81 mensagens de *phishing* e 36 emails legítimos, o que pode tornar questionáveis os resultados. Também não houve separação da base para a realização das etapas de treinamento e teste. Sem essa separação o resultado pode se tornar "viciado", visto que o ajuste da ferramenta é feito com as mesmas mensagens que originaram o percentual de acerto reportado. Além disso, são feitos acessos a informações em fontes externas ao serviço de email, o que pode aumentar demasiadamente o tempo necessário para a análise de cada mensagem.

3.4. Learning to Detect Phishing Emails

Fette e outros utilizaram uma técnica que envolve aprendizagem de máquina com 10 características (C1, C2, C3, C4, C12, C14, C15, C18, C19 e uma não classificada) [Fette, Norman e Anthony 2007]. Nesta abordagem, segundo os autores, a taxa de acerto do classificador chegou a 99,5% quando utilizada em conjunto com uma ferramenta *anti-spam*, em uma base contendo 6950 emails legítimos e 860 phishing de email.

O método de classificação utilizado como referência deste trabalho foi florestas aleatórias (método de classificação que consiste em várias árvores de decisão). Outros classificadores (incluindo SVM) foram testados, mas, segundo os autores, não apresentaram uma diferença estatística muito relevante nos experimentos realizados. Para a característica C18 (seção 3.1), foi utilizado o SpamAssassin sem treinamento (com as regras padrões de detecção) e com treinamento (após aprendizagem numa base de treinamento).

Apesar da alta taxa de acerto do classificador, esta técnica de detecção precisa de 10 características, ferramenta *anti-spam* e precisa fazer consultas a fontes externas de informações (o serviço de *whois*). Isto pode aumentar consideravelmente o tempo de análise de cada mensagem, inviabilizando a proposta em um gateway SMTP que receba um número considerável de mensagens, por exemplo.

3.5. Detection of Phishing Attacks: A Machine Learning Approach

Outro trabalho, desenvolvido por Basnet e seus colegas chegou a uma taxa de 97,99% de acerto no classificador, em uma base contendo 973 phishing de email e 3027 mensagens legítimas, utilizando 16 características (C2, C3, C5, C8, C9, C12, C13, C14, C17 – usando 6

grupos de palavras-chave e C19) e aprendizagem de máquina [Basnet, Mukkamala, e Sung 2008].

O uso de palavras-chave (C17) contabiliza o número destas repetidas no email. O número total de palavras do email é dividido pelo número contabilizado de palavras-chave, e esta razão (freqüência de palavras) é considerada uma característica contínua. Para a aprendizagem de máquina, foram testadas as técnicas SVM, BSVM, Redes Neurais, SOMs e K-Means, sendo que SVM se mostrou mais apropriada para detecção.

Esta abordagem se vale de muitas características e depende da identificação de palavras-chave no corpo do email, o que pode deixar a proposta muito lenta durante a detecção em sistemas reais. Além disso, a característica C17 busca somente conjuntos de palavras geralmente ligadas ao setor financeiro e em inglês. Apesar de este ser o tipo mais comum de *phishing*, estatísticas mostram que a quantidade dessa categoria tem diminuído recentemente, dando lugar a outras categorias (comércio, governamental, etc) [Anti-Phishing Working Group 2007-2008].

3.6. Visão Geral dos Trabalhos Relacionados Baseados em Características do Email

Apesar de várias abordagens utilizarem as mesmas características, nota-se que muitas delas recebem nomes diferentes ou possuem algumas particularidades. Como várias das características foram utilizadas neste trabalho, foram padronizados nomes e descrições sem envolver suas particularidades, conforme foi apresentado nas seções 2.1 e 3.1.

A Tabela 3.1 apresenta uma visão geral das abordagens apresentadas neste capítulo, incluindo características utilizadas em cada abordagem, de acordo com a classificação apresentada nas seções 2.1 e 3.1, total de características e percentual de acerto atingido de acordo com cada autor, usando seus próprios testes.

Tabela 3.1 – Resumo geral dos trabalhos relacionados

Característica		Abordagem/seção			
		3.2	**3.3**	**3.4**	**3.5**
C1		X	X	X	
C2		X	X	X	X
C3				X	X
C4				X	
C5		X			X
C6					
C7					
C8					X
C9					X
C10					
C11		X			
C12				X	X
C13					X
C14				X	X
C15		X			
C16		X			
C17				X	X (6)
C18				X	
C19			X (4)	X	X
Não documentadas			X (5)	X	X
Total de características		6	11	10	16
Percentual de acerto no classificador		96,00%	95,72%	99,50%	97,99%
Tamanho da base	**Phishing**	203	81	860	973
	Não-phishing	0	36	6950	3027

3.7. Outras Técnicas de Prevenção Não-baseadas em Características

Além das abordagens baseadas em características de email, outras técnicas tentam resolver este problema. A seguir serão citados alguns exemplos de técnicas encontradas na literatura, suas vantagens e desvantagens.

3.7.1 Ferramentas de Navegador

Entre as ferramentas de proteção contra phishing, encontram-se as barras de ferramentas, que são componentes adicionais (plugins) que podem ser instalados a parte no navegador. A função dessas ferramentas é determinar se um site visitado é ou não fraudulento, com base em informações extraídas durante a navegação. Após a classificação, o resultado normalmente é apresentado ao usuário através de um ícone na janela do navegador, uma janela de alerta ou bloqueio da navegação. Nesta categoria, muitas propostas já foram desenvolvidas. As barras de ferramenta de navegador normalmente utilizam listas (whitelists e blacklists) para a classificação do site visitado [Netcraft Toolbar 2010], [Herzberg e Gbara 2010], [eBay Account Guart 2010], [Earthlink Toolbar 2010]. A maioria dos desenvolvedores não informa como essas listas são gerenciadas.

Algumas abordagens permitem a participação do usuário para reportar algum endereço suspeito [eBay Account Guart 2010], [Earthlink Toolbar 2010], que posteriormente é analisado e adicionado em alguma lista, se for o caso. Também é comum o uso de algum tipo de heurística [eBay Account Guart 2010], [Earthlink Toolbar 2010] que, como no caso das listas, os desenvolvedores dos produtos dizem utilizar, mas não informam mais detalhes sobre seu uso. Algumas destas ferramentas ainda informam a localização onde o domínio está hospedado [Netcraft Toolbar 2010] e informações sobre sua Autoridade Certificadora (AC) [Herzberg e Gbara 2010].

Estudos demonstram que as barras de ferramentas de navegador atingem resultados insatisfatórios [Cranon, Egelman, Hong e Zhang 2007], [Wu, Miller, Robert e Garfinkel 2006]. Alguns pontos negativos, no uso dessas ferramentas em geral, também podem ser destacados:

- Algumas barras de ferramentas mostram apenas um pequeno ícone, se comparado à janela toda do navegador. Como consequência, o usuário pode não prestar atenção suficiente no

ícone de alerta, voltando toda a sua atenção ao conteúdo da página, sem perceber que está sendo ameaçado [Cranon, Egelman, Hong e Zhang 2007].

- Se a ferramenta alguma vez gerar um falso-positivo, o usuário irá desconfiar da sua eficácia, podendo fazer com que o ele não confie em sua detecção em um caso real de *phishing* [Cranon, Egelman, Hong e Zhang 2007].
- A maioria das abordagens desta categoria toma decisões com base em whitelists e blacklists. O problema desta forma de detecção é que as organizações de combate ao phishing se encontrarão em uma corrida contra os atacantes, ou seja, um caso semelhante ao das companhias que desenvolvem soluções antivírus [Cranon, Egelman, Hong e Zhang 2007] .
- Se considerarmos os ataques que ocorrem através de algum canal de comunicação como email, a tentativa de bloqueio é feita somente depois que a vítima foi persuadida a clicar no *link*, ou seja, o ataque já está mais próximo do seu objetivo.
- Caso a máquina cliente esteja comprometida, ou o servidor ou a ferramenta possua vulnerabilidades, o resultado pode não ser confiável.

Além das barras de ferramentas de navegador tradicionais para detecção de *phishing*, há também alguns componentes adicionais que possibilitam o re-uso de senha [Kirda e Kruegel, 2005], [Wu, Miller e Little, 2006]. Em geral, elas possibilitam o armazenamento de informações que são utilizadas pelo usuário (senhas, identificação, etc) juntamente com alguma informação que forneça uma ligação direta com o site no qual costumam ser inseridas (Ex: nome do domínio). Quando é verificado que as informações inseridas não conferem com o domínio para o qual serão enviadas, alguma ação é tomada para evitar seu envio indevido. Uma limitação dessa abordagem é o oferecimento de proteção apenas contra o *phishing* que objetiva o roubo de informações da vítima, não protegendo contra um ataque que objetiva a instalação de um *malware*, por exemplo.

3.7.2 Ferramentas de Email não-baseadas em características

Além das ferramentas de navegador, também há abordagens de detecção de *phishing* de email que não são baseadas em características. Em [Castillo, Iglesias e Serrano, 2007], os autores propuseram uma abordagem para classificação de emails baseada em três filtros: um

filtro Bayesiano, que classifica o conteúdo do corpo do email (classificação textual); um filtro baseado em regras não-textuais; um filtro baseado na emulação de acessos fictícios, que classifica as respostas dos sites referenciados em *links* contidos nos emails. Com base nesses filtros, um email pode ser classificado como legítimo ou fraudulento. A ferramenta construída foi feita para atuar no lado cliente. Afim de evitar falsos positivos, se em algum ponto de decisão houver carência de informações, o classificador aloca a mensagem a uma categoria que permita que futuramente o sistema faça uma análise mais aprofundada.

Primeiramente, o filtro de conteúdo é utilizado com a finalidade de detectar se o email tem um contexto financeiro/econômico ou não, visto que a maioria dos casos de phishing é direcionada a esse setor. Na primeira classificação (classificação de conteúdo textual), é utilizado um classificador para o assunto do email e outro para o corpo da mensagem. Na fase de treinamento do filtro, a probabilidade de cada palavra se enquadrar em cada categoria é estimada, criando um vocabulário de palavras com suas respectivas probabilidades. O texto é classificado baseando-se na categoria que tem o maior índice de probabilidade.

Em seguida, o filtro baseado em regras é acionado, classificando os emails anteriormente classificados como de conteúdo financeiro, nas categorias Legítimo, Fraude ou Suspeito. Esta segunda classificação é feita com base em três regras. A primeira verifica se o corpo da mensagem não contém formulários, imagens ou *links*. Em caso verdadeiro, o email é classificado como Legítimo. A segunda regra determina que emails que solicitam informações diretamente, através de um formulário no corpo da mensagem não é seguro, e estes são classificados como Fraude. Uma terceira regra é criada para classificar emails que contém imagens ou *links.* Como há emails legítimos que também contém links e imagens, e, devido à falta de conhecimento da ferramenta para poder tomar uma decisão, a terceira regra então os enquadra na categoria Suspeito.

Finalmente, os emails que foram classificados como Suspeito, passam pelo terceiro filtro, que analisa as respostas obtidas, através acessos feitos de forma fictícia a esses sites, categorizando-os como Legítimos ou Fraude. Esta última classificação consiste em duas etapas:

1) Uma busca extrai os *links* contidos no email. Posteriormente é realizada uma pesquisa em um mecanismo de busca (Ex: Google) pelos sites referidos pelos *links*. Esta busca é realizada

assumindo o princípio de que sites fraudulentos não ficam ativos por muito tempo, logo a busca não trará resultados. Se o site for encontrado na busca, é verificado se ele utiliza uma conexão segura (HTTPS). Quando a conexão não utiliza HTTPS, o email é classificado como Fraude. Se a página utiliza HTTPS, é verificado o tipo de formulário contido nas páginas referenciadas por esses links (Ex: Senha). Se não for solicitado nenhum tipo de informação sensível, o email é enquadrado na categoria Legítimo. Caso contrário, a análise parte para a segunda etapa.

2) Um simulador preenche os formulários utilizando dados fictícios, submetendo-os ao site a fim de obter uma resposta. A resposta é analisada para classificar o email, partindo do princípio de que sites fraudulentos não retornam nenhuma mensagem de erro, pois sua principal função é apenas coletar informações das vítimas.

Quanto às técnicas utilizadas pela ferramenta, podem ser destacadas as seguintes desvantagens:

- Na primeira etapa, é utilizado um filtro de conteúdo que classifica a mensagem como de conteúdo financeiro/econômico ou não. As mensagens que não possuem esse tipo de conteúdo não passam para a próxima etapa de classificação. Como o *phishing* não se resume apenas ao roubo de dinheiro das vítimas, mensagens fraudulentas sem conteúdo financeiro também existem [APWG 2008]. Isto significa que, incluir apenas as mensagens com o contexto financeiro na checagem, resultaria em um número alto de falsos negativos.
- Ainda na primeira etapa, a classificação do conteúdo do email é feita através da probabilidade de ocorrência de alguns termos no assunto e no corpo da mensagem. Isto se tornaria uma tarefa complexa para ser implementada a fim de suportar diversos idiomas. Além disso, se o corpo da mensagem se resumir a uma imagem, a avaliação seria feita apenas com base no assunto, o que pode prejudicar a classificação.
- Quando é feita a simulação de envio de dados através de formulários, não há nada que comprove a autenticidade da resposta. Os ataques costumam sofrer algumas alterações de comportamento para que não sejam mais detectados por ferramentas de filtragem. Pode ser retornado uma resposta de acesso concedido, intencionalmente, de forma que este tipo

de simulação se torne ineficiente contra ataques de *phishing* para coleta de informações através de formulários.

Em [Chandrasekaran, Chinchani e Upadhyaya 2006], é apresentada uma abordagem para detecção de *phishing* simulando as ações do usuário. Atua entre o MTA (Mail Transfer Agent) e o MUA (Mail User Agent), processando cada mensagem recebida. A ferramenta analisa o conteúdo da mensagem recebida, em busca de *links* e formulários HTML. Se a mensagem possuir formulários solicitando informações sensíveis, ela é classificada como maliciosa. Se possuir *links* no corpo da mensagem, eles são verificados através de uma simulação de acesso a eles. Se a mensagem não possuir nenhuma das duas propriedades, a análise é encerrada.

A simulação ocorre de forma semelhante a [Castillo, Iglesias e Serrano, 2007]. É verificado se as páginas que foram acessadas através dos *links* contêm algum tipo de formulário. Quando há formulários, eles são preenchidos com dados fictícios. Uma vez que um site fraudulento supostamente não conseguiria distinguir uma informação válida de uma informação falsa, sua resposta será a mesma para ambos os casos (aceitará as informações). A simulação representa uma entidade falsa (usuário fantasma) que insere dados aleatórios nos formulários, com o uso de *honeytokens* [Spytzner 2008]. As respostas do site são encaminhadas ao sistema de decisão para uma posterior análise. Uma vez que a autenticidade das páginas contidas nos *links* é confirmada, o usuário pode ver a mensagem.

Como já foi comentado anteriormente, a simulação de envio de informações através de formulários, pode ser facilmente burlada pelos atacantes, visto que não existe algo que comprove a autenticidade da resposta. Além do mais, esta abordagem é focada unicamente ao tipo de *phishing* que tenta coletar informações das vítimas através de formulários, sendo assim, limitada.

3.8. Considerações

Dentre vários tipos de abordagens para a detecção de *phishing* (barras de navegador, re-uso de senha, etc.) as baseadas em características da mensagem tem se mostrado promissoras, pois podem explorar diretamente as técnicas utilizadas pelos estelionatários que disseminam *phishing* de email. Contudo, a maior parte dessas abordagens foca apenas na

obtenção de alta taxa de acerto no classificador, sem se preocupar com a confiabilidade do resultado (que pode ser avaliada através da taxa de falsos-positivos e falsos-negativos e área sob a curva) e nem com o desempenho do classificador.

Quanto ao desempenho, é sabido que os filtros de email tem sido o principal gargalo nos *gateways* de email, podendo gerar uma fila de milhares de mensagens a serem analisadas. Essas propostas, em geral não tentam aperfeiçoar seu desempenho, que pode ser comprometido devido ao número excessivo de características utilizadas. Outra limitação encontrada literatura e que também interfere no desempenho é a consideração das características sem avaliar se realmente são fundamentais para identificar *phishing*, aumentando o tamanho do vetor de características sem necessidade real.

Capítulo 4

Obtenção do *Modelo do Adversário*

Este capítulo descreve em detalhes como o trabalho foi realizado, relatando todos os passos que levam a criação de uma metodologia para encontrar os modelos do adversário até a obtenção dos resultados e testes de desempenho.

4.1. Proposta

Já no início do projeto, foi definido que os esforços seriam concentrados na obtenção das características que melhor representam *phishing* (modelo do adversário). Assim sendo, seria necessário uma espécie de dicionário comum de características de *phishing*, pois nos trabalhos relacionados vários autores descrevem a mesma característica usando termos diferentes, ou então em outras vezes sem explicar adequadamente porque uma característica foi assim considerada.

O *modelo do adversário* pode ser definido como sendo as características essenciais (atributos distintos) para detectar *phishing*. É muito comum um profissional de TI ou responsável pela segurança da informação julgar que uma nova técnica de engenharia social é uma característica de *phishing*. Na verdade sem o uso da técnica de avaliação do conjunto de características apropriada não se pode fazer tal afirmação, pois esta pode ser um subconjunto de outra característica, ou quando avaliada em conjunto com as outras pode não ser distinta. Em casos mais complexos, a assim considerada nova característica pode até confundir a técnica de detecção em uso, tornando-a ineficiente.

Após uma consulta à literatura técnica e avaliação da base de mensagens de email, foi identificado um total de 19 características (detalhadas na seções 2.1 e 3.1). Para cada uma delas, foi procurada uma explicação plausível do ponto de vista de *phishing*, ou seja, a característica só é assim considerada se realmente usa artifício para enganar o usuário ou para agregar informações que facilitem a identificação do *phisher*.

Para a obtenção do modelo uma das premissas foi não consultar nenhum serviço/recurso externo (não nativo) ao sistema de email (como os que são descritos nas características de C17 a C19), visto que tal abordagem poderia causar um tempo de espera considerável quando muitas mensagens precisam ser avaliadas, e sua utilização pode não ser tão eficiente quanto se espera. Por exemplo, uma consulta ao serviço de *whois* só é eficiente se o domínio for recente, porém, os *phishers* podem se esconder sob domínios consolidados, exatamente para não serem descobertos por este tipo de consulta. Além disso, algumas características, apesar de justificáveis, possuíam pouca ou nenhuma ocorrência em nossa base de emails (e.g. características C12 e C13).

A avaliação realizada foi baseada em um conjunto de 11 características (C1 a C11) que se adequaram às premissas iniciais. Se o número de características utilizadas for relativamente pequeno, a busca pelo conjunto de características mais significativo (*modelo do adversário*) pode ser realizada através da técnica de força bruta ao invés de um algoritmo de busca [Kudo e Sklansky, 2000].

Como comentado anteriormente, a cada trabalho proposto para detectar *phishing* mais eficientemente, uma nova técnica para driblar os filtros é criada pelos *phishers*. Por isso, um dos objetivos foi criar uma sistematização do procedimento de obtenção do *modelo do adversário*, de tal modo que seja fácil reavaliar as características utilizadas em determinado momento e redefinir o modelo, mantendo o sistema de detecção de *phishing* sempre atualizado sem muito retrabalho.

O impacto no desempenho do classificador, causado pelo uso de um número desnecessário de características, também foi objeto de avaliação deste trabalho, principalmente porque quando o mesmo for colocado para funcionar em ambiente real, a extração das características pode se tornar onerosa para o sistema.

Consideramos também que o percentual de acerto do classificador não é um indicador suficiente para mostrar sua eficácia, assim foram adotadas curvas ROC para avaliar a confiabilidade da taxa de acertos do classificador.

Esta proposta foi executada seguindo os seguintes estágios: preparação das bases de treinamento e teste, teste de força bruta (para avaliar o conjunto de combinações de características que dão os melhores resultados), geração das curvas ROC (para avaliar a confiabilidade dos resultados dos classificadores), obtenção do *modelo do adversário* e avaliação de desempenho.

4.2. Preparação das bases de treinamento e teste

A base de *phishing* foi construída selecionando e rotulando-se manualmente mensagens de email a partir de uma base de mensagens filtradas no servidor SMTP da PUC-PR no período de janeiro de 2007 a agosto de 2009. A base de emails legítimos (não-*phishing*) foi construída a partir de amostras de emails reais. Foram incluídas mensagens de confirmação de cadastro, cartões de crédito e compras online na classe não-*phishing*, pois há mensagens com *phishing* muito parecidas com estas. Ou seja, para cada mensagem rotulada como *phishing*, buscou-se uma equivalente para a base não-*phishing*, mas nem sempre conseguimos tal amostra.

Para a montagem da base de *phishing* foram incluídas somente mensagens distintas (não-repetidas). Para alguns casos de *phishing*, o conteúdo da mensagem era o mesmo, mas com um link diferente. Como as características contidas nos links podem ser um fator determinante na classificação do email, estas mensagens também foram incluídas, sendo o link um fator de diferenciação entre as mesmas.

O tamanho total da base foi de 900 emails, sendo que metade para *phishing* e metade para não-*pishing*. Depois de selecionadas as mensagens, a base foi dividida em duas, sendo uma metade para a etapa de treinamento/aprendizagem e a outra para a etapa de teste. Ou seja, para cada uma das duas etapas foram utilizadas 225 mensagens de *phishing* e 225 para não-*phishing*.

Das 11 características utilizadas, oito são binárias (C1, C2, C3, C5, C6, C7. C10, C11).

A tabela 4.1 mostra a distribuição percentual de ocorrência das características binárias

em nossas bases de mensagens de *phishing e* não-*phishing*. As características C4, C8 e C9 são mostradas separadamente em outras tabelas por serem contínuas (podem assumir valores não binários). Durante o treinamento da base, o SVM executa a normalização das características contínuas, enquadrando-as em valores entre -1 e 1.

Tabela 4.1 – Estratificação das mensagens de acordo com as características

Característica		**C1**	**C2**	**C3**	**C5**	**C6**	**C7**	**C10**	**C11**
Ocorrência do total (%)	**Phishing**	29,11	20,66	96,66	100	68,44	3,55	6	64,44
	Não-phishing	0	0	68,88	45,33	3,55	0	0	16,88

Na leitura dos percentuais deve-se observar que uma mesma mensagem pode ocorrer simultaneamente em mais que uma característica de *phishing* e, portanto a soma dos percentuais não resulta 100%. Ou seja, na tabela 4.1 uma mesma mensagem pode estar contabilizada em mais de uma coluna.

As tabelas 4.2, 4.3 e 4.4 mostram o intervalo de variação de valores para as características contínuas C4, C8 e C9, respectivamente, incluindo o percentual de ocorrência nas bases de *phishing* e não-*phishing*.

Tabela 4.2 – Estratificação das mensagens e valores da característica C4

Número de pontos na URL	**< 2**	**2**	**3**	**4**	**> 4**
Phishing	6,44	27,78	24,22	21,11	20,45
Não-phishing	33,55	28,67	31,56	6,22	0

Na tabela 4.2 observa-se que a diferença maior entre as mensagens ocorre quando C4 é menor que 2 pontos na URL, pois há 27,11% mais mensagens rotuladas como não-*phishing* do que *phishing*. Já quando o número de pontos é maior que quatro, o percentual de *phishing* é de 20,45% das mensagens. De maneira geral a grande maioria das mensagens de *phishing* tem mais que 3 pontos na URL, e isto representa 35,34% das mensagens (21,11 + 20,45 – 6,22). Nos casos em que há mais de uma URL no email, considera-se apenas a URL com maior número de pontos.

Tabela 4.3 – Estratificação das mensagens e valores da característica C8

Número de domínios na mesma URL	1	2	3	4	5
Phishing	82,67	14,89	1,78	0,44	0,22
Não-phishing	100	0	0	0	0

Observa-se na tabela 4.3 que a característica C8, quando é equivalente a dois ou mais domínios em uma mesma URL, corresponde a 17,33% das mensagens de *phishing* de email.

Na tabela 4.4 não há nenhum caso discriminante, porém conforme comentado anteriormente pode ser que a característica C9 se torne importante se for combinada com as demais.

Tabela 4.4 – Estratificação das mensagens e valores da característica C9

Número de subdomínios na mesma URL	0	1	>1
Phishing	71,33	21,56	7,11
Não-phishing	74,22	12	13,78

4.3. Avaliação dos conjuntos de características através de técnica de seleção por “Força Bruta”

Afim de identificar os conjuntos de combinações de características mais significativas para a detecção de *phishing*, foram realizados testes usando a técnica de força bruta (é feita a analise combinatória de todos os conjuntos de combinação de características). Das 11 características resultaram 2.036 conjuntos de combinações diferentes, que foram avaliadas registrando suas respectivas taxas de acerto e confiabilidade para posterior análise.

É importante ressaltar que o método da força bruta é eficiente, porém viável somente para um pequeno numero de características. Para um número maior de características algum algoritmo de busca deve ser utilizado [Kudo e Sklansky, 2000].

A partir dos resultados da força bruta, são mostrados os melhores percentuais de acerto (Tabela 4.5). Observa-se que a partir da combinação de 9 características o percentual de acerto diminui, tornando o conjunto de características ineficiente.

Tabela 4.5 – Melhor percentual de acerto de acordo com o nº de características

Número de características	2	3	4	5	6	7	8	9	10	11
Acerto (%)	77,78	77,78	86,66	90,66	92,22	94,44	94,89	94,22	94,22	93,78

Pode-se observar que com apenas duas características foi possível atingir no máximo um percentual de acerto de 77,78% e o mesmo resultado foi atingido com 3 características. Com 4 características, o percentual sobe consideravelmente (quase 9%), chegando a 86,66. A partir da combinação de 6 características, o percentual fica acima de 90%, observando-se o limite máximo (94,44%) com 8 características. Acima da nona característica o resultado da classificação não melhora sua taxa de acerto.

4.4. Análise das ROCs e AUCs

As curvas ROC tiveram muita importância na avaliação dos classificadores, visto que permitiram a identificar não apenas da taxa de acerto do classificador SVM, mas também das taxas de falsos positivos (FP - *false positives*) e verdadeiros positivos (TP - *true positives*). Esta análise permite identificar a melhor relação entre verdadeiros positivos e falsos positivos.

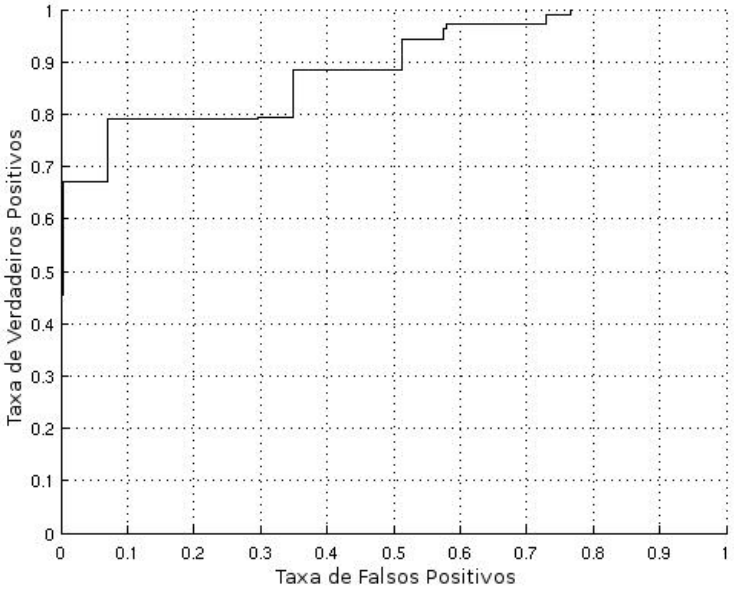

Figura 4.1 – Curva ROC do classificador C4.C5.C9

Observando a Figura 4.1 com 3 características é possível observar que o classificador pode atingir uma taxa de 67% TP com 0% FP. Já para 79% de TP haverá 7% de FP ou 88% de TP com 35% FP.

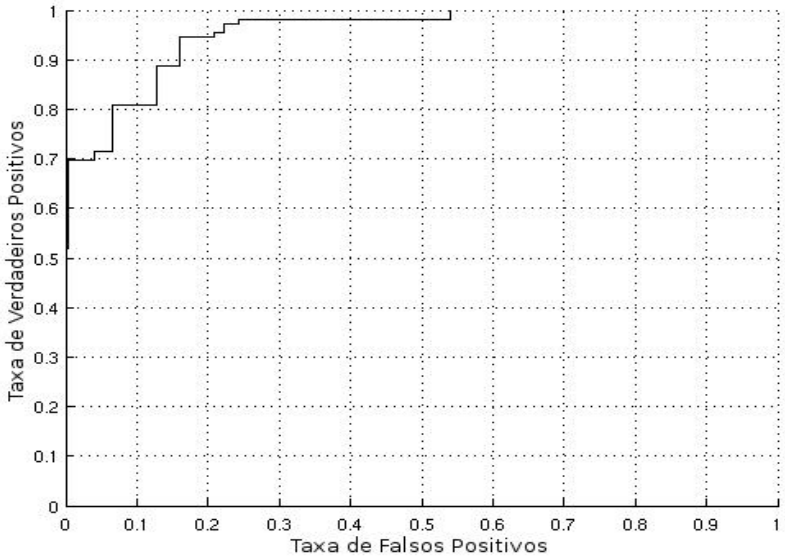

Figura 4.2 – Curva ROC do classificador C4.C5.C6.C9

Acrescentando apenas uma característica ao conjunto (Figura 4.1), agora 4C, nota-se a melhora nos resultados de TP e FP através das curvas ROC (Figura 4.2). Ou seja, a taxa de TP vai para 70% com FP igual a 0%, para 81% TP com 7% de FP ou para 89% de TP com 12% de FP. Pode-se ainda optar por um percentual de 96% de TP, mas com taxa de FP de 35%.

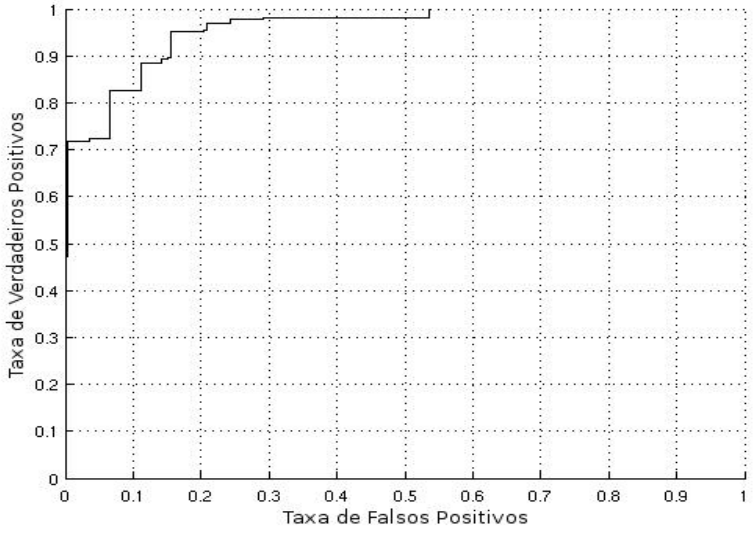

Figura 4.3 – Curva ROC do classificador C3.C4.C5.C6.C9

Com cinco características, para 0% de FP tem 72% de TP, com 7% de FP é possível chegar a 83% TP e com 11% de FP se alcança 89% de TP (Figura 4.3).

Quando são utilizadas todas as características juntas, se obtém a melhor relação TP x FP, ou seja, 90% de TP com menos de 5% de FP (Figura 4.4).

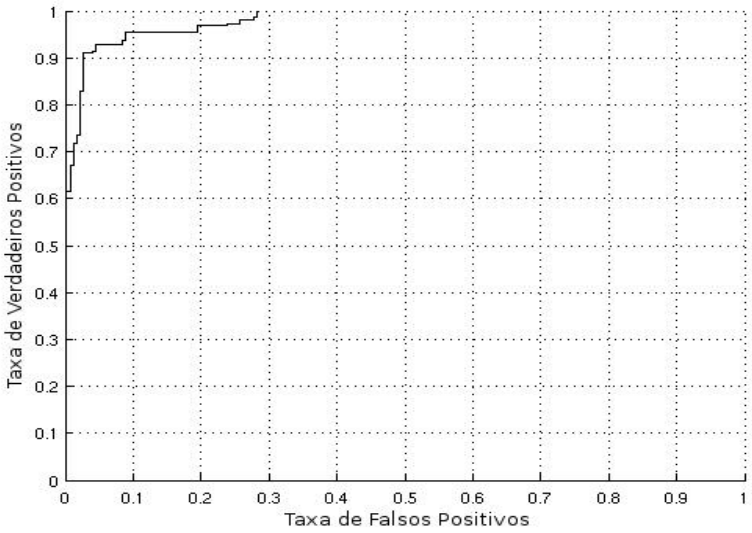

Figura 4.4 – Curva ROC do classificador com todas as características

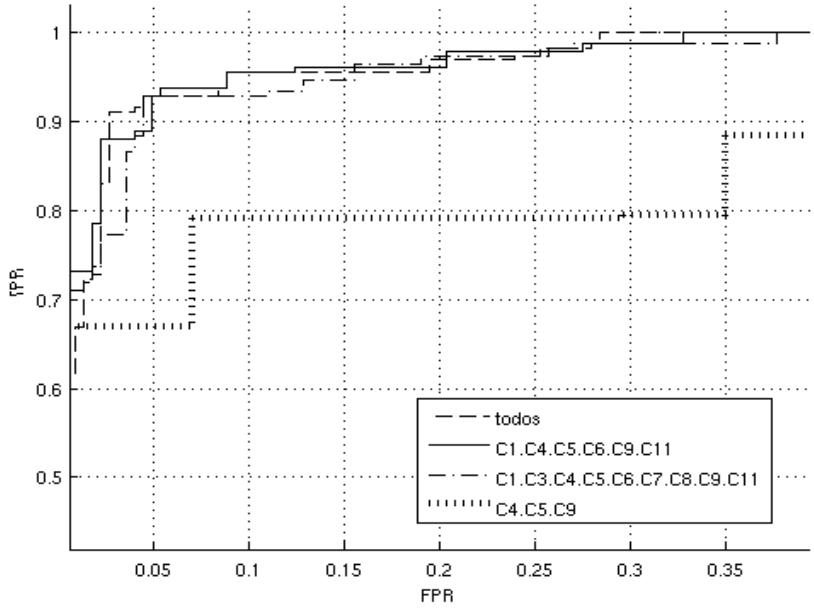

Figura 4.5 – Curvas ROC comparativas entre os melhores e o pior classificador

A Figura 4.5 ilustra a diferença entre os melhores e o pior classificador encontrado através das curvas ROC. Os melhores casos (6C, 9C e 11C) chegaram a resultados muito parecidos.

Outra forma de identificar os melhores classificadores é através das AUCs (*Area Under the Curve*). Pode-se ainda utilizá-las em combinação com as curvas ROCs para ajudar na escolha do melhor classificador. Em geral a melhor AUC é aquela mais próxima da unidade. A Tabela 4.6 mostra as AUCs dos melhores classificadores para os seus respectivos números de características.

Tabela 4.6 – AUCs dos melhores classificadores

Número de características	Valor da AUC
3	0.890289
4	0.955081
5	0.958537
6	0.980760
7	0.978943
8	0.951920
9	0.976770
10	0.975743
11	0.980247

No caso de 4C e 5C as curvas ROCs (Figuras 4.2 e 4.3) mostraram que as relações entre FP x TP foram muito parecidas, sendo em torno de 81% TP para 7% de FP. Outra situação parecida acontece com 89% de TP e em torno de 11% de FP para os dois conjuntos.

As AUCs desses classificadores poderiam ser utilizadas para definir qual seria o melhor entre os dois. Entretanto, nota-se que os valores são muito próximos (0,955 para 4C e 0,958 para 5C) e, portanto, não se justifica o uso de uma característica a mais no detector de *phishing* se não há melhora na confiabilidade dos resultados do mesmo.

As AUCs são úteis para escolher entre os melhores classificadores entre as curvas ROCs de cada conjunto. Como os conjuntos 6C, 9C e 11C alcançaram resultados muito parecidos, os valores das AUCs podem ajudar na escolha do melhor classificador entre os três.

Nesse caso os melhores foram 6C (0.980760) e 11C (0.980247), sendo que 11C possui uma AUC similar a 6C, porém usa quase o dobro de características.

Com a análise das ROCs e das AUCs conclui-se que não se justifica o uso de mais do que seis características (6C), visto que as diferenças nos resultados foram mínimas e a adição de novas características geralmente representa um aumento no custo computacional necessário para realizar a avaliação/classificação dos emails. A seção 4.9 trata exclusivamente da questão do desempenho.

4.5. O Modelo do Adversário

O Modelo do Adversário representa o perfil de *phishing* num dado momento, observando-se que os *phishers* estão sempre mudando as suas técnicas com o objetivo de não serem detectadas pelos respectivos sistemas de detecção/filtros. Assim, o ideal é que os filtros possuam uma gama de possibilidades, no intuito de se adaptar às variações no conjunto de características que os mesmos usam para identificar *phishing*. Perseguindo este objetivo observou-se nos vários conjuntos de combinações resultantes da avaliação por força bruta as características que apareciam recorrentemente entre os melhores resultados nas taxas de acerto do classificador. A seguir é descrito com mais detalhes o processo utilizado para obtenção dos *modelos do adversário*.

Com o uso da técnica de força bruta foi possível identificar um percentual de acerto do classificador de no mínimo 77,78% (Tabela 4.5) em 20 dos 165 conjuntos com três características (3C).

Entre os 20 melhores resultados observou-se que as características que mais apareceram combinadas foram C4, C5, C6 e C9 – este grupo de características foi denominado *candidato à modelo do adversário*. Nos 20 conjuntos com melhor percentual de acerto na classificação havia 10 onde apareciam pelo menos duas características *candidato à modelo do adversário* juntas. Em quatro desses conjuntos havia três características *candidato à modelo do adversário* juntas, 3C = {C4,C5,C6}, {C4,C5,C9}, {C4,C6,C9} e {C5,C6,C9}.

Estes quatro conjuntos podem ser considerados como possíveis *modelos do adversário* de três características, se a presença destas características se mantiver entre os melhores percentuais de acerto na análise de outras combinações com mais características juntas.

A Tabela 4.7 apresenta os 10 conjuntos de 3C que tem juntas pelo menos duas das características do *candidato à modelo do adversário* e o seu respectivo número de ocorrências na base de emails.

Tabela 4.7 – Melhores conjuntos envolvendo o *candidato à modelo do adversário* para 3C

Conjunto	Taxa de acerto do classificador
C1.**C4.C5**	77,78%
C2.**C4.C5**	77,78%
C4.C5.*C10*	77,78%
C4.C5.*C11*	77,78%
C4.C5.C6	77,78%
C4.C5.*C7*	77,78%
C4.C5.*C8*	77,78%
C4.C5.C9	77,78%
C4.C6.C9	77,78%
C5.C6.C9	77,78%

Observando a tabela 4.7 nota-se que as características C4 e C5 aparecem juntas em oito dos dez conjuntos de 3C, assim foi investigado o percentual de acerto do classificador para duas características combinadas e obteve-se o resultado da tabela 4.8.

Tabela 4.8 – Melhores conjuntos envolvendo o *candidato à modelo do adversário* para 2C

Conjunto	Taxa de acerto do classificador
C4.C5	77,78%
C5.C8	77,78%

Para quatro características (4C) foi considerado o percentual mínimo de 77,78%, o que resultou em 77 conjuntos diferentes. Novamente foi feita uma filtragem desses conjuntos, selecionando aqueles que possuíam pelo menos três características do *candidato à modelo do adversário* juntas. O resultado foi 11 conjuntos, sendo que o melhor desses é formado pelo

candidato à modelo do adversário com um percentual de acerto do classificador de 86,66% (Tabela 4.9).

Tabela 4.9 – Melhores conjuntos envolvendo o *candidato à modelo do adversário* para 4C

Conjunto	Taxa de acerto do classificador
C4.C5.C6.C9	86,66%
C2.**C4.C5.C9**	77,78%
C4.C5.C6.*C11*	77,78%
C4.C5.C6.*C7*	77,78%
C4.C5.*C7*.**C9**	77,78%
C4.C5.C9.*C10*	77,78%
C4.C5.C9.*C11*	77,78%
C5.C6.*C7*.**C9**	77,78%
C5.C6.C9.*C11*	77,78%
C3.**C4.C6.C9**	77,78%
C4.C6.C9.*C11*	77,78%

Com cinco características (5C) o melhor conjunto chegou a 90,66% mas continha somente duas características do *candidato à modelo do adversário* juntas. Consideramos este caso uma exceção, pois o segundo melhor resultado, com 87,33% (C3.C4.C5.C6.C9) de acerto do classificador, se apresenta com todas as características do *candidato à modelo do adversário* juntas (Tabela 4.10).

Tabela 4.10 - Melhores conjuntos envolvendo o *candidato à modelo do adversário* para 5C

Conjunto	Taxa de acerto do classificador
C3.**C5.C6**.*C10*.*C11*	90,66%
C3.**C4.C5.C6.C9**	87,33%
C4.C5.C6.C9.*C11*	85.55%
C4.C5.C6.C7.**C9**	85,11%
C3.**C4.C6.C9**.*C11*	84,00%

Conjunto	Taxa de acerto do classificador
C2.*C3*.**C4.C5.C9**	80,44%
C3.**C4.C5.C9**.*C10*	80,22%

Com seis características (6C) houve 33 conjuntos com percentual de acerto do classificador acima de 80%, sendo que em doze destes apareciam as características do *candidato à modelo do adversário* (Tabela 4.11), incluindo o conjunto com o melhor percentual de acerto.

Tabela 4.11 - Melhores conjuntos envolvendo o *candidato à modelo do adversário* para 6C

Conjunto	Taxa de acerto do classificador
C1.**C4.C5.C6.C9**.*C11*	92.22%
C2.**C4.C5.C6.C9**.*C11*	90.88%
C1.C2.**C4.C5.C6.C9**	88.88%
C1.*C3*.**C4.C5.C6.C9**	88.88%
C3.**C4.C5.C6.C9**.*C10*	87.33%
C4.C5.C6.*C7*.**C9**.*C11*	85.33%
C4.C5.C6.C9.*C10*.*C11*	85.11%
C3.**C4.C5.C6**.*C7*.**C9**	84.66%
C1.**C4.C5.C6**.*C8*.**C9**	84.44%
C1.**C4.C5.C6.C9**.*C10*	84.44%
C2.**C4.C5.C6**.*C7*.**C9**	84.22%
C2.**C4.C5.C6.C9**.*C10*	84.22%

A partir de sete características (7C) o número de combinações acima de 80% aumentou consideravelmente. Por esta razão, a partir desta fase da análise foram considerados apenas os conjuntos que atingiram um percentual superior a 90% (num total de 10 conjuntos – Tabela 4.12).

As características do *candidato à modelo do adversário* estiveram presentes em todos os melhores resultados para 7C. Nota-se ainda que o conjunto com melhor taxa de acerto do

classificador de 6C (C1.C4.C5.C6.C9.C11) é um subconjunto do melhor conjunto de 7C (**C1.C4.C5.C6**.C8.**C9**.**C11**). Assim, pode-se dizer que há também uma forte vinculação entre os conjuntos 6C e 7C, pois este subconjunto está presente em metade dos melhores resultados de 7C.

Tabela 4.12 - Melhores conjuntos envolvendo o *candidato à modelo do adversário* para 7C

Conjunto	Taxa de acerto do classificador
C1.**C4.C5.C6**.*C8*.**C9**.*C11*	94,44%
C1.**C4.C5.C6**.*C7*.**C9**.*C11*	94,22%
C1.*C3*.**C4.C5.C6.C9**.*C11*	93,78%
C1.*C2*.**C4.C5.C6.C9**.*C11*	93,78%
C4.C5.C6.*C7*.C8.**C9**.*C11*	92,68%
C2.**C4.C5.C6**.*C7*.**C9**.*C11*	92,44%
C2.**C4.C5.C6**.*C8*.**C9**.*C11*	92,44%
C1.**C4.C5.C6.C9**.*C10*.*C11*	92,22%
C3.**C4.C5.C6**.*C7*.**C9**.*C11*	90,89%
C2.*C3*.**C4.C5.C6.C9**.*C11*	90,68%

Com oito características (8C) o melhor resultado atingiu a taxa de acerto de 94,89% no classificador e o comportamento observado em sete características se repetiu. Ou seja, o conjunto com maior percentual de acerto no classificador de 8C contém o subconjunto de melhor resultado em 7C (C1.**C4.C5.C6**.C8.**C9**.C11). Ocorreram 18 conjuntos com percentual de acerto do classificador acima de 90%, sendo que apenas um não é um superconjunto das características do *candidato à modelo do adversário*.

Tabela 4.13 - Melhores conjuntos envolvendo o *candidato à modelo do adversário* para 8C

Combinação	Taxa de acerto do classificador
C1.C3.**C4**.**C5**.**C6**.C8.**C9**.C11	94,89%
C1.**C4.C5.C6**.C8.**C9**.C10.C11	94,44%
C2.C3.**C4.C5.C6**.C7.**C9**.C11	92,44%

Combinação	Taxa de acerto do classificador
C2.C3.**C4.C5.C6.**C8.**C9**.C11	92,44%
C2.**C4.C5.C6**.C7.C8.**C9**.C11	92,44%
C1.C2.**C4.C5.C6**.C8.**C9**.C11	94,22%
C1.C2.**C4.C5.C6.C9**.C10.C11	94,22%
C1.C3.**C4.C5.C6**.C7.**C9**.C11	94,22%
C1.**C4.C5.C6**.C7.C8.**C9**.C11	94,22%
C1.C2.C3.**C4.C5.C6.C9**.C11	93,78%
C1.C2.**C4.C5.C6**.C7.**C9**.C11	93,78%
C1.**C4.C5.C6**.C7.**C9**.C10.C11	93,78%
C1.C3.**C4.C5.C6.C9**.C10.C11	93,78%
C2.**C4.C5.C6**.C8.**C9**.C10.C11	93,11%
C3.**C4.C5.C6**.C7.C8.**C9**.C11	92,44%
C3.**C4.C5.C6**.C7.**C9**.C10.C11	90,67%
C2.**C4.C5.C6**.C7.**C9**.C10.C11	90,22%

Com nove características (9C) o melhor resultado atingiu a taxa de acerto de 94,22% no classificador e o comportamento observado em 8C se repetiu. Ocorreu quatorze conjuntos com 9C, sendo que apenas um deles não continha as características do *candidato à modelo do adversário* como subconjunto.

Tabela 4.14 - Melhores conjuntos envolvendo o *candidato à modelo do adversário* para 9C

Conjunto	Taxa de acerto do classificador
C1.C2.C3.**C4.C5.C6**.C8.**C9**.C11	94,22%
C1.C3.**C4.C5.C6**.C7.C8.**C9**.C11	94,22%
C1.C3.**C4.C5.C6**.C8.**C9**.C10.C11	94,22%
C1.**C4.C5.C6**.C7.C8.**C9**.C10.C11	94,22%
C1.C2.**C4.C5.C6**.C7.**C9**.C10.C11	94,00%
C1.C2.C3.**C4.C5.C6**.C7.**C9**.C11	93,78%
C1.C3.**C4.C5.C6**.C7.**C9**.C10.C11	93,78%
C1.C2.**C4.C5.C6**.C8.**C9**.C10.C11	93,78%

Conjunto	Taxa de acerto do classificador
C1.C2.C3.**C4.C5.C6.C9**.C10.C11	93,56%
C1.C2.**C4.C5.C6**.C7.C8.**C9**.C11	92,89%
C2.C3.**C4.C5.C6**.C7.C8.**C9**.C11	92,44%
C2.C3.**C4.C5.C6**.C8.**C9**.C10.C11	92,44%
C2.**C4.C5.C6**.C7.C8.**C9**.C10.C11	92,44%

Com dez características (10C) o percentual de acerto no classificador ficou em 94,22%. Seis conjuntos atingiram percentuais de acerto do classificador acima de 90%, todos contendo as características do *candidato à modelo do adversário*.

Tabela 4.15 - Melhores conjuntos envolvendo o *candidato à modelo do adversário* para 10C

Conjunto	Taxa de acerto do classificador
C1.C3.**C4.C5.C6**.C7.C8.**C9**.C10.C11	94,22%
C1.C2.C3.**C4.C5.C6**.C8.**C9**.C10.C11	94,00%
C1.C2.C3.**C4.C5.C6**.C7.C8.**C9**.C11	93,78%
C1.C2.C3.**C4.C5.C6**.C7.**C9**.C10.C11	93,78%
C1.C2.**C4.C5.C6.**C7.C8.**C9**.C10.C11	93,78%
C2.C3**.C4.C5.C6**.C7.C8.**C9**.C10.C11	92,44%

É importante levar em consideração que as características do *candidato à modelo do adversário* estiveram presentes em todos os melhores resultados a partir de seis características, o que confirma a relevância de tal conjunto.

A Tabela 4.16 é um resumo de todos os conjuntos com as melhores taxas de acerto no classificador para as características de 2C a 11C. Com as onze características juntas o percentual de acerto no classificador ficou em 93,77%. Em linhas gerais o que se observa é que a partir de 6C o percentual de acerto do classificador praticamente estabiliza em torno de 94%, com desvio padrão de cerca de 1%. Em todos os casos, a exceção de 6C, o conjunto em

avaliação contém o conjunto de melhor taxa de acerto no classificador da avaliação anterior (e.g. 7C contém 6C, conforme indicado na tabela 4.12).

Tabela 4.16 – Resumo do melhor conjunto envolvendo o *candidato à modelo do adversário*

Número de características	**Conjunto de características**	**Melhores percentuais de taxa de acerto do classificador**
4C	C4.C5.C6.C9	86,66%
5C	C3.C4.C5.C6.C9	87,33%
6C	C1.C4.C5.C6.C9.C11	92,22%
7C	C1.C4.C5.C6.C8.C9.C11	94,44%
8C	C1.C3.C4.C5.C6.C8.C9.C11	94,88%
9C	C1.C2.C3.C4.C5.C6.C8.C9.C11	94,22%
10C	C1.C3.C4.C5.C6.C7.C8.C9.C10.C11	94,22%
11C	C1.C2.C3.C4.C5.C6.C7.C8.C9.C10.C11	93,77%.

4.6. Avaliação dos Modelos do Adversário

Nesta seção são apresentados os *resultados do tes*te de avaliação dos *modelos do adversário* utilizados para comprovar a importância e justificar o motivo de tais características serem classificadas como tal. Para cada conjunto de características usado na avaliação do *candidato à modelo do adversário*, foi selecionado o resultado com o maior percentual de acerto no classificador. Em seguida, foram retirados conjuntos (combinações, de 1C, 2C, 3C etc) de características do *modelo do adversário* 4C (C4, C5, C5 e C9) da base de mensagens e refeito o processo de classificação. O objetivo foi medir o melhor percentual de acerto alcançado sem tais conjuntos e, portanto identificar na prática se o resultado da análise que nos levou ao *modelo do adversário* fazia sentido. Ou seja, se retirando certa característica da base o classificador notadamente tivesse dificuldade para fazer a classificação, aquela(s) característica(s) seria(m) fundamental(is).

Foi utilizado o *modelo do adversário* de 4C porque o de 6C geraria tabelas muito extensas e o 2C seria muito limitado, logo o valor intermediário é o 4C. Além disto, a área sob a curva (AUC) tem valores próximos entre 2C e 6C, assim como o comportamento das curvas ROC – preservadas as devidas proporções no que diz respeito ao número de características.

Houve casos em que o classificador não conseguiu ser preciso na classificação quando algumas dessas características foram retiradas – identificamos estes casos com a frase "classificação com resultado aleatório" (RR), na tabela 4.17. Em outros casos o percentual de acerto do classificador caiu drasticamente.

Para os conjuntos de 3C o melhor percentual sem a retirada de características foi de 77,77%. Entretanto, nota-se que quando se retira os conjuntos {C4, C5}, {C4, C5, C6}, {C4, C5, C9} e {C4, C5, C6, C9} o classificador não é preciso (Tabela 4.17). Ou seja, no caso da retirada do conjunto {C4, C5, C6, C9}, por exemplo, o classificador só poderia utilizar conjuntos de 3C formados por CMA = {C1, C2, C3, C7, C8, C10, C11}. Porém, nenhum conjunto de 3C contendo apenas as características do conjunto CMA (Candidato à Modelo do Adversário) foi capaz de fornecer informação suficiente ao classificador para que ele pudesse ser preciso na classificação. A tabela 4.17 apresenta o impacto causado na taxa de acerto dos classificadores (2C a 11C) com a remoção das características do Modelo do Adversário 4C.

Para o conjunto 4C = {C4, C5, C6, C9} o melhor percentual foi de 86,66%. Com a retirada de qualquer característica do *modelo do adversário* 4C observa-se que o percentual de acerto do classificador sofreu uma queda de quase 10%.

Mesmo aumentando o número do conjunto para 5C, ainda há casos em que a retirada de características do *modelo do adversário* 4C prejudica totalmente a classificação. Por exemplo, com a retirada do conjunto {C4, C5} o classificador não consegue ser preciso e gerar resultados aceitáveis. Embora neste caso a melhor combinação não contenha as quatro características do *modelo do adversário* 4C, e aparentemente o melhor resultado não dependa tanto delas, nota-se a fundamental importância que as características C5 e C6 agregaram ao percentual de acerto, mesmo isoladas, visto que sem essas a taxa de acerto cai mais de 6% e 8%, respectivamente.

Tabela 4.17 - Avaliação da influência do *modelo do adversário*

Características removidas	**Número de características combinadas e melhor taxa de acerto (%)**									
	2C	**3C**	**4C**	**5C**	**6C**	**7C**	**8C**	**9c**	**10C**	**11C**
{C4}	RR	77,78	77,78	90,66	77,78	77,78	77,78	77,78	77,78	-
{C5}	RR	77,78	79,11	84,00	88,66	88,88	88,88	89,77	88,88	-
{C6}	RR	77,78	77,78	82,22	87,55	87,55	92,44	92,66	77,78	-
{C9}	RR	77,78	77,78	90,66	82,00	82,00	82,00	78,00	77,78	-
{C4, C5}	RR	RR	RR	RR	RR	RR	82,00	RR	-	-
{C4, C6}	RR	77,78	77,78	77,78	77,78	77,78	77,78	77,78	-	-
{C4, C9}	RR	77,78	77,78	90,66	77,78	77,78	77,78	77,78	-	-
{C5, C6}	RR	70,88	70,44	82,22	87,55	71,77	87,33	87,33	-	-
{C5, C9}	RR	70,88	70,44	82,00	82,00	82,00	82,00	67,33	-	-
{C6, C9}	RR	77,78	77,78	77,78	77,78	77,78	77,78	78,00	-	-
{C4, C5, C6}	RR	RR	RR	RR	RR	RR	RR	-	-	-
{C4, C5, C9}	RR	RR	RR	RR	RR	RR	RR	-	-	-
{C4, C6, C9}	RR	77,78	77,78	77,78	77,78	77,78	77,78	-	-	-
{C5, C6, C9}	RR	70,88	70,44	RR	66,44	64,22	63,77	-	-	-
{C4, C5, C6, C9}	RR	RR	RR	RR	RR	RR	-	-	-	-
Melhor Percentual de acerto com 4C	**77,78**	**77,78**	**86,66**	**90,66**	**92,22**	**94,44**	**94,88**	**94,22**	**94,22**	**93,77**

Combinando seis características, o melhor resultado cai quase 4% apenas com a retirada de C5. A maior queda de percentual com a retirada de uma única característica do *modelo do adversário* 4C é quando C4 está ausente, pois a taxa de acerto caiu quase 15%.

Com sete características, o melhor percentual de acerto do classificador sem o *modelo do adversário* 4C foi 88,88%, o que representa uma queda de mais de 5% em relação ao melhor resultado. O pior caso que ocorreu com a retirada de uma única característica foi quando C4 esteve ausente, resultando em uma queda de quase 17% na taxa de acerto.

Com oito características, a mesma dependência de C4 se repete, com o percentual de acerto caindo mais de 17% com a sua retirada (Tabela 4.17).

Com os conjuntos de nove características, a dependência de C4 é comprovada mais uma vez, representando uma queda de mais de 16% no percentual (Tabela 4.17). O melhor

resultado sem o *modelo do adversário* 4C ocorreu com a retirada de C6, pois a taxa de acerto ficou em 92,66%.

Com 10 características as quedas de percentual são praticamente as mesmas que 9C, em torno de 16% no pior caso (Tabela 4.17).

A avaliação do *modelo do adversário* 4C serviu para comprovar a real importância das características identificadas como *candidato à modelo do adversário*. Nos conjuntos 4C, 5C, 6C, 7C e 9C, o classificador não foi preciso em sua tarefa sempre que o subconjunto {C4, C5} esteve ausente. Com a avaliação considerando 8C, não foi possível uma classificação precisa com a ausência dos conjuntos {C4, C5, C6} ou {C4, C5, C9}.

O único caso em que o classificador não teve problemas para executar sua tarefa foi em 10C, porém neste caso apenas os conjuntos individuais foram retirados, logo permaneciam na base de classificação no mínimo 3C do conjunto do *candidato à modelo do adversário* 4C. Mesmo assim, o percentual da taxa de acerto teve uma queda de 16% no pior caso, em relação a melhor taxa de acerto do resultado sem a retirada de nenhuma característica.

Além da avaliação acima, foi executado um teste em uma ferramenta real para detecção de SPAM/*phishing*, o *Spam Assassin* [The Apache SpamAssassin Project, 2010]. O objetivo deste teste foi avaliar se o modelo do adversário se confirmava na prática e como consequência a *engine* de detecção se simplificava. O teste com o *Spam Assassin* foi realizado em duas etapas, que serão detalhadas a seguir.

Na primeira etapa, o *Spam Assassin* foi treinado com a mesma base de treinamento que foi utilizada para obter os classificadores. Em seguida, foram feitos vários ajustes no *threshold* da ferramenta para identificar aquele que atingiria a melhor taxa de acerto. Com o *threshold* definido em 2.9, foi possível classificar corretamente 96,44% dos *phishing* de email.

Na segunda etapa, a otimização do *Spam Assassin* foi mantida (*threshold* 2.9), realizando um treinamento apenas com as mensagens de *phishing* com as características do *candidato à modelo do adversário*, {C4, C5, C6 e C9}. Após a classificação, a taxa de acerto na detecção de *phishing* de email foi de 89,78%.

É importante levar em consideração que, na segunda etapa, o treinamento foi feito só com as mensagens que continham as características do *candidato à modelo do adversário* 4C, o que corresponde a apenas 33,77% da base de treinamento da primeira etapa. Porém, mesmo

com apenas 36% das características (4 das 11), foi obtida uma taxa de acerto que é apenas 6,68% menor que a da primeira etapa. Isto mostra que com as mensagens que compõem o modelo do adversário 4C, o Spam Assassin capturou as propriedades/perfil de *phishing*. Como em geral para cada característica existe uma regra de detecção, este resultado nos permitiu inferir que a *engine* de detecção teve uma redução significativa, porque o conjunto de característica reduziu-se de 11 para 4, o que significa uma redução de 275%.

Os resultados relatados nesta seção mostram com clareza que o *modelo do adversário* é composto das características que formam essencialmente o cerne do conjunto de características que permitem a identificação de *phishing de email*. Nota-se que a ausência das características C4 e C5 influenciam de maneira importante a classificação. Esta informação é bastante importante, pois pode-se facilmente usar uma ferramenta que gere estatísticas das características que estão sendo extraídas do fluxo de email de entrada no gateway SMTP. De posse desta informação em uso real da proposta para detectar *phishing* de email o administrador do servidor terá um indício de que o *modelo do adversário* precisará ser refeito quando características essenciais com C4 e C5 deixarem de aparecer entre as mensagens que passam pelo detector no gateway SMTP.

4.7. Aspectos de implementação do protótipo

O módulo de extração das características e avaliação de desempenho foi desenvolvido em *shell script*, sendo composto por uma função principal que faz a leitura das bases de *phishing* e não-*phishing*. Em cada iteração, uma nova mensagem da base é submetida à extração de 11 características − para cada uma há subfunções de extração específica. Ao final de cada iteração, um arquivo contendo os vetores de característica é preenchido com as informações obtidas durante a extração. Adicionalmente, é gerado um arquivo de *log* com as saídas do interpretador de comando e do *script* de extração. Sendo assim, o resultado da extração para auxiliar na auditoria do sistema. Para os testes de força bruta foi desenvolvido um módulo que gera todas as combinações possíveis e, para cada uma delas, são buscadas as características correspondentes no arquivo que contem todos os vetores extraídos. Um novo arquivo de vetores é criado, eliminando possíveis vetores em duplicidade, e é armazenado em um local específico para posterior análise do classificador.

Finalmente, uma função cria uma iteração para cada arquivo (conjunto) de combinação existente, realizando as etapas de treinamento e classificação de forma automatizada utilizando ferramentas disponíveis pela ferramenta LIBSVM. Os resultados do classificador e as saídas do interpretador são organizados e registrados em vários arquivos para que possam ser avaliados a posteriori.

A escolha do *shell script* como linguagem utilizada neste trabalho teve os seguintes motivos:

- A extração de texto é uma atividade que nem sempre pode ser feita de modo simples com algumas linguagens de programação. Como a extração de características depende basicamente de uma extração textual, somando-se à necessidade constante de alterações para adaptação e criação de novas características, seria necessário o uso de uma linguagem rica em ferramentas/recursos para extração de texto e que pudesse ser facilmente compreendida e alterada quando necessário.
- O *shell script* é uma linguagem rica em ferramentas para filtros textuais, contendo dezenas dos mesmos e possibilita a combinação entre eles. Adicionalmente, é possível também utiliza-las juntamente com expressões regulares, o que torna os filtros mais robustos e refinados.
- Ao contrário do que muitos pensam, o *shell script* não é apenas uma linguagem criada somente para a implementação *scripts*, podendo ser utilizada para a criação de verdadeiros programas de forma bem organizada [Jargas 2008].
- Considerando as vantagens citadas anteriormente, somando-as ao conhecimento prévio desta linguagem pela autoria deste trabalho, o *shell script* possuía tudo o que era necessário para a implementação dos módulos necessários para este estudo.

A implementação do protótipo poderia ser feita em qualquer outra linguagem, só foi feita em o *shell script* porque esse é bastante conhecido e facilmente entendido por administradores de sistemas, possíveis usuários reais da proposta. Para a geração das curvas ROC e AUC, foi utilizado o software MatLab.

4.8. Avaliação de Desempenho

A avaliação de desempenho foi uma atividade difícil, visto que o tempo para extrair as características pode variar muito. Por exemplo, há características que são diretamente obtidas (em média gastando 0,01 segundo por email) e outras que necessitam de várias iterações para ser completadas (em média gastando 0,1 segundo por email). Evidentemente, não foi realizado nenhum tipo de otimização do *script shell* desenvolvido para obter as características.

Entretanto, avaliando somente as melhores combinações de cada conjunto do *candidato à modelo do adversário*, é possível ter uma idéia do tempo necessário para extrair um determinado número de emails e avaliar as diferenças do custo computacional necessário de acordo com o número de características. A tabela 4.18 contem os tempos normalizados (em percentuais) para a extração das características, considerando que o tempo para extração de todas elas é de 100%.

Tabela 4.18 – Tempo para extração dos conjuntos de características de 4C a 11C

Número de características	Combinação	Tempo para extração de 100 emails (%)
4C	C4.C5.C6.C9	43,22%
5C	C3.C5.C6.C10.C11	46,97%
6C	C1.C4.C5.C6.C9.C11	78,70%
7C	C1.C4.C5.C6.C8.C9.C11	84,85%
8C	C1.C3.C4.C5.C6.C8.C9.C11	88,77%
9C	C1.C2.C3.C4.C5.C6.C8.C9.C11	90,82%
10C	C1.C3.C4.C5.C6.C7.C8.C9.C10.C11	97,95%
11C	C1.C2.C3.C4.C5.C6.C7.C8.C9.C10.C11	100,00%

Uma das principais conclusões e obtidas com o teste de desempenho é que o tempo de extração da melhor combinação de 6C é ~32% maior em relação à 5C, para um ganho de apenas ~5% na taxa de acerto do classificador. Para atingir o maior percentual, ou seja, utilizando a combinação de 8 características, o aumento no tempo de processamento é de ~10% em relação à 6C para um ganho percentual de apenas 2,22% na taxa de acerto do

classificador. Ainda para 8C, haveria um aumento de ~42% no tempo de processamento em relação à 5C para um ganho de 7,55% na taxa de acerto do classificador.

Nas etapas de treinamento e teste, considerando o classificador, observou-se que o aumento de consumo do tempo de processamento de 11 características em relação a 4 (*modelo do adversário*) é de aproximadamente 10%. Ou seja, o aumento maior no custo de processamento ocorre durante a extração das características que, em um ambiente de correio eletrônico em produção, seria uma atividade realizada para toda mensagem recebida pelo MTA, podendo se tornar o gargalo do sistema de email.

Com o uso de combinações menores de características e que atingem um percentual razoável de acerto, poderia ser utilizado um método de cascateamento de classificadores, no qual a maior parte das mensagens seria classificada já na primeira filtragem (computacionalmente menos custosa) e, em seguida, as demais mensagens passariam por outros ciclos de filtragem, desta vez mais minuciosa, e assim por diante.

Capítulo 5

Conclusão

Apesar do *phishing* de email ser um problema já explorado em vários trabalhos, a maioria deles se limita a um determinado aspecto do problema. Nas abordagens de detecção baseadas em características do email, quando não é utilizada uma quantidade excessiva de características, o uso de muitas delas não é naturalmente entendido ou explicado do ponto de vista de caracterização distinta de *phishing*.

Outro aspecto negativo encontrado nas abordagens da literatura técnica que foram apresentadas até então, diz respeito ao número excessivo de características utilizadas na detecção ou que dependem de consultas a serviços externos, o que representa um aumento considerável no tempo de resposta para cada email avaliado em um sistema de detecção real. Além disso, as abordagens encontradas na literatura técnica não oferecem uma alternativa para a mudança do perfil de *phishing*.

Neste estudo, foi apresentada uma abordagem que sistematiza o processo de identificação das características essenciais (*modelo do adversário*) de *phishing*. Com isso evita-se o uso excessivo de características na detecção. O *modelo do adversário* oferece muitas vantagem que auxiliam na detecção de *phishing*, evitando desperdício de tempo e processamento (comprovado em nossos testes de desempenho), sem perder a eficácia da

detecção. Observou-se também que estas características essenciais estiveram presentes em quase todos os melhores conjuntos encontrados na avaliação *candidato à modelo do adversário*.

A fim de avaliar a real importância das características que compõem o *modelo do adversário* de 4C, realizamos testes que comprovaram que na maioria dos casos a ausência das mesmas compromete significativamente o resultado do classificador.

O uso das ROCs e AUCs também auxiliou bastante na busca pelos conjuntos de classificadores mais eficientes e confiáveis. Assim, não nos limitamos apenas à taxa de acerto do classificador, que não revela por si só outros aspectos fundamentais como as taxas de falsos positivos e verdadeiros positivos (encontrados através das curvas ROCs). O uso das curvas ROCs também foi importante para mostrar que esta é uma técnica muito útil para escolher classificadores de acordo com os requisitos do sistema (FP x TP). Nos casos em que as curvas ROCs foram muito parecidas, utilizamos as AUCs como critério de escolha.

Nossa proposta de *modelo do adversário* considera a possibilidade de mudança das características essenciais, ou seja, não será necessário refazer todo o estudo relatado neste trabalho, basta apenas rodar um teste com as combinações que fornecem a melhor taxa de acerto e confiabilidade. Por outro lado, se uma das características do *modelo do adversário* diminuir sua incidência de ocorrência, o mesmo pode perder a eficácia, assim a reavaliação periódica do *modelo do adversário* e possível reconfiguração da combinação de característica do mesmo é muito importante.

Como o *phishing* pode se apresentar de diferentes maneiras em diferentes épocas, o administrador poderia escolher entre as combinações que proveem o melhor resultado em cada época. Esta tarefa de avaliar periodicamente o melhor resultado e reconfigurar as características com melhores resultados pode ser facilmente automatizada.

Nesta proposta mostrou-se que o *modelo do adversário* (4 características) está presente em quase todas as melhores combinações para cada número de características. Também concluiu-se que em alguns casos, dependendo das características utilizadas, o aumento no tempo de processamento não justifica o percentual de ganho na taxa de acerto,

pois este é muito pequeno. Além disto, a avaliação do *modelo do adversário* permitiu perceber que se as características de suporte do modelo deixarem de existir a detecção pode ser tornar ineficiente. Neste caso, o administrador tem uma ferramenta importante para identificar quando a detecção vai falhar. Em qualquer situação a metodologia aqui apresentada precisará ser refeita. Ou seja, nenhuma outra abordagem oferece flexibilidade com relação a reconfiguração e nem uma métrica que evidencia o momento que o modelo vai falhar e precisa ser refeito, como a desenvolvida nesta proposta.

A obtenção do *modelo do adversário* permitiu um ganho de 17% na performance e permitiu diminuir a quase a metade o número de características de *phishing* a serem extraídas sem perdas significativas em termos de eficiência de detecção.

Este trabalho pretendeu mostrar que se está muito próximo da obtenção de uma metodologia para avaliação de *phishing*. Porém, tem-se consciência que os percentuais de acerto e confiabilidade podem ser melhorados. Pretende-se continuar trabalhando no aprimoramento das bases no sentido de gerar uma base de consulta online para treinamento de detectores de *phishing* (temos esta motivação porque não encontramos tal base para desenvolver o nosso trabalho).

Adicionalmente, pretende-se trabalhar com as características C17 a C19, numa abordagem distribuída e cooperativa visando disseminar alertas etc., que podem ser compartilhados no intuito de evitar o tempo de espera gerado por consultas externas na detecção de *phishing*.

Referências

ANTI-PHISHING Working Group (2007) Phishing Activity Trends: Report for the month of September. Disponível em: www.antiphishing.org/reports/apwg_report_sept_2007.pdf.

ANTI-PHISHING Working Group (2008) Phishing Activity Trends: 2nd Half 2008, Disponível em: http://www.antiphishing.org/reports/apwg_report_H2_2008.pdf, Maio.

BASNET, R., MUKKAMALA, S. E SUNG, A. (2008) Detection of Phishing Attacks: A Machine Learning Approach, Em: Soft Computing Applications in Industry, p. 373-383.

CASTILLO, M.; IGLESIAS, A; SERRANO, J. (2007) Detecting Phishing E-mails by Heterogeneous Classification. Em: INTELLIGENT DATA ENGINEERING AND AUTOMATED LEARNING (IDEAL), 8, Birmingham, UK. Springer, p. 296-305.

CHANDRASEKARAN, M.; CHINCHANI, R.; UPADHYAYA, S. (2006) PHONEY: Mimicking User Response to Detect Phishing Attacks. Em: WORLD OF WIRELESS, MOBILE AND MULTIMEDIA NETWORKS.

CHAVES, A. (2006) Extração de Regras Fuzzy para Máquinas de Vetor de Suporte (SVM) para Classificação em Múltiplas Classes. Em: PUC-Rio.

CHEN, J. E GUO, C. (2006) Online Detection and Prevention of Phishing Attacks, Em: Communications and Networking in China, p.19-21.

CHOU, N.; LEDESMA, R.; TERAGUSHI, Y.; MITCHELL J. (2004) Client-side Defense Against Web-based Identity Theft. Em: NETWORK AND DISTRIBUTED SYSTEM SECURITY SYMPOSIUM, 11, San Diego, EUA. p. 1-16.

Código Penal Brasileiro, Título II, Cap. VI, Art. 171, Disponível em: <http://www.planalto.gov.br/ccivil_03/Decreto-Lei/Del2848.htm>. Acesso em 18 fev. 2010.

COOK, D., GURBANI, V. E DANILUK, M. (2008) Phishwish: A Stateless Phishing Filter Using Minimal Rules, Em: Lecture Notes in Computer Science, p.182-186.

CRANON, L.; EGELMAN, S.; HONG, J.; ZHANG, Y. (2007) Phinding Phish: An Evaluation of Anti-Phishing Toolbars. Em: NETWORK AND DISTRIBUTED SYSTEM SECURITY SYMPOSIUM, 17, San Diego, EUA. p. 1-16.

EARTHLINK, INC. Earthlink toolbar. Disponível em: <http://earthlink.net/earthlinktoolbar>. Acesso em: 15 mar. 2010.

eBay, Inc. - eBay Toolbar with Account Guard. Disponível em: <http://pages.ebay.co.uk/toolbar/accountguard_1.html>. Acesso em: 13 mar. 2010.

FETTE, I., SADEH, N. E TOMASIC, A. (2007) Learning to Detect Phishing emails, Em: International World Wide Web Conference, p.649-656.

HERZBERG, A.; GBARA, A. TrustBar: Protecting (even Naïve) Web Users from Spoofing and Phishing Attacks. Disponível em: <www.cs.biu.ac.il/~herzbea/TrustBar/>. Acesso em: 12 mar. 2010.

JARGAS, A. (2008). Shell Script Profissional. São Paulo: Editora Novatec.

KIRDA, E.; KRUEGEL, C. (2005) Protecting Users Against Phishing Attacks with Antiphish. Em: COMPUTER SOFTWARE AND APPLICATIONS CONFERENCE, 29, Edinburgh, Escócia, p. 517-524.

M. CHANDRASEKARAN, K. NARAYANAN, S. UPADHYAYA (2006) Phishing email Detection Based on Structural Properties, Em: Cyber Security Symposium.

M. KUDO, J. SKLANSKY (2000) Comparison of algorithms that select features for pattern classifiers, Em: Pattern Recognition, p.25-41.

MESSAGELABS (2008) MessageLabs Intelligence: 2008 Annual Security Report, Relatório técnico. Disponível em: http://www.messagelabs.com/resources/mlireports.

NETCRAFT Toolbar. Disponível em: <http://toolbar.netcraft.com/>. Acesso em: 12 mar. 2010.

RADICATI Group, Inc. “Email Statistics Report, 2009-2013”. Relatório técnico. Disponível em: http://www.radicati.com/wp/wp-content/uploads/2009/05/e-mail-statistics-report-2009-pr.pdf, 2009.

RESENDE, S. (2003) Sistemas Inteligentes: Fundamentos e Aplicações. São Paulo: Editora Manolé.

SPITZNER, L. Honeytokens: The other honeypot. Disponível em: <http://www.symantec.com/connect/pt-br/articles/honeytokens-other-honeypot>. Acesso em: 22 mar. 2010.

T. FAWCETT (2006) An Introduction to ROC analysis, Em: Pattern Recognition Letters, p.861-874.

The Apache SpamAssassin Project, Disponível em: <http://spamassassin.apache.org/>. Acesso em: 10 jan. 2010.

V. VAPNIK (1995) The nature of statistical learning theory, Em: Springer Verlag.

WU, M.; MILLER, R.; LITTLE, G. (2006) Web Wallet: Preventing Phishing Attacks by Revealing User Intentions. Em: SYMPOSIUM ON USABLE PRIVACY AND SECURITY, 2, Pittsburgh, EUA. p. 102.113.

WU, M.; MILLER, ROBERT.; GARFINKEL, S. (2006) Do Security Toolbars Actually Prevent Phishing Attacks? Em: CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS. Montreal, Canadá. ACM, p. 601-610.

Y. PAN, X. DING (2006) Anomaly Based Web Phishing Page Detection, Em: Annual Computer Security Applications Conference.

Y-G. KIM, M-S. JANG, K-S. CHO E G-T. PARK (2008) Performance comparison between backpropagation, neuro-fuzzy network, and SVM, Em: Lecture Notes in Computer Science, p.438-446.