SPAMBANDS: UMA METODOLOGIA PARA IDENTIFICAÇÃO DE INFRAESTRUTURAS DE *SPAM* AGINDO DE FORMA ORQUESTRADA

ELVERTON CARVALHO FAZZION

SPAMBANDS: UMA METODOLOGIA PARA IDENTIFICAÇÃO DE INFRAESTRUTURAS DE *SPAM* AGINDO DE FORMA ORQUESTRADA

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Orientador: Dorgival Olavo Guedes
Coorientador: Ítalo Cunha

Belo Horizonte
Setembro de 2016

Ficha catalográfica elaborada pela Biblioteca do ICEx - UFMG

Fazzion, Elverton Carvalho.

F287s SpamBands: uma metodologia para identificação de infraestruturas de spam agindo de forma orquestrada. / Elverton Carvalho Fazzion. – Belo Horizonte, 2016.
xviii, 83 f.: il.; 29 cm.

Dissertação (mestrado) - Universidade Federal de Minas Gerais – Departamento de Ciência da Computação.

Orientador: Dorgival Olavo Guedes Neto.
Coorientador: Ítalo Fernando Scotá Cunha .

1. Computação - Teses. 2. Spam (Mensagens eletrônicas) 3. Segurança da informação. I. Orientador. II. Coorientador. III.Título.

CDU 519.6*74(043)

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

SpamBands: uma metodologia para identificação de infraestruturas de spam agindo de forma orquestrada

ELVERTON CARVALHO FAZZION

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROF. DORGIVAL OLAVO GUEDES NETO - Orientador
Departamento de Ciência da Computação - UFMG

PROF. ÍTALO FERNANDO SCOTÁ CUNHA - Coorientador
Departamento de Ciência da Computação - UFMG

DRA. CRISTINE HOEPERS
CGI.br.

DR. KLAUS STEDING -Jessen
CGI.br

PROF. WAGNER MEIRA JÚNIOR
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 18 de agosto de 2016.

Agradecimentos

Mais uma etapa da minha vida profissional é concluída e esse fato só se tornou possível devido à colaboração de diversas pessoas nesse processo e que merecem a devida menção nesta seção.

Inicialmente, gostaria de agradecer ao meu pai, Elcio, por sempre me apoiar em minhas decisões e sempre me incentivar, nos momentos mais difíceis, a continuar estudando. As viagens a trabalho que o senhor me levou, desde criança, indo de supermercado a supermercado oferecer produtos me ensinaram, com grande valia, a dar valor a cada conquista que eu tenho realizado e a me tornar a pessoa que sou hoje. Agradeço também a minha mãe, Aparecida, que me faz companhia nos momentos solitários da vida em Belo Horizonte. Agradeço também a minha prima, Carolina, pela companhia na infância e que me ajudava a zerar mario de supernintendo todos os finais de semana.

Agradeço à minha namorada, Luciana, pelo carinho e apoio oferecidos durante as fases mais díficeis desses últimos anos. Você é uma pessoa maravilhosa que eu amo cada dia mais. Obrigado por ser uma super namorada!

Gostaria de agradecer individualmente cada um dos três professores envolvidos no processo da minha dissertação. Agradeço ao professor Dorgival Guedes, meu orientador, pelos importantes direcionamentos feitos no processo de orientação os quais foram fundamentais o término desse trabalho. Agradeço ao professor Ítalo Cunha, meu coorientador, por fuzilar, diversas vezes, as ideias presentes nessa dissertação e me fornecer dicas importantes tanto para a vida pessoal quanto para a vida acadêmica. Agradeço ao professor Wagner Meira Jr., meu orientador na graduação, e a quem considero meu segundo coorientador (e também um amigo) nesse trabalho e que sempre me forneceu importantes informações em momentos de desespero.

Agradeço também aos meus amigos do laboratório SPEED por proporcionarem um ótimo ambiente de trabalho. Em particular Rubens, Paulo, Júlio, Camila, os dois Walters, Denise, Derick, Samuel, Rafael e Fernandinho que são amigos que irei levar

para a vida inteira. Também gostaria de agradecer a dois grandes amigos, Osvaldo e Vinícius, pelos diversos trabalhos que realizamos juntos seja em disciplinas (com o Osvaldo) quanto na parte operacional do laboratório (com o Vinícius). Claro que não poderia faltar a menção ao grupo de RPG do SPEED composto por Rubens (Photon, o clérigo leal bom mercenário), Camila (Cammy, a conjuradora arcana vaidosa), Vinícius (Ragnar, o ladino matador), Osvaldo (Morpheus, o zoológico ambulante), Ítalo (Valas, o arqueiro roubado), eu (o temido Krusk Feng) e Júlio (o mestre piedoso).

Agradeço também aos amigos de São João del Rei que foram bons companheiros durante minha infância e adolescência e ainda o são, principalmente Pablo e Diguinho pelos momentos de descontração proporcionado por conversas sobre diversos temas. Final do ano tem o RPG de SJDR o/!

Agradeço aos colaboradores do CERT.br, Klaus, Cristine e Marcelo, pelo suporte, sugestões e comentários que foram fundamentais para a realização deste trabalho.

Por fim, gostaria de agradecer aos álbuns do Foo Fighters e Queens of the Stone Age, as músicas temas de jogos de supernintendo (Mortal Kombat, Donkey Kong e Mega Man) e trilhas de filmes (Rocky, Exterminador do Futuro, Matrix e a abertura da CIC Video dos anos 90) que foram parte do combustível deste trabalho.

GG! :)

Resumo

Outrora a batalha contra os *spammers* era devido ao grande tráfego na rede ocasionado pelo alto volume de mensagens de *spam* enviadas. Atualmente, a batalha é travada devido ao conteúdo dessas mensagens enviadas por quem pratica esse abuso. Em geral, mensagens de *spam* possuem dois objetivos: realizar propaganda de produtos e serviços ilegais ou obter informações sigilosas do destinatário. Essas duas práticas levam a prejuízos sociais e financeiros aos usuários e, por isso, é necessária a criação de mecanismos para mitigar o problema. O histórico das diversas técnicas de combate a este abuso propostas na literatura mostram o caráter evolutivo do *spammer*, que também evolui os mecanismos para enviar suas mensagens. Neste trabalho propomos os *SpamBands*, uma técnica que combina informações de conteúdo e rede de mensagens de *spam* para identificar a infraestrutura utilizada pelo *spammer* como servidores e computadores infectados com *malware*. Aplicamos a técnica sobre mensagens coletadas em quatorze *honeypots* de baixa interatividade instalados ao redor do mundo que simulam serviços de *proxy* e *relay* abertos. Diante dos *SpamBands*, realizamos importantes observações: mostramos que o conteúdo de phishing está mais relacionado a grupos que exploram o *honeypot* como *relay* aberto, indicativo de *botnets*, e estão ligados a idiomas ocidentais enquanto propagandas ilegais podem estar sendo enviadas tanto por *botnets* quanto servidores dedicados e estão mais ligadas a idiomas orientais. Nós também apresentamos um modelo que permite identificar grupos colaborativos de campanhas entre endereços IP nos *SpamBands* ao longo do tempo. Nossas observações mostram que muitos desses grupos ficam ativos poucos dias com intervalo significativo entre suas atividades nos *honeypots*. Esses resultados levam a crer que são necessárias diferentes formas de combate ao abuso e motivam o desenvolvimento de novas técnicas.

Abstract

Once, the battle against spammers was due to the heavy network traffic caused by the high volume of spam messages sent. Today, the battle is against the content sent by those who practice this abuse. Generally, spam messages have two goals: make advertisements of illegal products and services, or retrieve confidential information from the recipient. These two practices lead to social and financial losses in the order of billions of dollars per year and, therefore, new mechanisms are needed to mitigate the problem. The history and diversity of anti-spam techniques proposed in the literature show the evolutionary behavior of spammers who also improve their techniques to send spam. In this work we propose SpamBands, a technique that combines content and network information from spam messages to identify the infrastructure used by the spammer, such as servers and computers infected with malware. We apply the technique on messages collected by fourteen low-interactivity honeypots around the world that simulate open proxy and relay services. The SpamBands detected allow us to make important observations: we show that phishing content is sent by groups that exploit the honeypot as an open relay, indicative of botnet machines, and are connected to western languages while illegal advertisements may be being sent by botnets and dedicated servers and are linked to oriental languages. We also present a model that identifies collaborative groups of campaigns among IP addresses in SpamBands over time. Our observations show that many of these groups stay active for only few days with a long periods of inactivity on the honeypots. These results suggest and motivate that different techniques are needed to combat this abuse.

Lista de Figuras

2.1 Linha do tempo das principais técnicas de combate ao *spam*. 5

3.1 Arquitetura de coleta com *honeypots*. 16
3.2 Distribuição dos 10 idiomas mais frequentes nas mensagens por *honeypot* e global 20
3.3 Ilustração da FP-Tree. 24
3.4 Ilustração do algoritmo de geração de campanhas aplicado em uma FP-Tree. 25
3.5 Treinamento do classificador bayesiano 27
3.6 Seleção de atributos utilizado para o algoritmo Naive Bayes 27
3.7 Ilustração do cálculo das probabilidades do modelo bayesiano 28
3.8 Exemplo de aplicação do classificador a um novo documento 28

4.1 Geração do grafo de relações entre endereços IP. 32
4.2 Grafo de relações entre IP sem utilizar o algoritmo de detecção de *SpamBands* 34
4.3 Detecção dos *SpamBands* utilizando o algoritmo de detecção de *SpamBands* 34
4.4 Distribuição dos *SpamBands* por dia e por *honeypot* 36
4.5 Visão geral dos *SpamBands* em relação aos protocolos. 37
4.6 Exemplo de mensagens de campanhas em russo e alemão. 39
4.7 Relação entre *SpamBands* e idiomas 41
4.8 Exemplo do anexo de uma mensagem sem texto no corpo da mensagem. . 42
4.9 Mensagens em italiano e francês. 44
4.10 Mensagens em japonês e russo. 44
4.11 Mensagem em chinês com termos aleatórios escondidos para confundir filtros. 45
4.12 Distribuição dos *SpamBands* por especialidade: propaganda ou *phishing* . 47
4.13 Distribuição dos *SpamBands* por classe (propaganda e *phishing*) nos protocolos em relação ao número de mensagens e número de campanhas 49

5.1 Ilustração dos eventos ocorridos em comunidades. 52
5.2 Detecção de redes de colaboração usando o índice de influência. 53

5.3 Visão geral do tamanho e número de dias ativos de cada grupo. 54
5.4 Estabilidade dos grupos colaborativos encontrados. 56
5.5 Exemplo de mensagem do PayPal enviada pelo grupo de colaboração detectado. 56
5.6 Exemplo de mensagem em holandês para pagamento de títulos. 57
5.7 Exemplo de mensagens de *phishing* com temas de serviços de email e do banco CAIXA. 59
5.8 Exemplos de mensagens de *phishing* com temas de ofertas de trabalho e dos Correios. 59
5.9 Exemplo de campanha de *phishing* com tema das lojas Americanas 60

A.1 Exemplo de mensagens de campanhas em chinês enviadas por um *SpamBand* HTTP. 73
A.2 Exemplo de mensagens de campanhas enviadas com intuito de furtar informações de contas da Apple. 75
A.3 Exemplo de mensagens de campanhas de *phishing* para bancos canadenses e mexicanos. 76
A.4 Exemplo de mensagens de campanhas de *phishing* para bancos espanhóis e italianos. 77
A.5 Exemplo de mensagens de campanhas de *phishing* para bancos brasileiros e americanos. 78

B.1 Esquema do sistema desenvolvido para o CERT.br. 80

Lista de Tabelas

4.1 Atributos dos *SpamBands* da figura 4.3. 36
4.2 Atributos do *SpamBand* com propagandas russas e alemãs. 39
4.3 Atributos do *SpamBand* com mensagens com conteúdo de texto vazio. . . . 42
4.4 Atributos do *SpamBand* com cinco idiomas. 43
4.5 Relação ccTLD e idioma nos *SpamBands*. 46
4.6 Distribuição dos idiomas por protocolo. 46
4.7 Matriz de relação dos *SpamBands* entre as especialidades do *spammer* e protocolo. 48
4.8 Relação entre idioma e especialização do *SpamBand* 49

5.1 Atributos do grupo colaborativo para envio de campanhas para furto de contas do Paypal. 57
5.2 Atributos do grupo colaborativo para envio de *phishing* holandês. 58

A.1 Atributos do *SpamBand* de venda de produtos farmacêuticos. 74
A.2 Atributos do *SpamBand* de furto de contas da Apple. 75

Sumário

Agradecimentos **vii**

Resumo **ix**

Abstract **xi**

Lista de Figuras **xiii**

Lista de Tabelas **xv**

1 Introdução **1**

2 Spam e Técnicas de Combate **5**

2.1 Identificação e filtragem de spam no destino 7

2.2 Identificação e combate à origem do spam 8

2.3 Campanhas de spam 11

2.4 Combate ao phishing 12

2.5 Determinando a infraestrutura do spammer 13

3 Detecção e análise das campanhas de spam **15**

3.1 Coleta e avaliação dos dados 16

3.2 Estrutura de dados FP-Tree 17

3.3 Detecção de idioma 19

3.4 Algoritmo para geração de campanhas 21

3.5 Exemplo do algoritmo de campanhas aplicado na FP-Tree 24

3.6 Identificação de phishing em campanhas 25

4 SpamBands **31**

4.1 Identificação dos SpamBands 31

4.2 Avaliação dos SpamBands em dados reais 33

4.3 Determinando infraestruturas pelos protocolos nos SpamBands 35
4.3.1 SpamBand híbrido 38
4.4 Avaliação dos idiomas nos SpamBands 40
4.4.1 SpamBand chinês com mensagens sem texto 41
4.4.2 SpamBand com cinco idiomas 42
4.5 Determinando o comportamento do spammer pelo idioma dos SpamBands 45
4.6 Entendendo os tipos de campanhas através dos SpamBands 47

5 Encontrando grupos persistentes ao longo do tempo 51
5.1 Método para identificação de grupos colaborativos 51
5.2 Avaliando grupos colaborativos 54
5.2.1 Furto de contas do PayPal 55
5.2.2 Propagação de malware 56
5.2.3 Mensagens de phishing com idioma português 58

6 Conclusão e Trabalhos Futuros 61

Referências Bibliográficas 65

Apéndices 71

Apêndice A Estudos de casos de SpamBands 73
A.1 SpamBand HTTP/SOCKS: Venda de produtos farmacêuticos 73
A.2 SpamBand SMTP: Furto de contas da Apple 74
A.3 SpamBands especializados em phishing para bancos 76

Apêndice B Sistema desenvolvido para o CERT.br 79

Capítulo 1

Introdução

A batalha contra o abuso do *spam* é antiga e ainda persiste. O *spam* é uma mensagem de email não solicitada que é disseminada pela rede. As motivações daqueles que realizam essa prática, os *spammers*, são diversas, sendo as duas mais comuns (i) a propaganda de produtos e serviços ilegais como itens farmacêuticos e plataformas de vídeos com conteúdo adulto (ii) ataques de *phishing*, uma categoria de *spam* que visa furtar alguma informação pessoal do destinatário ou enganá-lo de forma a fornecer dinheiro para uma finalidade fictícia. Um dos problemas do *spam* são os recursos financeiros gastos para controlar este abuso que exige que diversas redes que compõem a Internet gastem com filtros para reduzir o tráfego [Fonseca et al., 2016; Sipior et al., 2004]. Além disso há o prejuízo social, onde mensagens legítimas são perdidas por má classificação de filtros de *spam* [Cormack, 2008].

Existem diversas frentes consideradas no combate ao *spam*. Alguns estudos buscam construir filtros eficazes que descartem mensagens indesejáveis na caixa de email do destinatário. Outros fazem a análise do comportamento do *spammer* na rede, para entender como o *spam* é disseminado, de onde ele se origina e como ele atravessa a rede sem que os transmissores sejam facilmente identificados. O objetivo desses trabalhos é identificar comportamentos na rede que permitam bloquear as mensagens tão logo quanto possível para evitar consumo de recursos de filtragem e possível armazenamento [Las-Casas et al., 2013b].

Em todos os casos citados, fica visível que o combate ao *spam* requer o entendimento de um sistema complexo de ofuscação usado pelo *spammer* em sua atividade, que envolve uma orquestração de atores e recursos que não são visíveis para o profissional que se dedica a esse combate. Para se manter oculto, o *spammer* busca disfarçar sua localização na rede, seja enviando suas mensagens a partir de múltiplas origens, como máquinas infectadas que se organizam em *botnets*, ou usando servidores especializados

que podem, por sua vez, se aproveitar de máquinas mal-configuradas na rede para se ocultar dos destinatários. Além disso, os *spammers* também utilizam um segundo nível de ofuscação que usa programas de transmissão que geram diversas mensagens diferentes com versões de um mesmo conteúdo básico, a fim de tentar ludibriar os filtros baseados em conteúdo [Cormack, 2008]. Nesse processo, tem importância o conceito de campanhas de *spam*, que são grupos de mensagens que possuem um mesmo objetivo, mas que foram alteradas por métodos de ofuscação para tentar ludibriar filtros [Calais et al., 2008].

Este trabalho utiliza uma abordagem que combina aspectos de campanhas com aspectos de comportamento de rede a fim de tentar lançar mais luz sobre esse elemento orquestrador subjacente ao processo de envio de *spam*. Para este fim, utilizamos tanto características baseadas no conteúdo da mensagem, para permitir a identificação das campanhas de *spam*, quanto elementos do tráfego de rede, para identificar as máquinas originadoras de cada campanha. Com isso, propomos o *SpamBand*, um método capaz de identificar, a partir de um grupo de mensagens de *spam*, grupos de máquinas (*SpamBands*) na rede que se encontram em um certo momento sob o controle de um orquestrador oculto, o *spammer*. Um dos objetivos do método é fornecer ao profissional da área uma forma de entender como se organizam as máquinas de onde se origina o *spam* que ele observa.

As principais contribuições dessa dissertação são:

- Proposta de um novo algoritmo para identificação de campanhas de *spam* na árvore de padrões frequentes (FP-Tree) utilizada na literatura como uma estrutura de dados para representar mensagens de *spam*. A diferença do nosso algoritmo para o proposto na literatura é a desvinculação de parâmetros dependentes da estrutura da FP-Tree (número de filhos e altura mínimos) e a associação de toda mensagem a uma e apenas uma campanha de *spam*.

- Apresentação do algoritmo para detecção dos *SpamBands* e aplicação da técnica a um conjunto de dados composto por 650 milhões de mensagens coletadas por *honeypots* de baixa interatividade. Realizamos uma análise dos *SpamBands* detectados que levam a importantes conclusões sobre o tráfego de *spam* enviado através de servidores vulneráveis. Observamos que *spams* enviados em idiomas ocidentais têm um comportamento diferente de *spams* enviados com idiomas orientais. Por exemplo, o *phishing* está bastante relacionado a grupos que exploram o *honeypot* como *relay* aberto (indicativo de *botnets*) e estão ligados a idiomas ocidentais. Em contrapartida, *spams* de propaganda exploram o *honeypot* como

proxy aberto (indicativo de uso de máquinas pertencentes ao *spammer*) e parecem estar ligados a idiomas orientais, como chinês, japonês e russo.

- Discussão de estudos de casos de *SpamBands* que sugerem como são organizados *SpamBands* específicos que enviam campanhas de *phishing* de bancos ou que utilizam plataformas de nuvem, como a Microsoft Azure, para o envio de suas mensagens. Observamos ainda a existência de *spammers* que podem estar utilizando tanto *botnets* quanto infraestruturas próprias para o envio.

- Elaboração de uma técnica (baseada em trabalhos na literatura) para determinar grupos de endereços IP que se relacionam nos diversos *SpamBands* encontrados ao longo do período avaliado (grupos colaborativos). Encontramos que grande parte dos grupos detectados atuam durante poucos dias com intervalos significativos entre suas atividades.

O texto dessa dissertação é organizado da forma que segue. O capítulo 2 apresenta uma avaliação cronológica das diversas técnicas de combate ao *spam* desenvolvidas ao longo dos anos e como nossa contribuição se encaixa dentro desse cenário. O objetivo dessa seção também é apresentar, de forma sucinta, o histórico de trabalho ao leitor não familiarizado com o tema. O capítulo 3 apresenta a arquitetura de coleta e uma breve análise dos dados que utilizamos nesse trabalho. Ainda no mesmo capítulo é apresentada a técnica utilizada para detectar campanhas de *spam*: a estrutura de dados FP-Tree (árvore de padrões frequentes) proposta na literatura e o novo algoritmo para identificação de campanhas nessa estrutura. Também apresentamos uma análise minuciosa sobre o atributo de idioma utilizado na geração da FP-Tree e mostramos um exemplo de como é feita a detecção de campanhas nessa estrutura. O capítulo 4 apresenta o algoritmo de detecção de *SpamBands* e como esse algoritmo opera sobre dados reais. Esse capítulo também apresenta um estudo detalhado dos *SpamBands* e como eles ajudam a identificar padrões sobre o envio de trafego no conjunto de dados utilizado. O capítulo 5 mostra a técnica que utilizamos para detectar grupos colaborativos a partir dos *SpamBands* detectados ao longo dos dias e uma avaliação de como esses grupos atacam os *honeypots*. Por fim, o capítulo 6 sumariza os resultados encontrados nessa dissertação e discute trabalhos futuros.

Capítulo 2

Spam e Técnicas de Combate

Tão importante quanto oferecer uma técnica é contextualizar o problema e onde a contribuição se encaixa. Diante do vasto cenário que o combate ao *spam* produziu, buscamos adentrar um pouco mais sobre as diversas técnicas existentes para situar o nosso trabalho. Começamos com a figura 2.1 que mostra a evolução das principais técnicas de combate ao *spam* ao longo dos anos. É importante ressaltar que o ano mostrado na figura 2.1 pode não coincidir com o primeiro trabalho sobre o tema: nosso foco aqui é mostrar quando a comunidade científica começou a reagir diante as propostas.

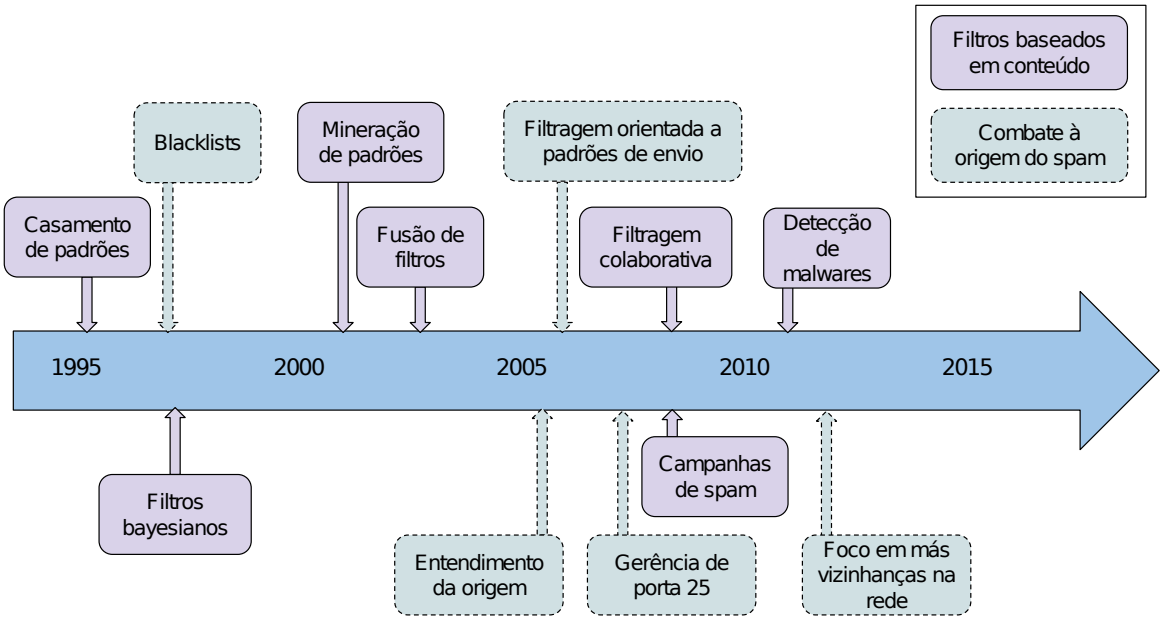


Figura 2.1: Linha do tempo das principais técnicas de combate ao *spam*.

Fica evidente na linha do tempo mostrada na figura 2.1 que a batalha contra os *spammers* é antiga. Os primeiros estudos que começaram a observar os prejuízos do *spam* surgiram na década de 1990 [Cranor & LaMacchia, 1998; Junod, 1997; Cohen, 1996]. Esses estudos mostravam tanto prejuízos envolvendo tempo gasto pelos usuários para diferenciar mensagens quanto prejuízos financeiros devido ao tráfego gerado na rede e o consumo de recursos para realizar filtragens [Pantel et al., 1998]. O *spam* dessa época era caracterizado como mensagens não solicitadas com o objetivo claro de realizar propagandas [Neumann & Weinstein, 1997], ocupando uma pequena parcela (cerca de 10%) das mensagens de email na caixa de entrada do usuário [Hawley, 1997]. Identificado como um problema, começaram a surgir, nesse período, trabalhos focados na filtragem por conteúdo dessas mensagens, utilizando modelos estatísticos (mais detalhes na seção 2.1) [Sahami et al., 1998; Drucker et al., 1999; Sahami et al., 1998].

O início da década de 2000 foi marcado pelo rápido crescimento do número de mensagens de *spam* na rede, ultrapassando o número de mensagens legítimas e indicando que o *spammer* havia encontrado novos meios para abusar o sistema de correio eletrônico [Sipior et al., 2004]. Esse cenário levou a uma crescimento de técnicas propostas para combater os novos métodos utilizados pelos *spammers* para disseminação de *spam*, como geração aleatória de texto no corpo das mensagens para confundir filtros probabilísticos baseados em conteúdo e o abuso de servidores de *proxy* e *relay* mal configurados para ludibriar *blacklists* [Hird, 2002; Ioannidis, 2003; Fawcett, 2003]. Os *spammers* também começaram a utilizar máquinas infectadas com *malware* para o envio de *spam* (*bots*), tornando a tarefa de bloquear a origem do spam bastante difícil uma vez que máquinas de usuários legítimos eram utilizadas para enviar *spam* [Leiba et al., 2005; Clayton, 2004; Levy, 2003]. Ainda no início da década de 2000, observou-se um aumento no número de mensagens com o intuito de induzir o usuário a fornecer dados pessoais e financeiros. Esse tipo de mensagem, conhecido como *phishing*, foi responsável por um enorme prejuízo financeiro aos usuários finais, agravando ainda mais a situação (mais detalhes na seção 2.4) [Hong, 2012; van der Merwe et al., 2005].

A partir de 2005, cerca de 80% das mensagens de email enviadas eram *spam* [Ramzan & Wüest, 2007; Blanzieri & Bryl, 2008]. O consumo de recursos causados pelo tráfego dessas mensagens na rede e a consequente defasagem das técnicas baseadas em conteúdo no destino levaram a abordagens que buscam bloquear o *spam* tão cedo quanto possível na rede, como gerência da porta 25 e uso de *blacklists*, a ganharem força (mais detalhes na seção 2.2) [Gellens & Klensin, 2011; Ramachandran et al., 2007]. Começaram a surgir trabalhos que visavam entender a infraestrutura utilizada pelo *spammer* e estudar melhor a orquestração por trás do problema, originando o conceito de campanhas de *spam* (mais detalhes na seção 2.3) [Kreibich et al., 2009].

Estratégias que utilizavam filtros entre a origem e o destino do *spam* também começaram a se tornar mais efetivas: (i) a filtragem colaborativa, que se beneficiava da crescente popularização dos webmails, que facilitavam a integração de dados de diversos usuários e (ii) ferramentas que combinavam diversos métodos de filtragem existentes na literatura executadas sobre todas as mensagens recebidas em servidores de email como, por exemplo, o SpamAssassin que constrói suas regras com grandes bases de dados de várias fontes [Attenberg et al., 2009; Lynam et al., 2006].

Apesar do esforço para erradicar o *spam*, estudos realizados nos últimos anos mostraram que o agente orquestrador, o *spammer*, ainda persiste em suas atividades e que as mensagens enviadas continuam com os objetivos de realizar ataques de *phishing* e anunciar produtos ilegais. [Las-Casas et al., 2016; Hong, 2012]. O problema para identificar com clareza como esse abuso é organizado ainda é um desafio: os *spammers* continuam utilizando *botnets* e servidores dedicados em redes coniventes para o envio de suas mensagens, explorando servidores vulneráveis como *proxies* e *relays* abertos [Las-Casas et al., 2013b; Fazzion et al., 2014]. Esse histórico de trabalhos deixam claro que a batalha contra o *spammer* persiste e motiva nosso trabalho a continuar investigando meios de combater esse abuso.

2.1 Identificação e filtragem de spam no destino

Os primeiros filtros para combater o *spam* se baseavam em casamento de padrões que utilizavam regras configuradas manualmente pelo administrador da rede de destino das mensagens. Isso levava a um esforço do administrador de rede para manter as regras do casamento sempre atualizadas e tratar casos que geravam falsos-positivos uma vez que era um problema para o usuário ter mensagens legítimas bloqueadas [Cranor & LaMacchia, 1998]. Alguns trabalhos começaram a observar a adaptação do *spammer* contra os filtros existentes, tornando as técnicas utilizadas até então ineficientes [Sahami et al., 1998; Pantel et al., 1998]. Dessa forma, surgiram propostas de modelos adaptativos de geração de regras para classificar mensagens de *spam* [Sahami et al., 1998; Drucker et al., 1999; Pantel et al., 1998]. Um dos filtros mais famosos atualmente, o filtro bayesiano, teve suas origens em 1996 [Cohen, 1996] mas a ideia só ganhou forma em 1998 com alguns trabalhos que compararam o modelo em bases de dados significativas [Sahami et al., 1998; Pantel et al., 1998]. Esses filtros utilizam a inferência bayesiana, que designa uma probabilidade a cada email de ser legítimo ou *spam* dado a ocorrência de algumas características aprendidas através de um conjunto de mensagens de treinamento [Sahami et al., 1998; Androutsopoulos et al., 2000; Pantel

et al., 1998; Cohen, 1996]. Essa técnica permitiu automatizar a atualização das regras sempre que aparecia um novo tipo de *spam*: bastava inserir a nova mensagem de *spam* no conjunto de treino e gerar o modelo bayesiano novamente. O bom desempenho e eficácia da técnica bayesiana em *spam* teve um impacto positivo na comunidade e inspirou a utilização de outros modelos de aprendizado de máquinas como o *support vector machine* (SVM) [Drucker et al., 1999; Androutsopoulos et al., 2000].

Os filtros bayesianos foram bastante eficazes até o *spammer* criar mecanismos de geração aleatória de texto no corpo de mensagens de forma a ludibriar os filtros baseados em conteúdos além de preservar as técnicas para fazer o usuário abrir as mensagens como, por exemplo, colocar assuntos parecidos com a de mensagens legítimas [Hird, 2002; Skoll, 2003]. Esse fator colaborou para o rápido crescimento do número de mensagens de *spam* na rede [Wagner, 2002; Fallows, 2003] que levaram a necessidade de novas abordagens como a utilização de algoritmos para descobrir padrões em bases de *spam* e utilizar o conhecimento para classificar mensagens [Segal et al., 2004; Rigoutsos & Huynh, 2004; Kolcz et al., 2004]. A grande diferença entre descobrimento de padrões e modelos estatísticos é a ausência de um casamento exato: no modelo bayesiano há a necessidade de definir o que é um termo para fazer a inferência enquanto a classificação por padrões não precisa dessa definição. Enquanto as palavras *spam*, *spamming* e *spammer* são vistas pelo modelo bayesiano como três termos, as técnicas de descobrimento de padrões consideram um único padrão: o radical *spam*. Uma vez que padrões são identificados, é realizado uma filtragem para recuperar um conjunto das características mais relevantes e gerar uma assinatura para a mensagem que é aplicada a emails utilizando casamento de padrões [Kolcz et al., 2004; Wittel & Wu, 2004]. Essa técnica se adaptou bem no combate aos métodos utilizados pelo *spammer* para ludibriar filtros estatísticos, como o uso de termos aleatórios no corpo de mensagens [Kolcz et al., 2004; Wittel & Wu, 2004].

Entretanto, o caráter evolutivo do *spammer* fez com que as técnicas baseadas em padrões também ficassem ineficazes. Diante desse cenário, surgiram algumas ferramentas, como o *spam*Assassin [Mason, 2002], que combinavam diversas regras para realizar a filtragem das mensagens de *spam*. Essa abordagem ganhou espaço uma vez que a combinação de diversos métodos resultou em uma melhoria na detecção de mensagens de *spam*. Vale ressaltar que esse tipo de ferramenta persiste até os dias atuais. Por fim, a última abordagem de conteúdo que também ganhou força recentemente foi a filtragem colaborativa que, apesar de problemas de implantação em seu início [Hird, 2002; Cunningham et al., 2003], ganhou força atualmente com os serviços de webmails. A principal razão disso é que essa técnica propõe combinar relatos de *spams* feitos por usuários em um repositório central para servir de consulta para novas mensagens

e os repositórios unificados de email como GMail, Yahoo e Hotmail favorecem essa técnica [Gray & Haahr, 2004; Attenberg et al., 2009]. Entretanto, a desvantagem dela é que a dependência do usuário torna a classificação subjetiva e produz falsos-positivos para usuários cuja classificação difere da maioria [Gray & Haahr, 2004].

2.2 Identificação e combate à origem do spam

Durante algum tempo vários esforços da comunidade para conter o *spam* se concentraram em criar filtros baseados em conteúdo no destino [Sahami et al., 1998; Drucker et al., 1999; Sakkis et al., 2003]. Entretanto, esses filtros começaram a se tornar obsoletos rapidamente, demandando sempre novas técnicas que sobrepujassem as novas estratégias utilizadas pelos *spammers* [Wittel & Wu, 2004]. Diante disso, começaram a ganhar força propostas de combate baseadas em rede (e não conteúdo) que impedissem que o *spam* chegasse ao seu destino evitando o consumo de recursos de processamento, armazenamento e rede.

Uma das primeiras abordagens de combate à origem focaram no uso de *blacklists* e consistiam em bloquear mensagens de domínios específicos [Junod, 1997]. Entretanto, foi observado que esse tipo de bloqueio não era eficaz contra *spammers* que atuavam de forma descentralizada (como o uso de servidores mal configurados) pois o bloqueio causava danos a usuários legítimos [Neumann & Weinstein, 1997]. Isso levou a abordagens menos agressivas como o bloqueio por endereços IP. Os endereços de possíveis *spammers* eram registrados em listas que eram consultadas por servidores de email. Essa estratégia ficou conhecida como DNSBL[1] e um dos mais famosos mantenedores de *blacklists* é o SpamHaus. A eficácia dessa técnica estava no bloqueio de *spammers* conhecidos e de servidores de *proxies* e *relays* abertos [Ramachandran & Feamster, 2006].

Algumas questões, porém, começaram a dificultar o acompanhamento do *spammer* pelas *blacklists*. Primeiro que o endereço IP não identificava unicamente um dispositivo na rede, uma vez que muitas máquinas utilizavam endereços dinâmicos. Segundo, o *spammer* começou a empregar *bots* para o envio do *spam* [Jung & Sit, 2004; Ramachandran & Feamster, 2006]. Esses fatos tornaram bastante difícil a tarefa de combater o *spammer* sem efeitos indesejáveis para usuários legítimos como o bloqueio de todas as mensagens de email deste usuário. Dessa forma, os algoritmos utilizados por *blacklists* tiveram que reunir mais informações para classificar um endereço IP, levando a grandes atrasos para atualizar a lista de endereços. Diante dessa situação,

[1]DNSBL: abreviação para Domain Name System-based Blackhole Lists

surgiram estudos focados em analisar o *spam* próximo à origem de forma a ajudar o aprimoramento das diversas ferramentas de filtragem.

Considerado um dos primeiros trabalhos que estudam o *spammer* a nível de rede, Ramachandran et al. [Ramachandran & Feamster, 2006] observaram o comportamento do *spammer* do ponto de vista de rede a partir de conexões TCP e endereços IP que são atributos difíceis de forjar. O estudo lançou luz em diversos elementos ainda desconhecidos: (i) grande parte do tráfego observado vinha de *bots* que tinham vida curta e enviavam poucas mensagens no período de atividade; (ii) mostrou que o *spam* vinha de poucas faixas de endereçamento IP indicando possíveis fontes do abuso; (iii) o *spammer* se utilizava de sequestros de prefixos para realizar rápidas conexões e enviar suas mensagens. Essas observações mostram oportunidades de combater o *spammer* a nível de rede e serviram como motivação para diversos trabalhos.

Continuando o trabalho anterior, os mesmos autores estudaram o perfil de envio dos *spammers* na rede de forma a tentar lançar mais luz na ofuscação ocasionada por *botnets* [Ramachandran et al., 2007]. O estudo procurou focar em padrões de envio do *spam*, como endereços IP que enviam mensagens em intervalos de tempo regulares, em vez de estudar unicamente endereços IP que podem variar ao longo do tempo. Os autores agruparam endereços IP semelhantes ao modo de envio para certos domínios e extraíram características como o número de mensagens que um endereço IP envia para um determinado domínio e a frequência desse envio. Eles utilizaram essa informação para identificar novos endereços IP que deveriam ser listados em *blacklists*. O agrupamento de máquinas também foi defendido por Xie et al. [Xie et al., 2008] que argumentam a impossibilidade de diferenciar o comportamento de máquinas maliciosas do comportamento de *bots* observações individuais uma vez que as técnicas dos *spammers* tendem a simular comportamentos legítimos. Um dos trabalhos mais recentes que buscam identificar padrões de envio próximo à origem foi o de Las-Casas et al. [Las-Casas et al., 2013a] que propuseram uma técnica para detectar padrões de envios de endereços IP utilizando algoritmos de aprendizado de máquina.

Focando em indícios de que máquinas maliciosas estavam concentradas em pequenas faixas de endereçamento IP, Moura et al. [Moura et al., 2011] introduziram o conceito de *Bad Neighborhoods* [Van Wanrooij & Pras, 2010] em *spam*: quanto mais máquinas enviando *spam* em uma sub-rede maior a probabilidade que uma máquina qualquer se comporte da mesma maneira dentro da mesma sub-rede. O trabalho observou a existência de dois tipos de redes: aquelas que são abusadas por serem má administradas e aquelas que compactuam com a atividade do *spammer*. O trabalho reforça o estudo feito por Peeking et al. [Pathak et al., 2008] que classifica *spammers* em dois tipos: *spammer*s que enviam um volume muito grande de mensagens através

de poucos endereços (*high spammers*) e *spammer*s que enviam carga leve de diversas máquinas (*low spammers*). Em geral, Moura et al. mostram que *high spammers* estão mais concentrados em redes que parecem compactuar com o tráfego e que *low spammers* estão mais concentrados em redes com baixa proteção que possuem um maior número de *bots*.

Uma outra frente que, apesar do foco não ser unicamente *spam*, visa mitigar o problema de sequestro de prefixos na rede que é uma das técnicas utilizadas pelo *spammer* para enviar suas mensagens. Essa frente se concentra na criação de ferramentas e técnicas de monitoramento de anúncios BGP para encontrar anomalias na rede. Lad et al. [Lad et al., 2006] propuseram um sistema de notificação em tempo real de sequestro de prefixos. Basicamente, o sistema exigia um registro de um prefixo pelo usuário e a partir da informação da origem do prefixo, o sistema monitorava anúncios BGP de projetos como o RouteViews e RIPE e buscava por alguma alteração na origem, reportando ao usuário em caso de algum incidente. Surgiram ainda abordagens baseadas em reputação dos ASes, cuja finalidade eram guiar sistemas de monitoramento para medir rotas de ASes com baixa reputação [Stone-Gross et al., 2009; Wagner et al., 2013]. O objetivo dessa abordagem era não depender de cadastros de usuários, classificando ASes pelo número de incidentes ocorridos dentro de seu domínio. O grande problema era que esses sistemas não conseguiam distinguir ASes maliciosos de ASes legítimos abusados. Esse assunto foi abordado recentemente por Konte et al. [Konte et al., 2015] que propuseram um sistema que observa o histórico de atividades de um AS, como a taxa de mudança de provedores e parcerias com outros ASes, para determiná-lo como um AS abusado ou AS malicioso.

Por fim, citamos a gerência da porta 25 (RFC 4409) que vem ganhando espaço atualmente por ser a única técnica que combate o *spammer* na origem[2]. Essa técnica se diferencia das técnicas anteriores pelo seu potencial de bloquear o *spam na* origem, sem que as mensagens sequer consigam ser enviadas e não consumam nenhum tipo de recurso na rede enquanto as outras abordagens atuam no caminho da mensagem o que ainda permite o desperdício de recursos até o *spam* ser detectado e filtrado. Essa técnica visa separar a submissão de mensagens do seu transporte. Em outras palavras, o remetente terá que autenticar para enviar mensagens e a porta 25 será exclusiva de máquinas autorizadas para realizar o encaminhamento das mensagens sendo bloqueada para uso residencial. A técnica, além de reduzir o tráfego de *spam*, permite uma melhor rastreabilidade em caso de algum tipo de abuso uma vez que o *spammer* só poderá enviar email se autenticado.

[2]http://www.antispam.br/admin/porta25/

2.3 Campanhas de spam

Campanhas de *spam* podem ser vistas como um esforço realizado pelo *spammer* para entregar uma mensagem particular a um grupo de destinatários [Stone-Gross et al., 2011]. Kreibich et al. [Kreibich et al., 2008] criaram mecanismos para infiltrar em uma *botnet* e interceptar mensagens trocadas por *botnets* e centrais de comando. Eles observaram que o tipo de uma campanha dependia do tipo de *malware* instalado no *bot*. Por exemplo, uma campanha para disseminar propagandas farmacêuticas utilizavam *bots* por um determinado *malware* enquanto ofertas de trabalho utilizavam *bots* por um segundo tipo de *malware*. Os autores também mostraram que os *bots* utilizavam um padrão leve de envio que dificultava a detecção por *blacklists*.

Os mesmos autores continuaram suas observações dentro da mesma *botnet* [Kreibich et al., 2009], cujos resultados são de interesse desse trabalho: (i) além do *spammer* subdividir *bots* para envio de acordo com o *malware*, esses *bots* possuem templates de mensagens com macros a serem substituídas (e.g., o assunto) e cada template é instanciado em campanhas de *spam* específicas (e.g., campanha para uma determinada marca); (ii) mais de 95% das campanhas observadas duraram, no máximo, 24 horas; (iii) metade das campanhas empregaram mais de um template (e.g., mensagens com dois assuntos diferentes e um mesmo *malware*). Esses resultados deixa claro que analisar apenas atributos de redes não é suficiente para detectar a infraestrutura utilizada por um agente orquestrador. Apesar do trabalho de Kreibich et al. trazer diversas observações importantes sobre como o *spammer* se organiza para enviar campanhas de *spam*, não é mencionada a técnica utilizada para identificar as campanhas observadas na *botnet*.

Em meio ao grande volume de mensagens de *spam* na rede, a detecção automática e eficiente de campanhas se tornou necessária e surgiram algumas propostas. O trabalho de Calais et al. [Calais et al., 2008] apresenta uma técnica de agrupamento baseada na árvore de padrões frequentes (FP-Tree). Essa estrutura, descrita no trabalho, mostrou robustez pois além de não ser suscetível a pequenas variações em partes das mensagens, como URLs com partes geradas aleatoriamente, ela não depende apenas de um único atributo. Entretanto, o algoritmo para detectar campanhas na FP-Tree apresentado no artigo não foi descrito formalmente, deixando em aberto diversas questões sobre seu funcionamento.

Outro trabalho que busca detectar campanhas de *spam* é o de Shoeb et al. [Md Shoeb et al., 2015] que realiza a detecção de campanhas traduzindo as URLs das mensagens de *spam* para endereços IP e agrupando mensagens com URLs que mapeiam para o mesmo destino. O desafio da técnica é que as URLs utilizadas pelos

spammers, em geral, tem curta duração, exigindo uma tradução em tempo real. Além disso, o *spammer* utiliza redirecionamentos HTTP e JavaScript para ofuscar o endereço IP real da URL e o artigo explora apenas o primeiro método.

É possível observar que campanhas de *spam* não são suficientes para detectar o agente orquestrador: outros trabalhos mostram que o *spammer* utiliza apenas partes de uma estrutura global para enviar mensagens de cada campanha, exigindo técnicas mais sofisticadas que relacionem campanhas para identificar a infraestrutura de rede utilizada [Kanich et al., 2008]. Isso vai ao encontro ao nosso trabalho, que combina atributos de rede e de conteúdo para detectar a orquestração da infraestrutura usada pelo *spammer*.

2.4 Combate ao phishing

O *phishing* é um tipo de *spam* que visa enganar e lesar o destinatário como, por exemplo, obter senhas de cartões de crédito ou arrecadar dinheiro para campanhas falsas. O phisher (uma categoria do *spammer* que envia *phishing*) utiliza técnicas de engenharia social e mensagens personalizadas para ludibriar a vítima [Hong, 2012]. O aumento significativo dessas mensagens na rede no início da década de 2000 começou a despertar a comunidade acadêmica para estudar esse tipo de ataque. Os primeiros trabalhos tentaram identificar quais características eram inatas ao *phishing* e que o diferenciava do *spam* tradicional [Drake et al., 2004]. Entre essas características, temos: (i) a imitação de companhias renomadas como bancos (CitiBank), sites de compras (eBay) ou plataformas de pagamentos (PayPal); (ii) usar técnicas para tornar o domínio do remetente da mensagem parecido com o domínio da organização alvo do ataque. Por exemplo, o remetente da mensagem de *phishing* que visa furtar informações do Banco do Brasil poderia ser *atendimento@atendimento-bb.com.br* enquanto uma mensagem verdadeira seria *atendimento@atendimento.bb.com.br*; (iii) envio para menos destinatários que o *spam* tradicional; (iv) uso de coletores na Web que associem emails a nomes de forma a tornar a mensagem bastante específica; e (v) utilização de sites que imitam a aparência dos sites de companhias legítimas e ofuscação desses links nas mensagens para parecerem reais. Há alguns trabalhos que mostram que o phisher chega a traçar os perfis das vítimas de forma a mandar conteúdos especializados, o que leva até mesmo pessoas experientes a serem vítimas do *phishing* [Hong, 2012].

As diferenças do *phishing* para o *spam* tradicional fizeram com que as técnicas de combate ao spam não fossem muito efetivas por causa do nível de personalização utilizado nas mensagens, o que tornava muito mais difícil separar de mensagens legí-

timas [Drake et al., 2004]. Dessa forma, alguns métodos começaram a ser utilizados exclusivamente contra o *phishing*. Abordagens iniciais envolviam o treinamento do usuário onde companhias alertavam seus usuários sobre possíveis ataques [Kumaraguru et al., 2007]. Porém, avisos não foram suficientes uma vez que mensagens de *phishing* continuavam a lesar usuários [Almomani et al., 2013] e propostas de técnicas automatizadas começaram a surgir. As duas mais famosas e que persistem até hoje são: (i) analisar o código fonte das URLs no corpo da mensagem a fim de detectar se a página copia uma página legítima[3] [Ludl et al., 2007; Whittaker et al., 2010; Geng & Hong, 2016] e (ii) minerar padrões no texto da mensagem para identificar padrões recorrentes de *phishing* como senso de urgência e solicitações de recuperação de senhas que são formas de interação do phisher com o usuário de forma a fazê-lo abrir links/anexos ou responder o email. [Chandrasekaran et al., 2006; Bergholz et al., 2010; Las-Casas et al., 2016].

2.5 Determinando a infraestrutura do spammer

Nossa visão revendo os diversos trabalhos produzidos é clara: o *spammer* utiliza técnicas que dificultam a sua detecção tanto por filtros baseados em conteúdo quanto filtros baseados em características de rede. Dessa forma, nosso trabalho propõe fazer a ligação entre essas duas frentes e conseguir revelar elementos que antes não eram possíveis. Até o limite de nosso conhecimento, o único trabalho que tentou fazer essa ligação foi o trabalho de Li & Hsieh [Li & Hsieh, 2006], que agrupou endereços IP baseados nas URLs presentes nas mensagens: se dois endereços IP enviaram mensagens com a mesma URL, eles se conectam no grafo. A partir dos grafos gerados, os autores mostraram, empiricamente, que endereços IP associados a múltiplos grupos têm uma maior chance de enviar *spam* em um futuro próximo e que fazer filtros baseados nessa abordagem poderia bloquear 70 a 90% dos *spams*.

Existem duas diferenças essenciais em nosso trabalho. A primeira diferença é que consideramos campanhas de *spam* como ligação entre endereços IP, que é um conceito mais abrangente que apenas URLs, uma vez que existem mensagens que não utilizam URLs ou ofuscam sua estrutura, que é algo que tratamos quando determinamos quais mensagens fazem parte de uma mesma campanha. A segunda diferença é que nosso trabalho considera a existência de diferentes máquinas que recebem o mesmo endereço IP por estarem em um mesmo NATs, o que pode relacionar grupos distintos. Por exemplo, duas máquinas em uma rede com NAT podem pertencer a diferentes *spammers*.

[3] O projeto PhishTank reporta diversas URLs falsas através da colaboração de usuários e empresas.

Em nosso trabalho, consideramos esse tipo de ruído de forma a identificar diferentes estruturas utilizadas por diferentes *spammers*, o que não é realizado no trabalho de Li & Hsieh [Li & Hsieh, 2006].

Capítulo 3

Detecção e análise das campanhas de spam

Neste capítulo descrevemos as técnicas utilizadas para detectar campanhas de *spam*. Uma campanha de spam é definida como um conjunto de mensagem enviadas com um objetivo claro como a venda de um determinado produto ou furto de informações de clientes de um banco específico. Na seção 2.3 discutimos duas técnicas para detectar campanhas de *spam*. A primeira delas, proposta por Shoeb et al., propõe traduzir URLs para endereços IP e agrupar mensagens com URLs servidas pelo mesmo endereço IP. Entretanto, esse método não é viável para o nosso trabalho uma vez que realizar requisições DNS para traduzir os endereços colocaria em risco a anonimização dos *honeypots* de um projeto em andamento: há uma chance do *spammer* ter acesso a algum servidor DNS e verificar que as URLs de mensagens enviadas por um *honeypot* geram inúmeras requisições de poucas fontes.

Nossa opção foi adotar, parcialmente, o segundo método, proposto por [Calais et al., 2008]: utilizamos a representação das mensagens na estrutura de dados FP-Tree (seção 3.2 e 3.3) e propomos um novo método para identificar campanhas nessa estrutura (seção 3.4). Também apresentamos um exemplo do funcionamento de detecção de campanhas na seção 3.5. Na seção 3.6 mostramos como identificamos o objetivo de cada campanha, ou seja, como determinamos se uma campanha está relacionada a propaganda ou *phishing*. A próxima seção descreve o sistema de coleta das mensagens de *spam* utilizadas em todo esse trabalho.

3.1 Coleta e avaliação dos dados

As mensagens utilizadas nesse trabalho foram obtidas através de um projeto realizado em parceria com o CERT.br (Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil) que coletam, diariamente, mensagens utilizando *honeypots* de baixa interatividade. Nesse trabalho, utilizamos quatorze *honeypots* instalados em diferentes Sistemas Autônomos (ASes) em dez *country codes*: dois no Brasil, dois nos Estados Unidos e um *honeypot* em cada um dos seguintes *country codes*: Argentina, Austrália, Áustria, Chile, Equador, Holanda, Hong Kong, Reino Unido, Taiwan e Uruguai. Estes *honeypots* são coletores que simulam servidores mal configurados de *relays* (protocolo SMTP) e *proxies* abertos (protocolos HTTP e SOCKS).

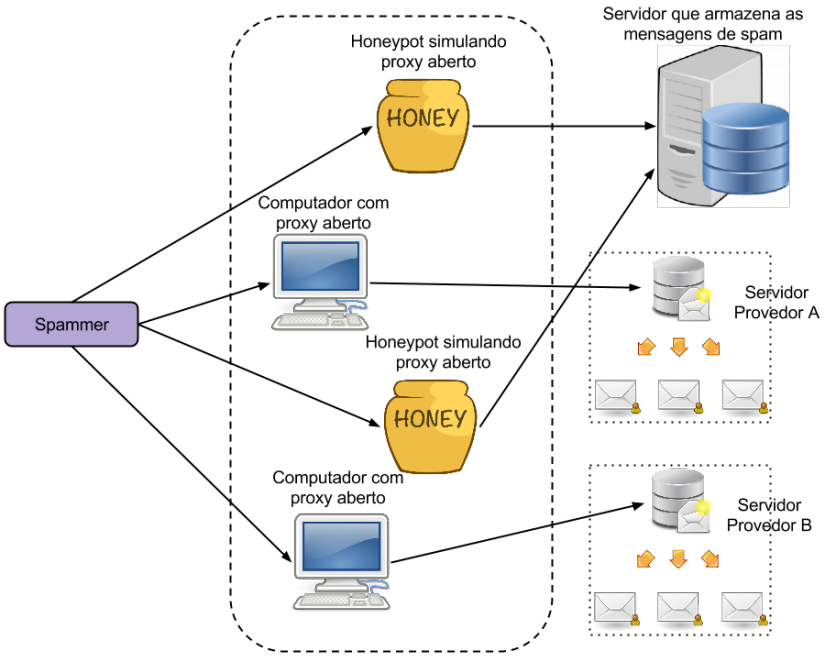


Figura 3.1: Arquitetura de coleta com *honeypots*.

Quando um *spammer* se conecta ao servidor SMTP de um *honeypot*, ele é levado a crer que está interagindo com um servidor SMTP operando como um *relay* aberto, que encaminham mensagens de e-mail recebidas para outros servidores SMTP. Quando uma máquina se conecta a um *honeypot* através dos protocolos HTTP ou SOCKS, é levada a crer que é capaz de estabelecer conexões com outros servidores SMTP na rede. Estes serviços são frequentemente utilizados para o envio de *spam*. A figura 3.1 mostra a a arquitetura de *honeypots* na coleta feita pelo CERT.br. Neste trabalho, utilizamos dados coletados durante o período de dois meses: 01 de abril 2016 a 31 de maio de 2016.

Como os *honeypots* não prestam serviço para nenhuma rede e não são anunciados publicamente, assumimos que toda a interação com os *honeypots* provém de *spammers* e as mensagens de *spam* são armazenadas localmente. Nenhuma mensagem de *spam* é efetivamente entregue ao seu destino e nem conexões SMTP via proxies efetivamente estabelecidas—exceto mensagens classificadas como mensagens de teste segundo regras pré-definidas como, por exemplo, verificando presença de texto específico no assunto ou corpo da mensagem. As informações de conexões que podem variar ao longo do tempo e que são utilizadas nesse trabalho, como o número do sistema autônomo e código do *country code* da máquina que se conecta ao *honeypot*, são obtidas no momento da conexão.

Durante o período analisado de dois meses, foram coletadas 650 milhões de mensagens de *spam* enviadas por 147.361 endereços IP distintos pertencentes a 4.716 sistemas autônomos localizados em 173 *country codes*. Nós observamos que grande parte das mensagens de *spam* coletadas (74,16%) tentam ser enviadas por poucos endereços IP (9,19%) que exploram o *honeypot* como proxy aberto. Esse comportamento já foi mencionado na literatura como típico de *spammers* que utilizam servidores dedicados para o envio e que procuram ofuscar sua identidade utilizando máquinas intermediárias [Las-Casas et al., 2013b]. O restante das mensagens (25,84%) é enviada por um grande número de endereços IP (90,81%) explorando o *honeypot* como relay aberto. Esse comportamento é típico de máquinas de usuários legítimos infectadas por *malware* onde pequenas quantidades de mensagens de *spam* são enviadas para tornar o tráfego imperceptível ao usuário [Xie et al., 2008]. É importante salientar que encontramos apenas um endereço IP que explora os *honeypots* tanto como relay quanto proxy aberto e que os endereços IP que exploram o *honeypot* como proxy aberto com ambos protocolos (HTTP e SOCKS) são apenas 127, ou seja, mais de 99,99% dos endereços IP detectados utilizam apenas um dos protocolos.

3.2 Estrutura de dados FP-Tree

Em nosso trabalho, utilizamos a estrutura de dados FP-Tree aplicada em mensagens de *spam* apresentada no artigo de Calais et al. [Calais et al., 2008]. Os dois requisitos para a construção da FP-Tree são um conjunto de mensagens $\mathcal{M}$ com $\mid \mathcal{M} \mid = n$ e um conjunto de características $\mathcal{C}$ a serem extraídas de cada mensagem. Para cada mensagem, cada característica extraída $\mathcal{C}_i$ produz um ou mais atributos $\mathcal{A}$ e a mensagem é representada pelo conjunto desses atributos $(\mathcal{A}_i, \mathcal{A}_{i+1}, \ldots, \mathcal{A}_m$ com $\mid \mathcal{A} \mid \geq \mid \mathcal{C} \mid)$. Por exemplo, a característica $\mathcal{C}_i$ =URL pode gerar dois atributos como $\mathcal{A}_1$ =www.exemploA.com

e $\mathcal{A}_2$ =www.exemploB.com. Neste trabalho, consideramos o conjunto de características $\mathcal{C}$ sendo o mesmo utilizado no trabalho de Calais et al. [Calais et al., 2008] e explicitamos cada uma delas a seguir:

- **Leiaute:** o leiaute é a codificação da formatação da mensagem em uma sequência de caracteres. No trabalho, utilizamos linha de texto como bloco de construção do leiaute e representamos cada bloco por quatro caracteres: linha em branco (B), linha com URL (U), linha com HTML (H) e linha com texto (T). Essa ordem é considerada em termos de empate: linhas com URLs em HTML são consideradas como linhas com URLs e linhas com texto e HTML são consideradas linhas com HTML.

- **URLs:** para cada URL presente na mensagem, dividimos sua estrutura por / e ? e tornamos cada parte um atributo diferente, não considerando repetição. Essa é uma característica alvo de muitas variações, conforme avaliada no trabalho de Calais et al..

- **Assunto:** Consideramos o assunto de cada mensagem em sua forma original.

- **Destinatários:** Consideramos todos os endereços de email dos destinatários envolvidos na mensagem.

- **Remetente:** O email assinalado no campo do remetente da mensagem.

- **X-Mailer:** software utilizado pelo *spammer* para o envio das mensagens. Apesar desse atributo poder ser facilmente manipulado pelo *spammer*, consideramos que o *spammer* varia esse atributo para cada mensagem enviada ou realiza apenas uma alteração no atributo e aplica para todas as mensagens. Para o primeiro cenário, esse atributo praticamente não tem relevância para o agrupamento realizado pela FP-Tree. No segundo cenário, esse atributo nos ajuda a identificar um padrão utilizado pelo *spammer*.

- **X-Helo:** o nome da máquina fornecido para envio. Apesar de também poder ser facilmente manipulado pelo *spammer*, justificamos o uso desse atributo utilizando o mesmo argumento dado para o uso do atributo X-Mailer.

- **Idioma:** para realizar a detecção do idioma, realizamos uma pré-processamento da mensagem removendo conteúdos HTML/CSS, e unimos o texto em mensagens *multi-part*. Por fim, aplicamos a biblioteca *langdetect* do Python[1] sobre o texto

[1]Langdetect: https://github.com/Mimino666/langdetect

gerado após o pré-processamento. Analisamos a exatidão do classificador na seção 3.3.

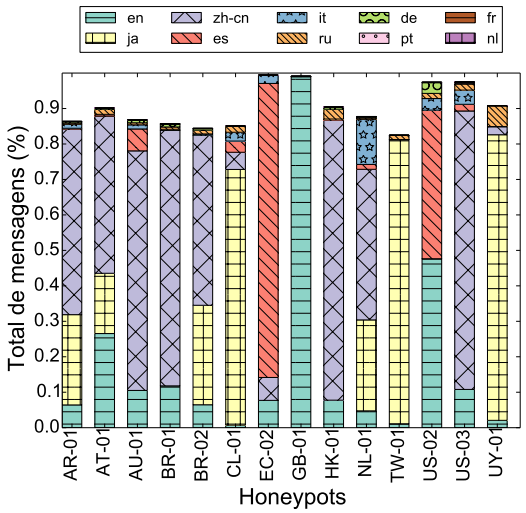
Esses atributos são representados por vértices na FP-Tree que possuem a propriedade de que um atributo em um vértice qualquer seja mais frequente que os atributos em seus descendentes, considerando todas as mensagens da base. Uma das formas para construir essa árvore é ordenar os atributos das mensagens do mais frequente para o menos frequente. Dessa forma, uma mensagem representada pelos atributos $\mathcal{A}_i, \mathcal{A}_{i+1}, \mathcal{A}_{i+2}$ tem a frequência de $\mathcal{A}_i$, considerando todas as mensagens da base, maior que a frequência de $\mathcal{A}_{i+1}$ e $\mathcal{A}_{i+2}$ e assim por diante. Formalmente, chamamos a representação de uma mensagem pela sequência de atributos ordenados pela sua frequência de *assinatura da mensagem*. Dessa forma, a construção da FP-Tree se torna trivial: a partir de um vértice raiz, que faz a ligação entre grupos de mensagens totalmente diferentes, basta adicionar os atributos ordenados de uma mensagem na árvore. Ao final desse processo, temos todas as mensagens compactadas em uma estrutura onde mensagens similares tendem a ter o mesmo caminho na árvore (exploramos esse fato para o algoritmo de identificação de campanhas apresentado na seção 3.4).

3.3 Detecção de idioma

Nós observamos que a detecção de idioma é bastante exata. Amostramos 200 campanhas aleatórias da base e, para cada campanha, amostramos uma mensagem de exemplo, também aleatoriamente. Verificamos que 95% das mensagens tiveram seu idioma identificado corretamente. As mensagens que não tiveram o idioma identificado corretamente, na amostra, foram devido a dois motivos principais: (i) problemas de codificação em mensagens em chinês que verificamos ser, aproximadamente, 3,8% de todas as mensagens da base e (ii) presença de termos aleatórios em mensagens em chinês que constatamos ser 0,2% das mensagens da base. Identificamos outros casos na amostra que se mostraram muito raros quando observamos sua frequência na base inteira: esses casos, somados, resultam em menos de 0,001% do total de mensagens. Observamos também que cerca de 7,1% das mensagens tiveram pouco ou nenhum texto e não foram classificadas com um idioma.

Decidimos manter todas as mensagens da base com os idiomas detectados pelo algoritmo de detecção de idiomas por duas razões principais. A primeira se baseia no fato que pouca ou nenhuma presença de texto nas mensagens (7,1% do total de mensagens) indicam outros tipos de mensagens utilizadas pelo *spammer* como, por exemplo, emails que possuem somente imagens ou anexos. Remover essas mensagens levaria a

perda de informação desse tipo de *spammer* e, por isso, decidimos manter essas mensagens na base com uma representação diferenciada das mensagens que tiveram um idioma associado pelo algoritmo de detecção de idiomas. A segunda razão está associada a especialização do método: queremos entender como as mensagens com problemas na detecção correta de seu idioma (4,0% do total de mensagens) se distribuem pelos *SpamBands* (discutido na seção 4.4).



(a) Por *honeypot*

IDIOMA	% MENSAGENS
Alemão	0,38
Chinês	39,16
Espanhol	2,74
Francês	0,03
Holandês	0,02
Inglês	10,66
Italiano	2,31
Japonês	31,07
Português	0,03
Russo	1,31

(b) Global

Figura 3.2: Distribuição dos 10 idiomas mais frequentes nas mensagens por *honeypot* e global

Dadas considerações sobre os problemas detectados na identificação dos idiomas, apresentamos agora uma visão geral dos idiomas nos *honeypots* utilizados. Escolhemos os dez idiomas mais frequentes na base para fazer a análise: alemão, chinês, espanhol, francês, holandês, inglês, italiano, japonês, português e russo. A figura 3.2a mostra a distribuição dos dez idiomas mais frequentes nas mensagens por *honeypot* e a tabela 3.2b mostra a distribuição global. É interessante observar a variabilidade dos *honeypots* em relação aos idiomas das mensagens que eles observam. Os *spammers* que conectam no *honeypot* da Grã-bretanha enviam mensagens com conteúdos em inglês em mais de 99% dos casos. Já o *honeypot* instalado no Equador observa que os *spammers* que se conectam a ele, em geral, mandam mensagens em espanhol e se distancia bastante dos outros *honeypots* instalados na américa latina (Brasil, Argentina, Chile e Uruguai) onde há um predomínio de mensagens em idiomas orientais (chinês e japonês). Um comportamento híbrido é encontrado em um dos *honeypots* instalado nos EUA (US-02), onde as mensagens enviadas se dividem, em sua maioria, em espanhol e inglês.

3.4 Algoritmo para geração de campanhas

O algoritmo proposto por Calais et al. [Calais et al., 2008] para identificar campanhas na FP-Tree procura por vértices nessa estrutura que tenham um aumento significativo no número de seus filhos, indicando a pontos de ofuscação de alguma característica como partes de URLs. Uma vez encontrado esse vértice, a campanha é detectada com assinatura $\mathcal{S}$, que é a sequência de atributos do vértice raiz até o ponto e ofuscação. As mensagens pertencentes a campanha são todas aquelas que possuem a assinatura da campanha $\mathcal{S}$ como prefixo de suas assinaturas. Em outras palavras, são todas as mensagens que são descendentes do vértice onde ocorreu o ponto de ofuscação.

É bastante claro que a detecção de campanha do algoritmo de Calais et al. é voltada para campanhas que utilizam algum tipo de ofuscação em seu conteúdo e a vantagem é que a técnica detecta essa ofuscação naturalmente, sem nenhum tipo de conhecimento anterior. Dois parâmetros são utilizados pelo algoritmo: a altura mínima da árvore e o número mínimo de filhos de um vértice para determinar um ponto de ofuscação. O problema que encontramos nessa técnica é a calibração dos valores desses parâmetros: o artigo não discute sobre quais valores foram utilizados para detecção. Nós verificamos que esses parâmetros são muito dependentes da estrutura da FP-Tree gerada. Além disso, é possível notar que se a árvore não possui nenhum ponto de ofuscação (indicativo de que o *spammer* não utiliza ofuscação no conteúdo), o algoritmo não detecta nenhuma campanha. Acreditamos que, apesar dos problemas observados na técnica de detecção de campanhas, a estrutura de dados utilizada para representar mensagens de *spam* é bastante poderosa pois consegue agrupar de uma maneira inteligente mensagens similares em relação aos atributos que possuem. Dessa forma, utilizamos a estrutura de dados (FP-Tree) e propomos um novo algoritmo de detecção de campanhas nessa estrutura.

A intuição para o algoritmo que propomos para detectar campanhas explora a propriedade da FP-Tree em agrupar mensagens de *spam*: dado um vértice na árvore, todas as mensagens em seus descendentes compartilham, pelo menos, os atributos do vértice raiz até o vértice dado. Observe que quanto maior o número de atributos compartilhados maior é a similaridade entre as mensagens. Dessa forma, para detectar campanhas de *spam* podemos buscar por grupos que compartilham mais atributos entre si, ou seja, mais similares (perto das folhas da árvore) e aumentar, gradativamente, a busca para grupos menos específicos em direção ao vértice da raiz que é comum a todas as mensagens da árvore. É necessário observar que existe um compromisso entre alta similaridade e o tamanho da campanha: a ofuscação observada no trabalho de Calais et al. [Calais et al., 2008] mostrou que a FP-Tree especializa demasiadamente

nas folhas, ou seja, vértices folhas possuem frequência muito baixa. Dessa forma, é necessário definir um número mínimo de mensagens a ser considerado para determinar uma campanha.

A presença do limiar mínimo de mensagens em uma campanha gera o problema em que mensagens podem não ser associadas a uma campanha. Entretanto, uma observação importante resolve esse problema: uma mensagem muito especializada pode ser generalizada removendo atributos menos frequentes de sua assinatura. Isso quer dizer que se um vértice na árvore não tem o número suficiente de mensagens para ser considerado uma campanha, podemos relaxar a assinatura dessas mensagens removendo esse atributo de cada assinatura. O caso base ocorre quando as mensagens chegam a ao vértice raiz, onde cada mensagem se torna uma campanha. O algoritmo 1 sintetiza o algoritmo proposto para a detecção de campanhas na FP-Tree utilizando o processo descrito.

Algorithm 1 Geração de campanhas

```
1:  function CAMPANHAS(fptree FP, raiz R, limiar_mensagens ℒ,caminho C)
2:      M := FP[R].mensagens();
3:      F := FP[R].filhos();
4:      C += FP[R].atributo();
5:      for vertice em FP do
6:          M += Campanhas(FP, vertice, L, C);
7:      if M.tamanho() > L then
8:          ImprimeCampanha(C,M);
9:          Retorna emptyset;
10:     else
11:         Retorna M;
```

O único parâmetro configurável do algoritmo é o número mínimo de mensagens que uma campanha deve possuir ($\mathcal{L}$). O algoritmo realiza uma travessia pós ordem na FP-Tree: se nenhuma campanha for detectada no retorno de um vértice, i.e., o número de mensagens for menor que o $\mathcal{L}$ então as mensagens pertencentes a ele são repassadas para o vértice pai, que passa a contar com mais mensagens. Observe que, neste caso, o que o algoritmo faz é relaxar o número de atributos da mensagem para associar cada mensagem a uma campanha, o que não acontece no algoritmo de Calais et al.. Interessante mencionar que campanhas com ofuscação são detectadas facilmente pelo algoritmo uma vez que Calais et al.. observou que a frequência de um atributo ofuscado é baixa, indicando poucas mensagens em cada ramo do ponto de ofuscação. Dessa forma, todas as mensagens são retornadas ao vértice ancestral (ponto de ofuscação) e a campanha é detectada assim como seria detectada pelo algoritmo de Calais et al. [Calais et al., 2008].

Ao final da execução obtemos o conjunto de assinaturas de campanhas, que é o caminho da árvore da raiz até o vértice onde houve a detecção. Se uma mensagem

não foi associada a nenhuma campanha, ou seja, chegou até a raíz da FP-Tree, então consideramos essa mensagem como uma campanha. Para fins de reprodutibilidade do nosso trabalho, nós utilizamos $\mathcal{L} = 1.000$ que é o número de mensagens da menor campanha encontrada por Calais et al. [Calais et al., 2008].

É importante citar que ambas as técnicas, tanto a que propomos quanto a de Calais et al., podem dividir uma campanha maior em campanhas menores. Na técnica de Calais et al. isso acontece quando uma campanha varia duas ou mais características $\mathcal{C}$, porém a variação gera um número de atributos $\mathcal{A}$ na característica mais frequente que é menor que o limiar mínimo do número de filhos para detectar pontos de ofuscação na FP-Tree. Dessa forma, enxergamos diversos pontos de ofuscação que, de fato, fazem parte da mesma campanha. No algoritmo que propomos, esse problema acontece se o número de mensagens de um atributo $\mathcal{A}$ na característica ofuscada $\mathcal{C}$ for maior que o valor de $\mathcal{L}$. Dessa forma, enxergamos diversas campanhas iguais que deveriam ter sido agrupadas em um vértice ancestral.

Nós aplicamos a técnica de detecção de campanhas por dia e por *honeypot* detectando, ao todo, 189.993 campanhas. Para avaliar a técnica, nós amostramos, aleatoriamente, cinco dias do período e, para cada dia, sorteamos um *honeypot* para avaliação, sem repetição entre os cinco dias considerados. Para avaliar o agrupamento de mensagens dentro da campanha, amostramos 20 mensagens aleatórias de até 10 campanhas para cada dia (limitados pelo número máximo de campanhas disponível no dia). O critério que utilizamos foi observar padrões de texto (idioma, conteúdo) e URLs presentes. Nós encontramos erro de classificação em apenas uma das 46 campanhas[2] amostradas, onde detectamos mensagens com objetivos diferentes. Isso mostra que nosso algoritmo é bastante preciso pois agrupa mensagens com objetivos similares em uma mesma campanha. Nós avaliamos a única campanha com erro de classificação e constatamos que que 40 das 698 mensagens dessa campanha deveriam formar uma campanha diferente.

Para avaliar se objetivos únicos foram associados a apenas uma campanha, ou seja, se campanhas deveriam ser agrupadas em uma campanha maior, nós selecionamos até 10 campanhas aleatórias para cada dia. Nós verificamos que 46 campanhas e encontramos 26 objetivos únicos, indicando que 20 dessas campanhas eram repetidas. Ou seja, verificamos que nossa técnica é pouco exata pois detecta partes da campanha dependendo do parâmetro $\mathcal{L}$ utilizado, conforme mencionado anteriormente. Entretanto, aplicando a técnica dos *SpamBands*, que iremos discutir no capítulo 4, observamos que as campanhas associadas ao mesmo objetivo foram agrupadas no mesmo *SpamBand*.

[2]Um *honeypot* tinha apenas 6 campanhas.

3.5 Exemplo do algoritmo de campanhas aplicado na FP-Tree

A figura 3.3 ilustra uma FP-Tree construída a partir de 10 mensagens fictícias. A representação de uma mensagem indica o último atributo daquela mensagem inserido na árvore. Podemos observar que 4 mensagens possuem exatamente os atributos *Idioma=en,Assunto=apple* e que o atributo *Idioma=en* é o mais frequente na base. O contrário também pode ser observado: não existem mensagens formadas apenas pelos atributos *Idioma=en,Layout="TTUBB"* onde "TTUBB" significa a sequência de duas linhas de texto, uma URL e duas linhas em branco na mensagem. Além disso, pode-se observar que existem mensagens cujos atributos são um subconjunto de outras mensagens como é o caso das mensagens com atributos *Idioma=en,Assunto=apple,URL=apple.com* e *Idioma=en,Assunto=apple*. É possível observar que mensagens podem não estar necessariamente nos atributos folhas e, por isso, representamos-as explicitamente.

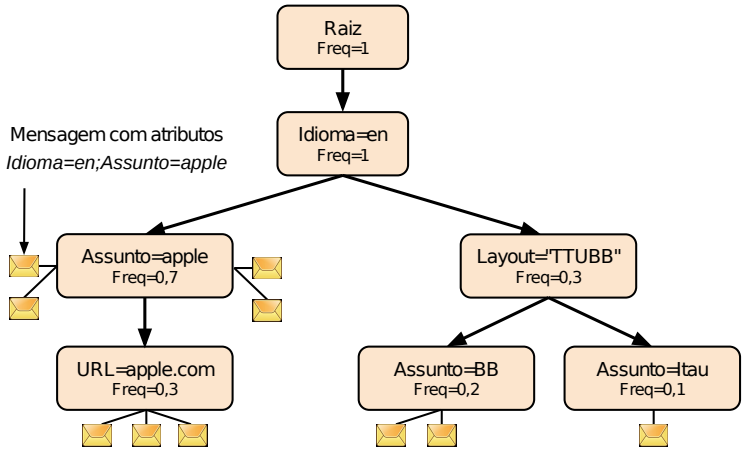
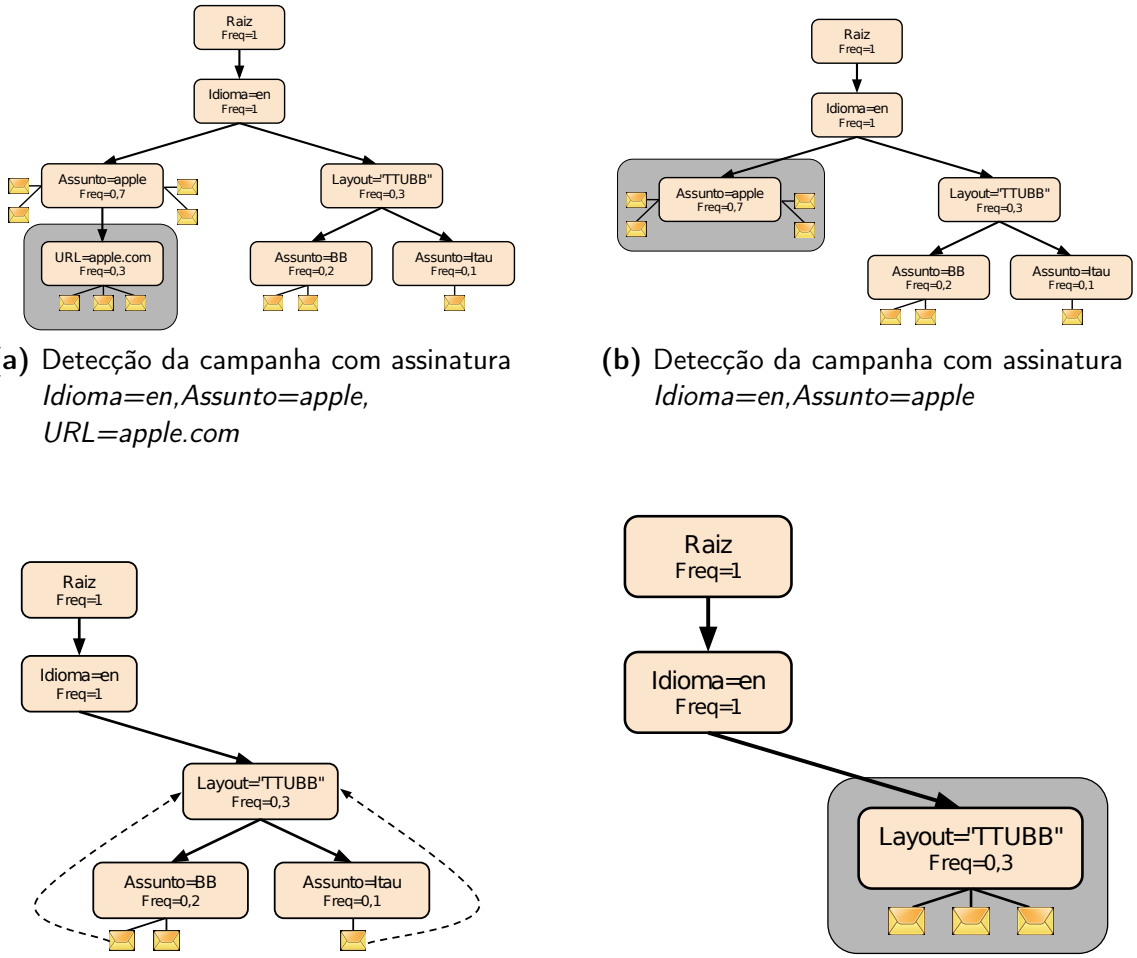


Figura 3.3: Ilustração da FP-Tree.

A figura 3.4 ilustra passo a passo como é feita a detecção de campanhas na árvore da figura 3.3 com $\mathcal{L} = 3$. Inicialmente, a busca pós-ordem identifica que o vértice *URL=apple.com* obedece o *limiar_mensagens* imposto e detecta a primeira campanha com assinatura *Idioma=en,Assunto=apple,URL=apple.com* (figura 3.4a). Como a campanha foi detectada, nenhuma mensagem foi repassada para o nó ancestral do vértice *URL=apple.com*. A segunda campanha é detectada com assinatura *Idioma=en,Assunto=apple* (figura 3.4b). Observe que as duas campanhas detectadas possuem o tópico (apple) porém uma delas tem URLs e a outra não, ou seja, são

campanhas diferentes pela nossa definição. A continuação da busca pós-ordem detecta que os vértices *Assunto=BB* e *Assunto=Itau* não possuem mensagens suficientes para integrar uma campanha e, dessa forma, uma relaxação é feita: as mensagens desses vértices são repassadas para o vértice *Layout="TTUBB"* (figura 3.4c). Por fim, a busca detecta a campanha *Idioma=en,Layout="TTUBB"* com três mensagens.



(a) Detecção da campanha com assinatura *Idioma=en,Assunto=apple, URL=apple.com*

(b) Detecção da campanha com assinatura *Idioma=en,Assunto=apple*

(c) Repasse de mensagens para um vértice ancestral

(d) Detecção da campanha com assinatura *Idioma=en,Layout="TTUBB"*.

Figura 3.4: Ilustração do algoritmo de geração de campanhas aplicado em uma FP-Tree.

3.6 Identificação de phishing em campanhas

Nesta seção, mostramos o método utilizado para identificar se uma campanha tem elementos de *phishing* ou propaganda associados, que são os dois objetivos mais comuns

dos *spammers* e amplamente discutidos na literatura. É importante destacar que o problema que estamos buscando resolver é a separação entre *spam* de propaganda e *phishing* em uma base composta apenas por *spam*. Não é nosso objetivo a criação de uma nova técnica para detecção de *phishing* em bases com emails legítimos.

A premissa básica de uma campanha é que esta é composta por mensagens similares usadas para um propósito comum. Essa premissa é a base dos métodos utilizados para identificação de campanhas e indica que a utilização de uma única mensagem é suficiente para generalizar o resultado para toda a campanha. Dessa forma, para cada campanha no período avaliado, amostramos uma mensagem como sua representante e aplicamos o resultado obtido para essa mensagem a todas as mensagens da campanha.

Para classificar as mensagens representantes de cada campanha em uma dessas duas classes, utilizamos o classificador Naive Bayes que é um algoritmo supervisionado bastante utilizado para classificar textos [Mitchell, 1997; Shimodaira, 2014]. Inicialmente, esse classificador recebe um conjunto de treinamento que consiste em textos e suas respectivas classes. Nós definimos nosso conjunto de treinamento a partir de uma amostra das mensagens representantes das campanhas identificadas em oito *honeypots* durante o período avaliado. Ao todo, utilizamos 107 mensagens de campanhas diferentes, distribuídas entre propaganda (36 mensagens) e *phishing* (71 mensagens). Nós observamos que mensagens de propaganda são mais comuns em japonês e chinês enquanto *phishing* possuem uma variação maior entre os idiomas. Também buscamos balancear os conjuntos de treino associando pelo menos uma mensagem de cada um dos dez idiomas mais frequentes a cada uma das classes. Além disso, submetemos cada mensagem a um conjunto de regras para torná-las menos poluídas e mais padronizadas, evitando que diferenças sutis como letras maiúsculas e minúsculas afetem o modelo. Mais especificamente, convertemos cada carácter da mensagem para sua representação minúscula em seu idioma (quando houver) e removemos links, números, pontuações e *stopwords* de todos os dez idiomas mais frequentes. A figura 3.5 ilustra a configuração inicial do classificador.

O passo seguinte do algoritmo é a extração de um conjunto de atributos de cada documento. A definição do que é um atributo é bastante variável entre diversas implementações do Naive Bayes. Em nossa implementação, consideramos como atributo a presença ou não de um termo no documento, sem observar o número de ocorrências. Nós baseamos essa escolha em resultados de outros trabalhos que mostram que a presença de certos termos em uma mensagem podem ser usados para definir se uma mensagem é *phishing* ou não [Las-Casas et al., 2016]. Uma vez que os termos tenham sido extraídos do documento, nós associamos cada um deles à classe do documento, reservando apenas uma ocorrência de um termo por documento, conforme mostrado

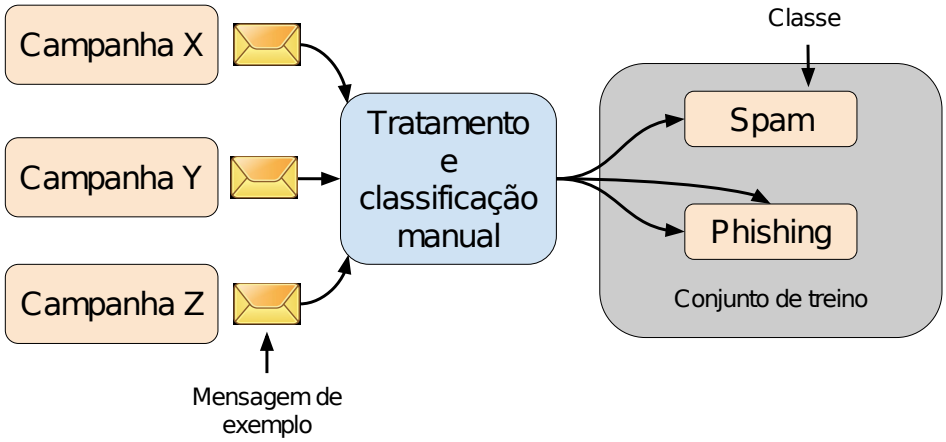


Figura 3.5: Treinamento do classificador bayesiano

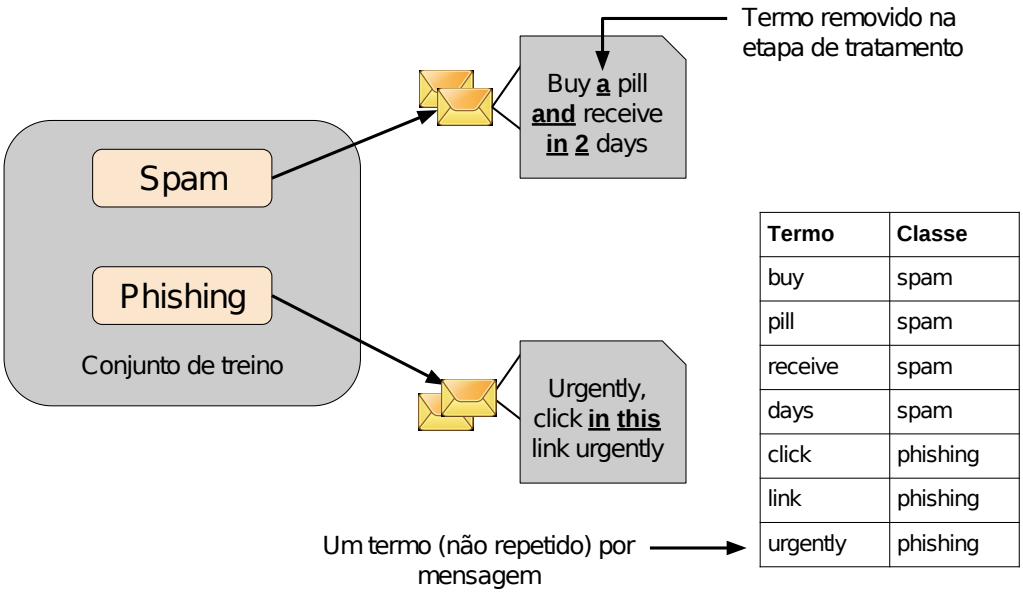


Figura 3.6: Seleção de atributos utilizado para o algoritmo Naive Bayes

na figura 3.6.

Dada seleção de atributos, calculamos a probabilidade de encontrar um atributo específico dado um documento. Esse cálculo é feito observando a frequência do atributo entre todos os documentos de uma classe. Formalmente, se a é um atributo e c uma classe, estamos calculando $P(a|c)$. Além disso, calculamos a representatividade $P(c)$ de uma classe entre todos os documentos. Essas probabilidades, exemplificadas pela figura 3.7, são a base para a classificação de novos textos. Isso porque o Naive Bayes busca associar uma probabilidade de um novo documento pertencer a cada classe dado o conjunto de atributos que são extraídos (ignorando aqueles que não aparecem no conjunto de treino).

Atributo *a*	P(*a*|spam)	P(*a*|phishing)
click	0,1	0,8
pill	0,9	0,0
please	0,4	0,9
buy	0,9	0,2
attachment	0,05	0,6
...	...	...

Classe	P(classe)
spam	0,4
phishing	0,6

P(Classe *c*) = #documentos em *c* / #total de documentos

P(Atributo *a*|Classe *c*) = #documentos em *c* que contêm *a* / #documentos em *c*

Figura 3.7: Ilustração do cálculo das probabilidades do modelo bayesiano

Formalmente, dado um novo documento com um conjunto de atributos A e uma classe c, a probabilidade do documento pertencer à classe c é dado pelo teorema de Bayes:

$$P(c|A) = \frac{P(A|c) \times P(c)}{P(A)}$$

Como o Naive Bayes considera independência entre atributos, ou seja, a ausência ou presença de um atributo não implica na ocorrência ou ausência de outros atributos, essa equação pode ser reescrita conforme se segue:

$$P(c|a_1, a_2, ..., a_n) = \frac{P(a_1|c) \times P(a_2|c) \times ... \times P(a_n|c) \times P(c)}{P(A)}$$

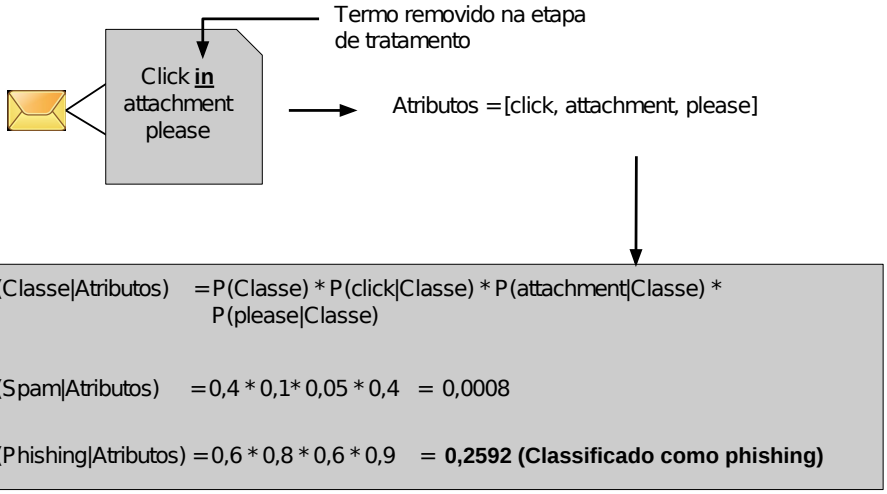


Figura 3.8: Exemplo de aplicação do classificador a um novo documento

Em nosso trabalho, utilizamos a classe que resultou em maior probabilidade para

classificar o documento. É importante notar que o termo $P(A)$ é o mesmo para o cálculo de todas as classes e, dessa forma, seu cálculo não é necessário: basta comparar qual classe tem o maior numerador. A figura 3.8 ilustra como é realizada a classificação de um novo documento.

Para testar o modelo gerado pelo conjunto de treino, foram sorteadas campanhas aleatórias dos seis *honeypots* que não foram utilizados no treinamento. Envolvemos no teste uma mensagem representante de campanhas de todos os dez idiomas mais frequentes nas mensagens, totalizando 54 mensagens. Cada mensagem do conjunto de teste foi tratada exatamente como as mensagens do conjunto de treino removendo pontuações, números, *stopwords* e convertendo todos os caracteres para sua representação minúscula (quando houver). Nós classificamos manualmente cada uma dessas mensagens e verificamos que a taxa de acerto do algoritmo de classificação foi de 92,59%, sugerindo que o classificador consegue distinguir bem *spam* de propaganda de mensagens *phishing*. Para o restante dessa dissertação, usamos o modelo descrito nessa seção para classificar campanhas como *spam* de propaganda ou *phishing*.

Capítulo 4

SpamBands

O conceito de campanhas de *spam*, embora útil para detectar e agrupar os diferentes tipos de mensagens de spam possibilitando a identificação de padrões nas mensagens agrupadas, é incompleto para identificar a infraestrutura utilizada pelo *spammer*, pois partes de mesma estrutura podem enviar diferentes campanhas de *spam*. Para identificar essa infraestrutura podemos observar como as campanhas se sobrepõem entre endereços IP. Essa observação é a base dos *SpamBands* (definidos na seção 4.1) que buscam identificar infraestruturas orquestradas para o envio de *spam*.

Utilizando os *SpamBands* detectados, avaliamos as campanhas de *spam* da base de dados utilizada e detectados pontos chave que nos ajudam a entender o comportamento do *spammer*: (i) identificamos grupos de *spammers* que exploram os *honeypots* tanto como proxy quanto relay abertos, o que sugere a existência de grupos que utilizam tanto servidores dedicados quanto *bots* para o envio de *spam* (seção 4.3); (ii) identificamos que o *spammer* busca maximizar sua chance de sucesso na entrega de um email enviando mensagens com idiomas relacionados ao ccTLD do endereço de destino e que *SpamBands* que exploram os *honeypots* apenas como proxy aberto (HTTP/SOCKS) tendem a estar mais relacionados com idiomas orientais enquanto *SpamBands* que exploram os *honeypots* como relay aberto (SMTP) estão relacionados a idiomas ocidentais ou ao idioma chinês (seção 4.5); (iii) identificamos que *SpamBands* HTTP/SOCKS estão mais associados ao envio de propaganda enquanto *SpamBands* de *phishing* estão mais ligados ao protocolo SMTP (seção 4.6).

4.1 Identificação dos SpamBands

A base dos *SpamBands* é a premissa de que máquinas que enviam mensagens pertencentes às mesmas campanhas são controladas por um mesmo agente orquestrador estando,

assim, relacionadas a uma mesma origem. A relação entre máquinas e campanhas pode ser modelada como um grafo G, onde as máquinas são vértices e há uma aresta entre duas máquinas se elas enviaram mensagens associadas a uma mesma campanha. A figura 4.1 ilustra a construção desse grafo.

A partir do grafo G, um *SpamBand* pode ser identificado como um sub-grafo denso (diversas origens que compartilham um mesmo conjunto de campanhas). A identificação desses subgrafos pode ser obtida aplicando-se algoritmos de agrupamento de grafos; entretanto, tais algoritmos tendem a ser bastante complexos e difíceis de calibrar [Almeida et al., 2011]. Com base nas características particulares do problema em questão onde observamos componentes com subgrafos densos conectados apenas por poucos vértices, adotamos uma estratégia mais simples e interativa buscando remover apenas esses poucos vértices conforme descrito a seguir.

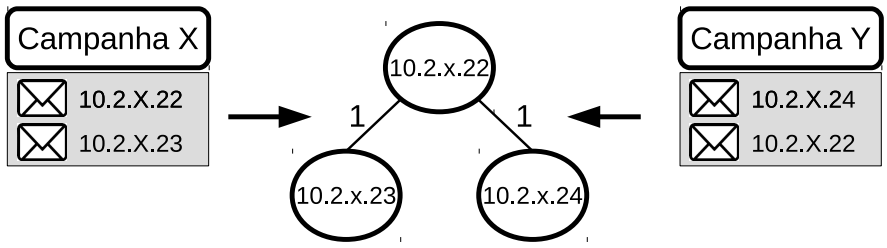


Figura 4.1: Geração do grafo de relações entre endereços IP.

Inicialmente, cada componente conectado de G poderia ser identificado como um *SpamBand*. Entretanto, aspectos práticos exigem que essa definição seja refinada. Por exemplo, quando um endereço IP pode se referir a diferentes máquinas atrás de um mecanismo de NAT ou ao longo do tempo devido à alocação de endereços feita por um serviço DHCP, duas máquinas podem estar atuando sob coordenações diferentes mas serem vistas no resto da rede com um mesmo endereço de origem. Os vértices referentes a esses endereços IP aparecem no grafo como nós de ligação entre sub-grafos mais densos que, na prática, se referem a *SpamBands* diferentes.

A forma adotada para identificar esses casos e isolar os *SpamBands* envolvidos foi utilizar o conceito de *betweenness*, que mede o grau de centralidade de nós em um grafo. Essa métrica quantifica o número de caminhos mínimos entre todos os pares de nós no grafo que passam por um vértice em questão. A premissa é que, se alguns vértices possuem um valor de *betweenness* muito elevado em relação ao que seria esperado para um grafo fortemente conectado, existe uma chance maior desses vértices conectarem dois sub-grafos internamente mais densos. Assim, se removemos esses vértices, acentuamos a separação entre os sub-grafos densos desejados.

A determinação de *SpamBands* é então apresentada no algoritmo 2, que recebe três parâmetros de entrada: o grafo (**G**), o limiar de *betweenness* mínimo a ser considerado (**limiar_bt**) e o número máximo de endereços IP (vértices) que podem ser removidos para dividir um componente (**limiar_ips**). O primeiro passo determina os componentes conectados de G, que constituem uma primeira aproximação dos *SpamBands*. A seguir, identificamos sub-grafos densos em cada componente conectado removendo nós com betweenness acima de um **limiar_bt** do maior betweenness encontrado, respeitando o limite **limiar_ips** que define o número máximo que nós que podem ser removidos evitando retirar um grande número de endereços IP do grafo e, dessa forma, perder propriedades importantes do grafo associado a esses endereços como AS, Country Codes, prefixos, etc.. O algoritmo retorna o conjunto S que contém todos os *SpamBands*.

Algorithm 2 Geração de *SpamBands*

```
1:  function SpamBands(G,limiar_bt,limiar_ips)
2:      S := ∅;
3:      C := G.componentes_conectados();
4:      for comp em C do
5:          ips_a_remover := ∅;
6:          for ip em comp do
7:              if ip.betweenness() < limiar_bt × comp.maior_betweenness() then
8:                  ips_a_remover.Adiciona(ip);
9:          if ips_a_remover.tamanho() > limiar_ips × comp.num_vertices() then
10:             S += comp;
11:         else
12:             S += comp.remove_vertices(ips_a_remover);
13:     Retorna S;
```

4.2 Avaliação dos SpamBands em dados reais

Nesta seção, aplicamos a metodologia descrita anteriormente em dados reais de um *honeypot* para um dia. Inicialmente, geramos o grafo utilizando a técnica ilustrada na figura 4.1: conectamos um par de endereços IP se eles compartilham uma campanha em comum. O grafo da figura 4.2 ilustra a primeira fase da técnica onde fica evidente que esse grafo possui três comunidades distintas. O algoritmo 2 realiza essa detecção, ilustrado pela figura 4.3, utilizando valores de 0,1 e 0,2 para o *limiar_bt* e *limiar_ips*. Nós utilizamos esses valores para a geração de todos os *SpamBands* desse trabalho. É um valor empírico baseado em trabalhos anteriores que assume o compromisso entre maximizar a detecção de comunidades (grupos bem conectados entre si) sem perdas significativas de informação (sem a remoção de um número significativo de vértices).

Inicialmente, o algoritmo 2 identifica o vértice preto que liga os três *SpamBands* como sendo o que tem maior betweenness (35% dos menores caminhos no grafo passam

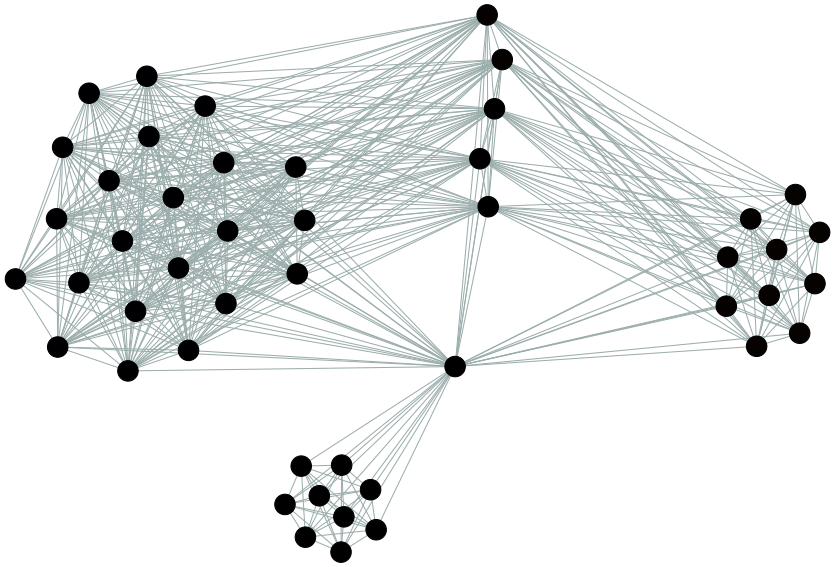


Figura 4.2: Grafo de relações entre IP sem utilizar o algoritmo de detecção de *SpamBands*

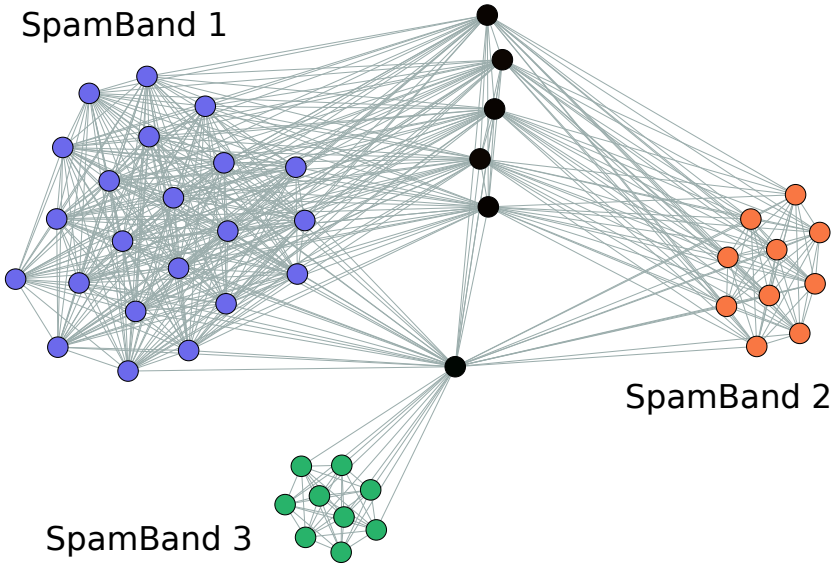


Figura 4.3: Detecção dos *SpamBands* utilizando o algoritmo de detecção de *SpamBands*

por ele). Imediatamente, o algoritmo detecta outros cinco vértices pretos que estão entre os *SpamBands* 1 e 2 utilizando o valor do limiar de betweenness *limiar_bt* (por cada vértice passam 3,5% dos menores caminhos no grafo). Nosso algoritmo realiza a confirmação de que o número de vértices a serem removidos respeita a restrição imposta pelo parâmetro *limiar_ips* e remove todos os seis vértices pretos, gerando os

três *SpamBands* mostrados na figura 4.3.

A seguir, fazemos uma análise individual de cada *SpamBand* mostrando a capacidade da técnica em fazer a distinções a nível de rede utilizando, a princípio, atributos de conteúdo para gerar campanhas e um atributo de rede (endereço IP) para a geração do grafo. As informações de cada um estão sumarizadas na tabela 4.1.

***SpamBand* 1:** Nós observamos que o *spam*Band 1 é composto por 21 endereços IP alocados para Taiwan e que enviaram 29.902 mensagens com tamanho médio de 1,9KB. Esses endereços IP pertencem ao AS 3462 e todos exploraram o *honeypot* como proxy aberto através do uso do protocolo HTTP. Além disso, esses endereços IP estão classificados na PBL[1]. O conteúdo das mensagens pertencentes às campanhas desse *SpamBand* é em chinês e tem o objetivo claro de fazer propaganda de produtos.

***SpamBand* 2:** Este *spam*Band, composto por 10 endereços IP, é diferente do *SpamBand* anterior em quesitos interessantes: (i) os endereços IP estão associados ao *country code* do Japão; (ii) os endereços estão distribuídos em quatro ASes (2527, 2519, 2497 e 4713); (iii) menos da metade dos endereços estão na PBL; (iv) o tamanho médio das mensagens é quase o dobro e muito variável em relação ao *SpamBand* 1; (v) as mensagens das campanhas estão em japonês e (vi) o número de mensagens enviadas é muito superior mesmo com o número de endereços IP inferior. É importante ressaltar o poder de separação da técnica pelo item (ii): nenhum dos quatro ASes é o AS 3462 onde está localizado o *SpamBand* 1, ou seja, a técnica é capaz de separar grupos distintos na rede apenas com atributos de conteúdo e o endereço IP.

***SpamBand* 3:** A primeira diferença desse *spam*Band visível na tabela 4.1 é que todos os endereços IP desse *SpamBand* estão localizados em Taiwan e nenhum deles está listado em *blacklists*. Ainda mais, o AS 38478 não coincide com nenhum AS dos *SpamBands* 1 e 2. Por fim, observamos que as mensagens enviadas por esse *SpamBand* também são em japonês, ou seja, algum *spammer* japonês utiliza a rede de Hong Kong para fazer a entrega das mensagens no Japão.

4.3 Determinando infraestruturas pelos protocolos nos SpamBands

A figura 4.4 mostra a distribuição dos 9,652 *SpamBands* encontrados por dia e por *honeypot* no período avaliado. Pode ser observado que o número absoluto não difere

[1]PBL: abreviação para Policy *blacklist*. Essa *blacklist* indica endereços IP de usuários finais que estão enviando mensagens de email para servidores SMTP que não são os oferecidos pelo seu ISP.

Tabela 4.1: Atributos dos *SpamBands* da figura 4.3.

ATRIBUTO	SPAMBAND 1	SPAMBAND 2	SPAMBAND 3
Número de Mensagens	29.902	161.040	119.761
Número de IP	21	10	9
Protocolos	HTTP	SOCKS	SOCKS
Country codes	TW	JP	HK
blacklist PBL (# IP)	21	4	0
Tamanho médio (KB)	1,9±0,1	3,9±2,6	2,9±1,1
ASes	3462	2527, 2519, 2497, 4713	38478
Idioma das mensagens	chinês	japonês	japonês

muito de um *honeypot* para outro. A maior diferença observada entre um *honeypot* e outro é de 24 *SpamBands* (maior valor subtraído pelo menor valor considerando o desvio padrão).

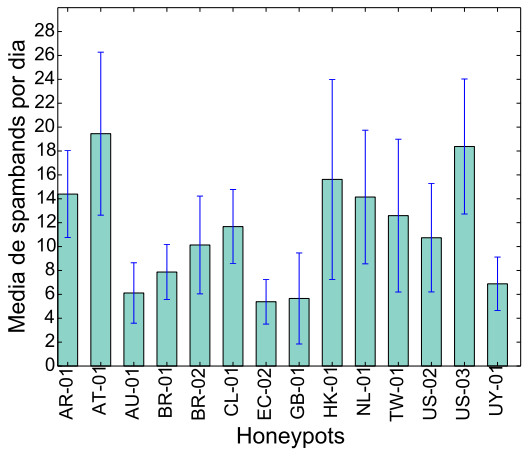
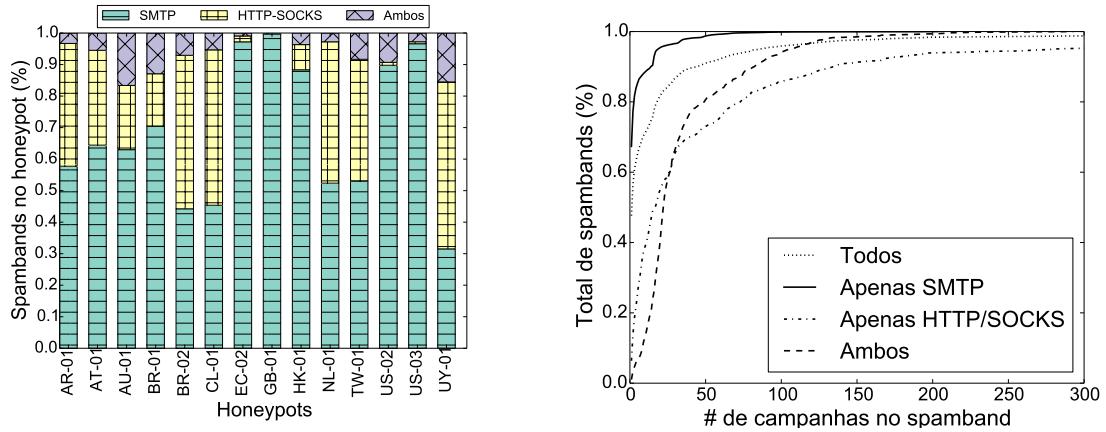


Figura 4.4: Distribuição dos *SpamBands* por dia e por *honeypot*

A figura 4.5a oferece uma visão sobre os *SpamBands* e os protocolos por eles utilizados: se pelo menos um endereço IP de um *SpamBand* utiliza um protocolo, agregamos esse protocolo à lista de protocolos utilizados pelo *SpamBand*. Além disso, agrupamos os protocolos HTTP e SOCKS no grupo HTTP/SOCKS uma vez que estamos interessados em observar o *SpamBand* do ponto de vista de abuso e ambos estão associados ao abuso do *honeypot* como proxy. É interessante observar a separação dos *SpamBands* em relação aos protocolos sem utilizar essa informação para sua constru-

ção. Analisando todos os *SpamBands* encontramos que apenas 26% deles possuem apenas endereços IP que utilizam HTTP/SOCKS enquanto 68% dos *SpamBands* utilizam apenas o protocolo SMTP. Conforme informado no capítulo 3.1, grande parte das mensagens coletadas utilizaram o protocolo HTTP/SOCKS porém vemos poucos grupos enviando essas mensagens. O oposto ocorre para o protocolo SMTP, que possuem muitos grupos que enviam poucas mensagens. Em nosso entendimento, essa observação mostra que *SpamBands* HTTP/SOCKS podem indicar grupos de servidores utilizados por *spammers* para entregar suas mensagens enquanto *SpamBands* SMTP indicam grupos de máquinas infectadas com *malware* para o envio de campanhas de *spam*. Os *SpamBands* que possuem endereços IP que se distribuem entre SMTP e HTTP/SOCKS (6% dos *SpamBands*) indicam grupos de *spammers* que buscam os dois meios para entregar suas mensagens. Exemplificamos um *SpamBand* híbrido na seção 4.3.1.



(a) Distribuição dos *SpamBands* por protocolo e por *honeypot*

(b) Número de campanhas por *SpamBand* e por protocolo

Figura 4.5: Visão geral dos *SpamBands* em relação aos protocolos.

O gráfico da figura 4.5b mostra o número de campanhas por *spam*Band por protocolo. No geral, 47% dos *SpamBands* possuem apenas uma campanha e 90% possuem 50 campanhas ou menos. Esse fato mostra que se fizéssemos uso de campanhas ao invés de *SpamBands*, deixaríamos de detectar infraestruturas mais completas. Por exemplo, um *SpamBand* composto por dez campanhas seria dividido em dez grupos de endereços IP que se sobrepõem entre si tornando a detecção desses grupos mais fracionadas. A análise por protocolos mostra que 68% dos *SpamBands* compostos apenas de endereços IP que enviam mensagens através do protocolo SMTP possuem uma campanha enquanto essa porcentagem é bem menor (6%) em *SpamBands* compostos apenas de endereços IP que enviam HTTP/SOCKS. Esse fato sugere que *SpamBands* SMTP es-

tão mais relacionados a *botnets* uma vez que o *spammer* deve enviar poucas mensagens por máquina de usuário para não ser detectado e *SpamBands* HTTP/SOCKS podem indicar uma infraestrutura própria do *spammer* dado que o número de campanhas observadas é maior. Também pode-se observar que *SpamBands* com ambos os protocolos tendem a ter um comportamento similar ao dos *SpamBands* HTTP/SOCKS e possuem um número maior de campanhas em relação a *SpamBands* SMTP. Nós investigamos esse comportamento nesses *SpamBands* e verificamos que em mais de 99% dos casos a maior parte das campanhas são enviadas pelos endereços IP que utilizam os protocolos HTTP/SOCKS.

4.3.1 SpamBand híbrido

Nesta seção, descrevemos um *SpamBand* híbrido encontrado no *honeypot* localizado nos EUA (US-03) no dia 1 de abril de 2016 a fim de exemplificar *spammers* que possam estar usando de servidores dedicados e máquinas infectadas para enviar suas mensagens. Observamos que apenas um endereço IP explora o *honeypot* como proxy aberto para o envio das mensagens que também são enviadas por outros endereços IP que utilizam o protocolo SMTP para envio. Investigamos esse endereço e descobrimos que ele é proveniente do AS 8615 (CNT-AS OJSC Central telegraph) instalado na Rússia e o destino da conexão SOCKS foi uma máquina no AS 17090 (Database by Design) localizado nos EUA. Encontramos ainda que esse endereço IP está listado apenas na PBL, o que indica ser máquina de usuário final dedicada ao envio e possivelmente pertencente ao *spammer*. Entretanto, não encontramos outras evidências que nos permitissem confirmar essa informação.

A figura 4.6 mostra dois exemplos de mensagens enviadas pela maioria dos endereços IP do *SpamBand* em questão, incluindo o endereço IP que utiliza SOCKS para o envio. A mensagem em alemão oferece um tipo de serviço e busca uma confiabilidade do usuário informando nomes de antivírus mundialmente populares para acessar o serviço. Observamos diversas mensagens em alemão com a mesma estrutura, modificando o nome dos antivírus (Avira, Kaspersky) e o tipo de produto anunciado. Acreditamos que essas mensagens são propagandas de produtos e a menção de antivírus é para ganhar a confiança do usuário para checar a propaganda. O algoritmo de detecção de *phishing* da seção 3.6 também classifica essas mensagens como *spam*. Analisamos a mensagem em russo e constatamos também ser propaganda de serviços aparentemente ilegais para empresas. Essa foi uma mensagem de treino para o algoritmo da seção 3.6 e classificamos-a como sendo *spam* de propaganda. Nós observamos esse comportamento nas outras 21 campanhas de *spam* que compõem o *SpamBand*.

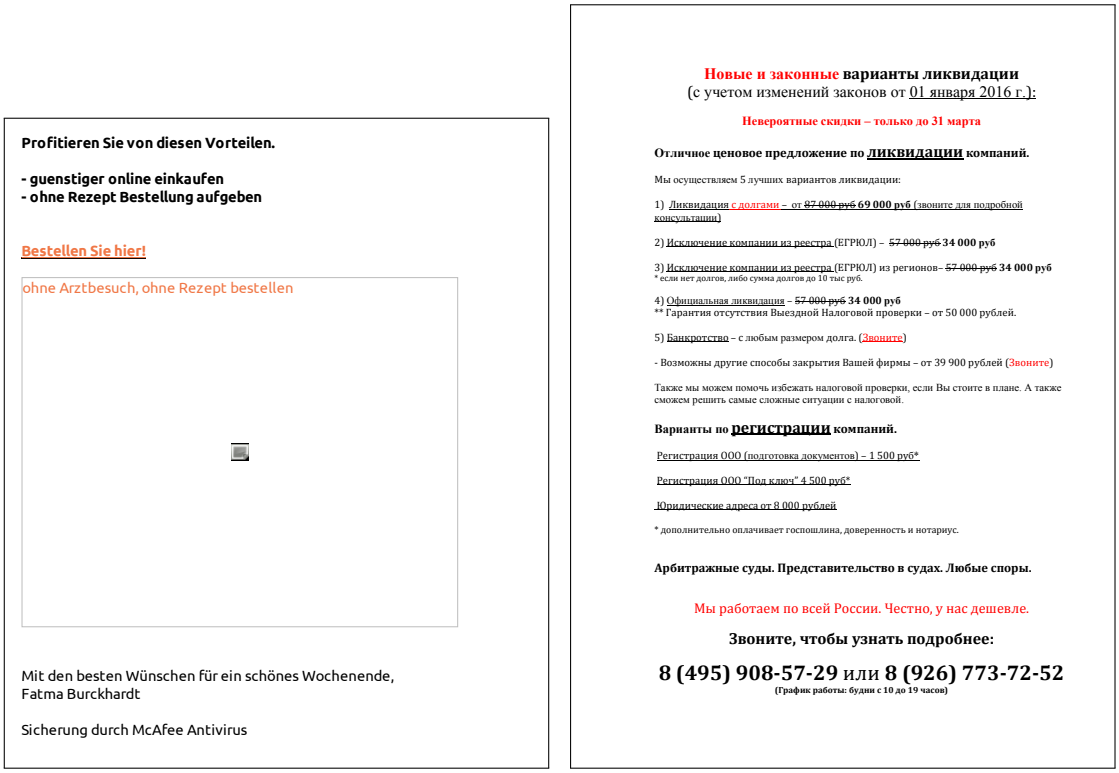


Figura 4.6: Exemplo de mensagens de campanhas em russo e alemão.

Tabela 4.2: Atributos do *SpamBand* com propagandas russas e alemãs.

Atributo	Valor
# mensagens	37.176
# endereços IP distintos	17
# número de campanhas	24
ASes	55053, 12714, 196689, 50297, 45899, 38753, 12880, 41164, 25454, 197343, 43793, 59988, 8615, 8400, 28192, 9044, 3239
Country Codes	MD, FR, CH, RU, RS, CA, IR, VN, CZ, BR, NO, UA, ID
Protocolos (# IP)	SOCKS (1), SMTP (16)
blacklist (# IPS)	PBL (6), XBL (15)
Idioma das mensagens	russo, alemão
Coef. Agrupamento	0,96

É interessante observar pela tabela 4.2 que, apesar dos 17 endereços IP compartilharem mensagens entre si, eles estão distribuídos ao redor do mundo e em diferentes

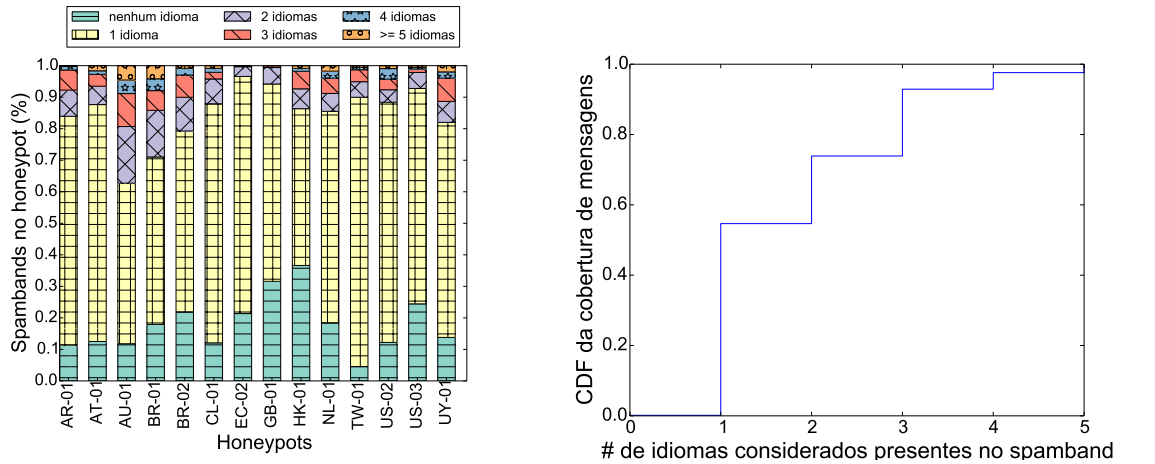
sistemas autônomos. Além disso, a maioria dos endereços IP estão presentes na XBL. Esses fatos sugerem que os endereços IP desse *SpamBand* (exceto o IP que envia mensagens através do protocolo SOCKS) podem fazer parte de uma *botnet* que oferece serviços para *spammers*, fato que explica o porque do uso de dois idiomas e tipos diferentes de mensagens.

4.4 Avaliação dos idiomas nos SpamBands

Analisamos os 9,652 *SpamBands* encontrados no período a fim de entender como as mensagens com idiomas não classificados ou identificados incorretamente foram endereçados pela técnica de geração dos *SpamBands*. Observamos que cerca de 10% dos *SpamBands* foram formados apenas por campanhas cujo idioma das mensagens não foi identificado e que as mensagens nestes *SpamBands* representam 0,6% do total de mensagens. Identificamos que mais de 99% desses *SpamBands* são compostos apenas por endereços IP que utilizam os protocolos HTTP/SOCKS. O restante das campanhas com idiomas não identificados (6,4% do total de mensagens) foram distribuídas por cerca de 20% dos *SpamBands* não sendo, em nenhum caso, a maioria das campanhas dentro do *SpamBand*.

As mensagens em chinês com conteúdo aleatório (estudo de caso da seção 4.4.1) ou com problemas de codificação foram detectadas, em sua totalidade, em *SpamBands* compostos apenas por endereços IP que utilizam HTTP/SOCKS ou híbridos. Verificamos ainda que esses *SpamBands*, em 92% dos casos, contém campanhas detectas em chinês pelo algoritmo de detecção o que mostra que a maioria das campanhas em chinês com problemas de codificação na mensagem foram agrupadas corretamente. Observamos ainda que muitos idiomas ocidentais considerados (português, espanhol, inglês, alemão, francês, holandês e italiano) em *SpamBands* HTTP/SOCKS foram detectados devido aos problemas encontrados com as mensagens em chinês. Solucionamos esse problema substituindo esses idiomas no exemplo de mensagem das campanhas de *SpamBands* HTTP/SOCKS. Observamos ainda que os *SpamBands* compostos por apenas por campanhas com idioma não identificado possuem apenas endereços IP que utilizam o protocolo SMTP, não necessitando de correções.

A figura 4.7a mostra o número de idiomas entre os 10 mais frequentes dentro de cada *SpamBand* em cada *honeypot*. É possível observar a existência de *SpamBands*, por dia e por *honeypot*, que não possuem nenhum idioma dentre os considerados (17% do total). Nós constatamos que grande parte desses *SpamBands* (97%) são *SpamBands* formados apenas por um endereço IP que utiliza o protocolo SMTP. A figura 4.7b



(a) Número de idiomas, dentre os 10 considerados, nos *SpamBands* por *honeypot*

(b) Cobertura do total de mensagens dos *SpamBands* por número dos fez idiomas mais frequentes

Figura 4.7: Relação entre *SpamBands* e idiomas

mostra que esses *SpamBands* possuem pouca representatividade na base: a soma de todas as mensagens enviadas por eles resulta em menos de 1% das mensagens. Nós avaliamos o conteúdo das mensagens representantes das campanhas nesses *SpamBands* e encontramos mensagens sem nenhum texto (64%), outras com conteúdo sem semântica (35%) e muito poucas (1%) com outros idiomas muito pouco representativos em nossa base como dinamarquês e romeno. Isso indica que esses *SpamBands* podem ser pequenos *spammers* que exploram os *honeypots* para enviar mensagens específicas.

Entre os grupos com um ou mais idiomas dentre os dez considerados, observamos que eles representam cerca de 99% das mensagens com concentração de um a três idiomas (90% do total das mensagens). Comparamos os *SpamBands* que possuem apenas um idioma (67% do total) com os *SpamBands* que possuem mais de um idioma (16% do total) e encontramos que ambos possuem o mesmo comportamento: a maioria utiliza apenas SMTP. Uma observação interessante se refere a *SpamBands* híbridos onde 92% desses *SpamBands* possuem mais de um idioma. Esse fato mostra a existência de grupos especializados para o envio com o uso de *botnets* e servidores dedicados. Nós apresentamos um estudo de caso de um *SpamBand* com cinco idiomas na seção 4.4.2.

4.4.1 SpamBand chinês com mensagens sem texto

Nesta seção discutimos brevemente um *SpamBand* composto por uma campanha de mensagens que não possuem texto em seu conteúdo encontrada no *honeypot* US-03 no dia 04 de abril de 2016. Nesse *SpamBand* encontramos apenas anexos, conforme

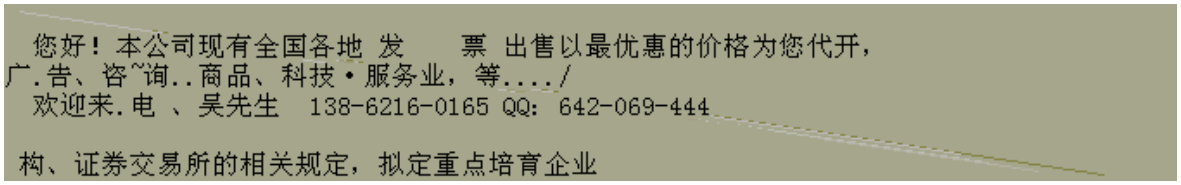


Figura 4.8: Exemplo do anexo de uma mensagem sem texto no corpo da mensagem.

mostrado na figura 4.8. Essa mensagem, em chinês, parece envolver a propaganda de serviços ilegais na China. Observamos que os atributos desse *SpamBand* (sumarizada na tabela 4.3) reforçam a atividade ilegal: apenas uma mensagem é enviada por endereço IP do *SpamBand* e cada uma dessas mensagens possui apenas um destinatário que envolvem os domínios 163.com e 126.com, o que acreditamos ser alvos específicos da propaganda. Observamos ainda que apenas os dois endereços IP do AS 18182 (SONET-TW Sony Network Taiwan Limited) estão na PBL e que todas as máquinas utilizadas nesse *SpamBand* se localizam em Taiwan.

Tabela 4.3: Atributos do *SpamBand* com mensagens com conteúdo de texto vazio.

ATRIBUTO	VALOR
# mensagens	17
# endereços IP distintos	17
# número de campanhas	1
ASes	3462,18182
Country Codes	TW
Protocolos (# IP)	SMTP (17)
blacklist (# IPS)	PBL (15)
Idioma das mensagens	chinês

4.4.2 SpamBand com cinco idiomas

Nesta seção, apresentamos um estudo de caso com um *SpamBand* com cinco idiomas encontrado no *honeypot* do Uruguai (UY-01) no dia 07 de abril de 2016. Observe que grande parte dos endereços IP desse *SpamBand* utilizam o protocolo HTTP/SOCKS. Esses endereços estão ligados ao envio das mensagens em russo, japonês e chinês presentes no *SpamBand*. Apenas o endereço IP que utiliza SMTP, localizado no AS 197226 (SPRINT-SDC), envia as mensagens em italiano e francês. Investigamos a conexão desse endereço IP com os demais e descobrimos que ele participa no envio de mensagens

em chinês do *SpamBand*, compartilhando campanhas com grande parte dos endereços IP que utilizam HTTP/SOCKS. Os atributos desse *SpamBand* estão sumarizadas na tabela 4.4.

Nós observamos também alguns fatos interessantes nesse *SpamBand*: metade dos endereços IP utilizando o protocolo SOCKS possuem o *country code* do Brasil e enviam mensagens em chinês e russo, indicando que este país possui máquinas exploradas por *spammers* de outras regiões do mundo para esconder sua identidade. Observamos ainda que somente metade das máquinas do Brasil nesse *SpamBand* estão na PBL e que o único endereço IP que está na XBL é um endereço HTTP/SOCKS do mesmo país.

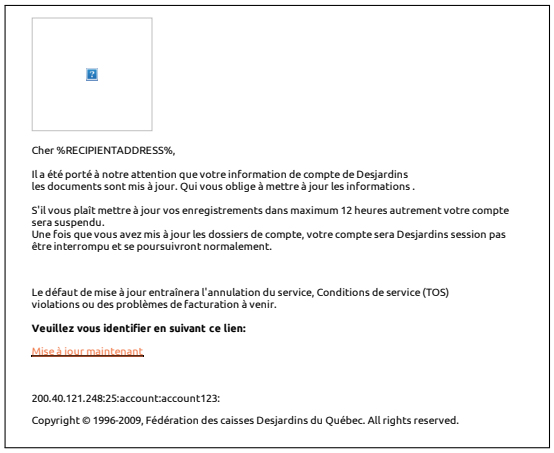
Tabela 4.4: Atributos do *SpamBand* com cinco idiomas.

Atributo	Valor
# mensagens	8.498
# endereços IP distintos	25
# número de campanhas	23
ASes	16735, 6830, 8167, 263599, 4134, 28343, 3269, 27699, 9116, 8615, 2514, 10429, 4766, 14868, 53006, 9595, 7029, 197226, 12334
Country Codes	ES, RU, CN, JP, IT, US, KR, IL, BR, PL
Protocolos (# IP)	SOCKS (24), SMTP (1)
blacklist (# IPS)	PBL (10), XBL (1)
Idioma das mensagens	russo, francês, italiano, japonês e chinês
Coef. Agrupamento	0,99

As mensagens enviadas em francês e italiano pelo endereço IP que utilizam SMTP são *phishing*, como mostra a figura 4.9. A mensagem em francês tem o objetivo de fazer o usuário clicar na URL do email, ameaçando cancelar a conta do usuário na associação de crédito Desjardins Group. A mesma técnica é utilizada na mensagem em italiano porém para o contexto de serviços de correios. Esse *SpamBand* também envia mensagens de propaganda conforme pode ser visualizado na figura 4.10, que mostra exemplos de mensagem em russo e japonês. A mensagem em russo oferece serviços de advocacia para empresas em relação à normas governamentais e a mensagem em japonês oferece produtos eróticos.

Por fim, apresentamos um exemplo de mensagem em chinês na figura 4.11 que utiliza termos aleatórios para confundir filtros de *spam*. Observe que na figura 4.11a não aparece nenhum desses termos dado que eles estão com a mesma cor de fundo da mensagem. Nós evidenciamos esses termos na figura 4.11b. Além disso, é possível

notar que não existe URL na mensagem: o *spammer* pede ao usuário que copie e cole o conteúdo no browser, indicando que métodos que utilizam URLs em mensagens para estudos devem ser estendidos para capturar esse tipo de técnica.



Cher %RECIPIENTADDRESS%,

Il a été porté à notre attention que votre information de compte de Desjardins
les documents sont mis à jour. Qui vous oblige à mettre à jour les informations .

S'il vous plaît mettre à jour vos enregistrements dans maximum 12 heures autrement votre compte
sera suspendu.
Une fois que vous avez mis à jour les dossiers de compte, votre compte sera Desjardins session pas
être interrompu et se poursuivront normalement.

Le défaut de mise à jour entraînera l'annulation du service, Conditions de service (TOS)
violations ou des problèmes de facturation à venir.

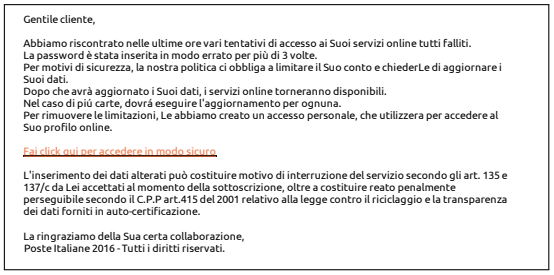
Veuillez vous identifier en suivant ce lien:

Mise à jour maintenant

200.40.121.248:25:account:account123:

Copyright © 1996-2009, Fédération des caisses Desjardins du Québec. All rights reserved.

(a) Exemplo de mensagem em francês.



Gentile cliente,

Abbiamo riscontrato nelle ultime ore vari tentativi di accesso ai Suoi servizi online tutti falliti.
La password è stata inserita in modo errato per più di 3 volte.
Per motivi di sicurezza, la nostra politica ci obbliga a limitare il Suo conto e chiederLe di aggiornare i Suoi dati.
Dopo che avrà aggiornato i Suoi dati, i servizi online torneranno disponibili.
Nel caso di piú carte, dovrá eseguire l'aggiornamento per ognuna.
Per rimuovere le limitazioni, Le abbiamo creato un accesso personale, che utilizzera per accedere al Suo profilo online.

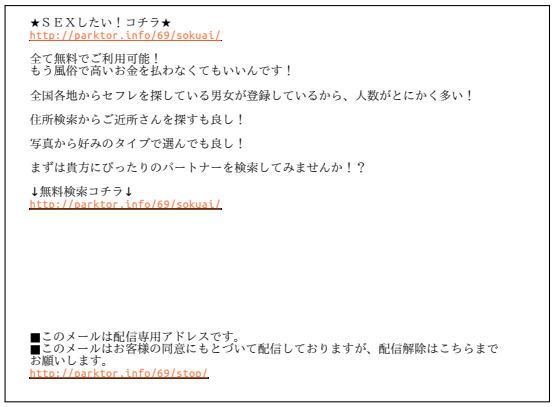
Fai click qui per accedere in modo sicuro

L'inserimento dei dati alterati può costituire motivo di interruzione del servizio secondo gli art. 135 e 137/c da Lei accettati al momento della sottoscrizione, oltre a costituire reato penalmente perseguibile secondo il C.P.P art.415 del 2001 relativo alla legge contro il riciclaggio e la transparenza dei dati forniti in auto-certificazione.

La ringraziamo della Sua certa collaborazione,
Poste Italiane 2016 - Tutti i diritti riservati.

(b) Exemplo de mensagem em italiano.

Figura 4.9: Mensagens em italiano e francês.



★ＳＥＸしたい！コチラ★
http://parktor.info/69/sokuai/

全て無料でご利用可能！
もう風俗で高いお金を払わなくてもいいんです！

全国各地からセフレを探している男女が登録しているから、人数がとにかく多い！

住所検索からご近所さんを探すも良し！

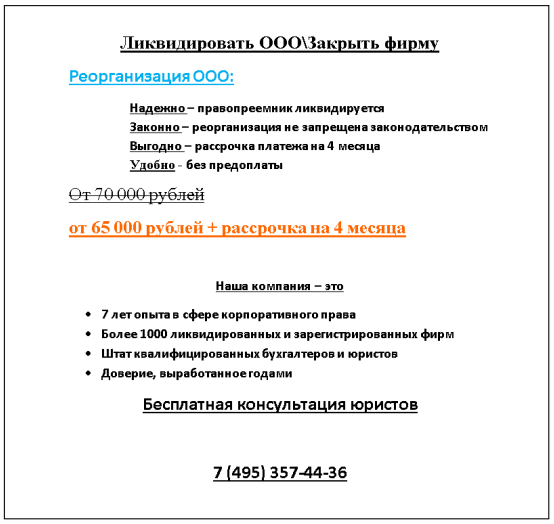
写真から好みのタイプで選んでも良し！

まずは貴方にぴったりのパートナーを検索してみませんか！？

↓無料検索コチラ↓
http://parktor.info/69/sokuai/

■このメールは配信専用アドレスです。
■このメールはお客様の同意にもとづいて配信しておりますが、配信解除はこちらまで
お願いします。
http://parktor.info/69/stop/

(a) Exemplo de mensagem em japonês.



Ликвидировать ООО\Закрыть фирму

Реорганизация ООО:

Надежно – правопреемник ликвидируется
Законно – реорганизация не запрещена законодательством
Выгодно – рассрочка платежа на 4 месяца
Удобно - без предоплаты

~~От 70 000 рублей~~

от 65 000 рублей + рассрочка на 4 месяца

Наша компания – это

- **7 лет опыта в сфере корпоративного права**
- **Более 1000 ликвидированных и зарегистрированных фирм**
- **Штат квалифицированных бухгалтеров и юристов**
- **Доверие, выработанное годами**

Бесплатная консультация юристов

7 (495) 357-44-36

(b) Exemplo de mensagem em russo.

Figura 4.10: Mensagens em japonês e russo.

(a) Exemplo de mensagem em chinês ofuscada original.

(b) Exemplo de mensagem em chinês com a ofuscação evidenciada.

Figura 4.11: Mensagem em chinês com termos aleatórios escondidos para confundir filtros.

4.5 Determinando o comportamento do spammer pelo idioma dos SpamBands

Nessa seção, descrevemos duas observações importantes em relação ao idioma presente nos *SpamBands* que nos permite entender algumas características atuais do *spammer* que utiliza servidores mal configurados para o envio de suas mensagens. A primeira delas é a relação entre o idioma utilizado pelos *spammers* e o ccTLD (Top Level Domain) dos destinatários que tendem a coincidir no *SpamBand*. Essa observação é sumarizada na tabela 4.5 que apresenta relações entre idioma e os ccTLDs dos domínios dos destinatários presentes no *SpamBand*. Pode-se observar uma relação interessante: se o *SpamBand* possui campanhas de um determinado idioma (primeira coluna), ele tende a ter um ccTLD relacionado daquele idioma. Por exemplo, mais de 99% dos *SpamBands* que possuem o idioma japonês também possuem o ccTLD ".jp" e apenas 24% dos *SpamBands* que não enviam japonês possuem o ccTLD ".jp". É possível observar que este cenário acontece para todos os idiomas considerados: a porcentagem de *SpamBands* com o par idioma e ccTLD relacionado supera a porcentagem de *SpamBands* que não possuem o idioma mas possuem o ccTLD relacionado. Dessa forma, é possível perceber, diante os dados analisados, que o *spammer* procura maximizar sua chance de sucesso personalizando o idioma das mensagens para domínios.

O segundo fato interessante que determinamos é em relação ao uso de protocolos. A tabela 4.6 mostra as porcentagens dos idiomas por protocolo nos *SpamBands*. Por

Tabela 4.5: Relação ccTLD e idioma nos *SpamBands*.

IDIOMA	CCTLD REPRESENTATIVO	% DE SPAMBANDS COM IDIOMA E CCTLD	% DE SPAMBANDS SEM IDIOMA E COM O CCTLD
Alemão	.de	62,14	20,60
Chinês	.tw	62,25	23,59
Espanhol	.mx	70,19	14,54
Francês	.fr	43,24	23,43
Holandês	.nl	31,89	19,75
Inglês	.uk	62,04	11,04
Italiano	.it	65,78	21,60
Japonês	.jp	99,89	24,20
Português	.br	48,33	19,45
Russo	.ru	96,27	18,55

Tabela 4.6: Distribuição dos idiomas por protocolo.

IDIOMA	SPAMBANDS		
	PROXY	RELAY	AMBOS
Alemão	0,01	41,24	58,75
Chinês	22,20	74,46	3,34
Espanhol	0,01	79,17	20,82
Francês	0,01	63,51	36,48
Holandês	0,01	57,46	42,53
Inglês	0,01	79,71	20,28
Italiano	0,00	77,75	22,25
Japonês	96,08	0,10	0,02
Português	0,01	56,97	43,02
Russo	28,73	22,69	48,58

exemplo, de todos os *SpamBands* que enviam japonês, 96,08% possuem apenas endereços IP que exploram o *honeypot* com o protocolo HTTP/SOCKS. A tabela nos permite inferir dois fatos interessantes: (i) os protocolos HTTP/SOCKS, que exploram o *honeypot* como proxy aberto, possuem uma maior representatividade nos idiomas chinês, japonês e russo em relação aos demais, sugerindo que *spammers* que enviam mensagens nesses idiomas podem estar utilizando máquinas dedicadas ao envio, principalmente mensagens em japonês dado que o Japão teve uma grande adoção de gerência de porta 25; (ii) *SpamBands* com idiomas ocidentais enviam suas mensagens explorando o *honeypot* tanto como relay aberto quanto utilizando ambos ambos os protocolos, sendo

raramente compostos apenas por endereços IP que utilizam HTTP/SOCKS. Esse fato indica que *spammers* orientais, na base observada, utilizam servidores dedicados para envio de suas mensagens e buscam os *honeypots* para esconder sua identidade enquanto *spammers* ocidentais tendem a enviar suas mensagens a partir de serviços de *botnets* uma vez que utilizam o *honeypot* diretamente como um servidor de email. Esse fato, em particular, sugere que o combate ao *spammer* deveria divergir segundo nossos dados: países ocidentais poderiam focar em identificar e bloquear o envio de mensagens por *botnets* como, por exemplo, fazendo a gerência da porta 25, enquanto países orientais, principalmente o Japão que implanta fortemente a gerência de porta 25, devam focar no combate a ASes coniventes com o tráfego conforme indicado na literatura [Fonseca et al., 2016].

4.6 Entendendo os tipos de campanhas através dos SpamBands

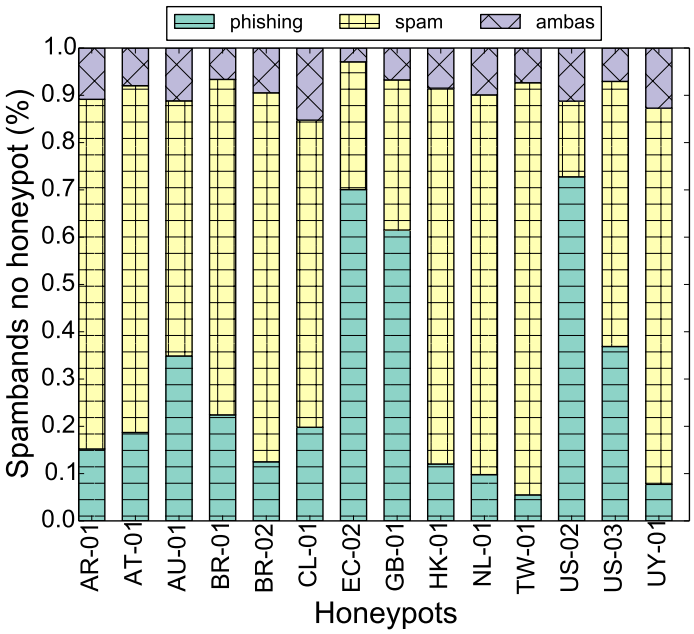


Figura 4.12: Distribuição dos *SpamBands* por especialidade: propaganda ou *phishing*

Nessa seção buscamos entender diferenças da infraestrutura utilizada por diferentes tipos de *spammer*. Aplicamos a metodologia descrita na seção 3.6 para classificar a mensagem representante de cada campanha nos *SpamBands* como *spam* de propaganda ou *phishing*. Relacionamos as campanhas que constituem um *SpamBand* da seguinte forma: *SpamBands* especializados em *phishing*, *SpamBands* especializados em

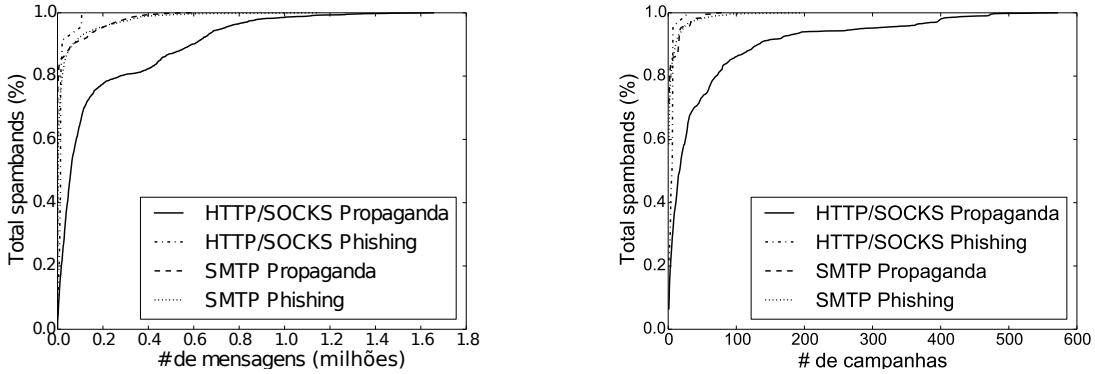
propaganda e *SpamBands* que enviam ambos tipos de mensagens. Nós não avaliamos os *SpamBands* cujas campanhas eram mensagens sem conteúdo uma vez que a técnica utilizada para a classificação depende da presença de texto. Também descartamos campanhas formadas por idiomas que não fossem os dez mais frequentes considerados, uma vez que o algoritmo que diferencia propaganda de *phishing* não possui mensagens desses idiomas no treino fornecido. Sumarizamos as especializações dos *SpamBands*, por *honeypot*, na figura 4.12.

Tabela 4.7: Matriz de relação dos *SpamBands* entre as especialidades do *spammer* e protocolo.

PROTOCOLO NO SPAMBAND	TIPO DA CAMPANHA		
	PROPAGANDA (%)	PHISHING (%)	AMBOS (%)
SMTP	58,27	37,73	4,00
HTTP/SOCKS	88,80	0,93	10,27
Ambos	41,88	0,36	57,76

Nós observamos que grande parte dos *SpamBands* encontrados em todos os *honeypots* (65.85%) são especializados no envio de mensagens com conteúdo de propaganda, seguido de *SpamBands* especializados no envio de mensagens de *phishing* (24.83%) e *SpamBands* que enviam ambos tipos de mensagens (89.68%). Para entender qual a infraestrutura utilizada por esses *SpamBands*, buscamos correlacionar o tipo de especialização do *SpamBand* com o protocolo utilizado, conforme mostrado na tabela 4.7. Podemos observar que 88,80% de todos os *SpamBands* com apenas protocolos HTTP/SOCKS se especializam no envio de propaganda. Ainda é possível observar que mais da metade de todos os *SpamBands* constituídos apenas de SMTP (58,20%) enviam propagandas e que os *SpamBands* que enviam *phishing* estão mais relacionados ao protocolo SMTP.

Na seção 4.3 foi observado que *SpamBands* que enviam mensagens apenas pelos protocolos HTTP/SOCKS estão envolvidos em um número maior de campanhas que *SpamBands* que enviam mensagens apenas pelo protocolo SMTP. Buscamos entender como esses *SpamBands* se relacionam com o tipo de mensagem que enviam analisando apenas aqueles especializados em um tipo de mensagem: propaganda ou *phishing*. Na figura 4.13a analisamos como cada *SpamBand*, por protocolo e por classe, se comportam em relação ao número de mensagens enviadas. Pode-se observar que *SpamBands* HTTP/SOCKS de propaganda enviam um número muito maior que os demais *SpamBands* analisados que são *SpamBands* apenas HTTP/SOCKS que enviam *phishing*,



(a) Por mensagem **(b)** Por campanha

Figura 4.13: Distribuição dos *SpamBands* por classe (propaganda e *phishing*) nos protocolos em relação ao número de mensagens e número de campanhas

Tabela 4.8: Relação entre idioma e especialização do *SpamBand*

Idioma	Especialização do SpamBand	
	Propaganda (%)	Phishing (%)
Alemão	66,21	33,79
Chinês	99,60	0,40
Espanhol	0,01	99,99
Francês	0,01	99,99
Holandes	0,01	99,99
Inglês	27,23	72,77
Italiano	0,01	99,99
Japonês	98,72	1,28
Português	0,00	100,00
Russo	99,51	0,49

SpamBands apenas SMTP que enviam propaganda e *SpamBands* apenas SMTP que enviam *phishing*. Além disso, é possível observar que esses *SpamBands* enviam uma quantidade similar de mensagens. Realizando a mesma análise a nível de campanha (figura 4.13b), observamos um comportamento similar ao anterior que nos permite suspeitar que *SpamBands* HTTP/SOCKS são grupos que parecem ser especializados no envio de propaganda e que podem estar sendo contratados por diversos clientes, fato que explica a grande quantidade de campanhas observadas.

Também buscamos entender como os idiomas estão envolvidos nas especializações dos *spammers* através dos *SpamBands* especializados, ou seja, ou que enviam apenas *phishing* ou que enviam apenas propaganda. A tabela 4.8 mostram como estão divi-

didos os idiomas entre esses *SpamBands*. Pode-se observar que idiomas orientais como russo, japonês e chinês estão ligados a *SpamBands* de propaganda e que o *phishing* está concentrado em *SpamBands* com idiomas ocidentais. Essas observações sobre os *SpamBands* em conjunto com a tabela 4.6 sugerem uma divisão dos grupos de *spammers* ao redor do mundo segundo a visão oferecida por nossos dados. De um lado temos mensagens com idiomas orientais que podem estar sendo enviadas diretamente de máquinas do *spammer* uma vez que este tenta esconder sua identidade utilizando os protocolos HTTP/SOCKS. Do outro lado, temos idiomas ocidentais que estão mais intimamente ligados ao envio de *phishing* com a exploração de *relays* abertos, indicativo de *botnets*.

Capítulo 5

Encontrando grupos persistentes ao longo do tempo

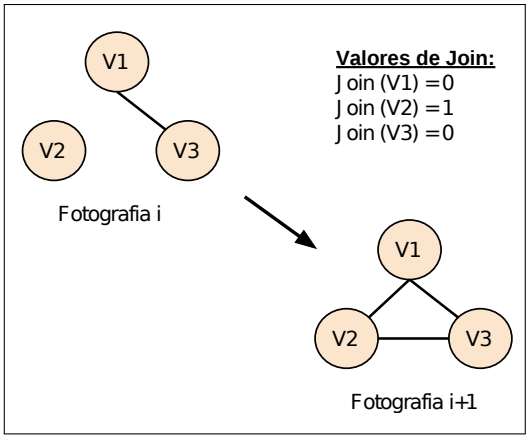
Neste capítulo, procuramos entender como os endereços IP presentes nos diferentes *SpamBands* detectados colaboram entre si para enviar mensagens ao longo do período avaliado. Mais especificamente, estamos interessados em encontrar grupos de endereços IP que participam dos mesmos *SpamBands* ao longo dos dias pois eles podem revelar redes bem definidas de envio que não são possíveis de observar no período de um dia. Para identificar esses grupos colaborativos nos *honeypots* apresentamos uma adaptação do algoritmo de Asur et al. [Asur et al., 2009] explicada na seção 5.1 e apresentamos as principais observações sobre os grupos detectados na seção 5.2.

5.1 Método para identificação de grupos colaborativos

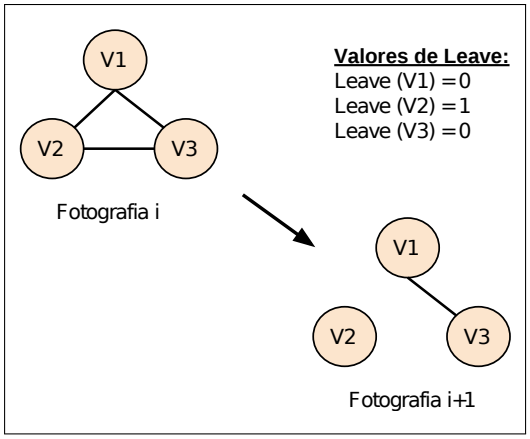
Os *SpamBands* podem ser vistos como fotografias de como as máquinas se organizam na Internet para enviar *spam* e permitem avaliar questões sobre a dinamicidade das máquinas na rede. Existem técnicas de mineração de dados propostas na literatura que abordam a avaliação de grafos dinâmicos. Em nosso trabalho, utilizamos a técnica proposta por *Asur et al.* [Asur et al., 2009] que se baseia em eventos ocorridos ao longo do tempo, como saída e entrada de vértices de um componente no grafo ao longo do tempo.

A técnica proposta pelo artigo explora diversos eventos. Em nosso trabalho, estamos interessados em apenas dois que focam em movimento dos vértices: *join* e *leave*. O evento *join* ocorre quando um vértice que não pertence a um grupo g em

uma fotografia anterior e passa a pertencer ao grupo g em uma fotografia posterior. O evento *leave* é o oposto do evento *join*: um vértice que pertence a um grupo g em uma fotografia anterior passa a não pertencer ao grupo g em uma fotografia posterior. Uma ilustração desses eventos é mostrada na figura 5.1. Em nosso trabalho, um grupo é representado por um *SpamBand* e um vértice são endereços IP. Consideramos também que um grupo g de uma fotografia i persiste na fotografia $i + 1$ se existe um grupo nesta fotografia que tenha mais da metade dos vértices do grupo g naquela fotografia. Além disso, consideramos que todo novo vértice realiza um evento de *join* para algum grupo e todo vértice que desaparece entre duas fotografia realiza um evento *leave*. Consideramos ainda que todos os vértices presentes na primeira fotografia realizam um evento *join* nos grupos dessa fotografia. Escolhemos essa abordagem para tratar a ausência de informação anterior à primeira fotografia de maneira conservadora, ou seja, não assumir que tanto os vértices quanto os grupos já existiam anteriormente.



(a) Exemplo de um evento *Join*



(b) Exemplo de um evento *Leave*

Figura 5.1: Ilustração dos eventos ocorridos em comunidades.

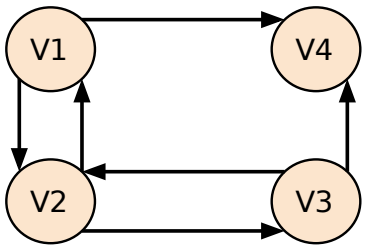
Esses dois eventos são importantes para explicar o índice que utilizamos do trabalho de *Asur et al.* para avaliar comportamentos de máquinas ao longo do tempo: o índice de influência. Esse índice mede o grau de influência de um vértice sobre os demais vértices durante um intervalo de tempo. Esse índice é calculado como se segue. Seja $\mathcal{E}(x)$ um evento *leave* ou *join* ocorrido para um vértice x e $\mathcal{C}(x)$ o número de vértices que participaram do mesmo evento, ou seja, saíram do mesmo grupo g se ocorreu um evento *leave* para o vértice x ou entraram no mesmo grupo g se o evento para x foi *join*. Seja também $\mid \mathcal{E}(x) \mid$ o número de eventos que o vértice x participou e $\sum \mathcal{C}(x)$ a soma de todos os $\mathcal{C}(x)$ para todos os eventos do vértice x. Calculamos o índice de acordo com a equação abaixo:

$$\mathcal{I} = \frac{\sum \mathcal{C}(x)}{\mid \mathcal{E}(x) \mid}$$

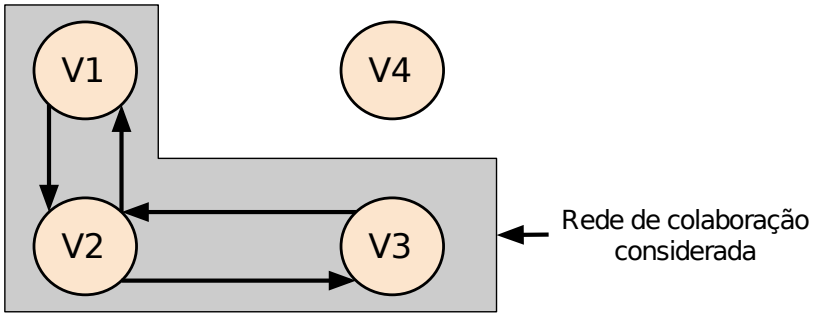
É possível notar que quanto mais vértices mudam junto com o vértice x nas diversas fotografias, maior o índice de influência. É importante ressaltar que esse índice não está normalizado no intervalo $[0, 1]$.

	V1	V2	V3	V4
V1	-	0,6	0,3	0,7
V2	0,55	-	0,8	0,2
V3	0,4	0,9	-	0,7
V4	0,1	0,2	0,1	-

(a) Exemplo de matriz de influência



(b) Grafo de influência utilizando a matriz 5.2a com $l = 0.5$



(c) Rede de colaboração detectada entre os vértices V1, V2 e V3

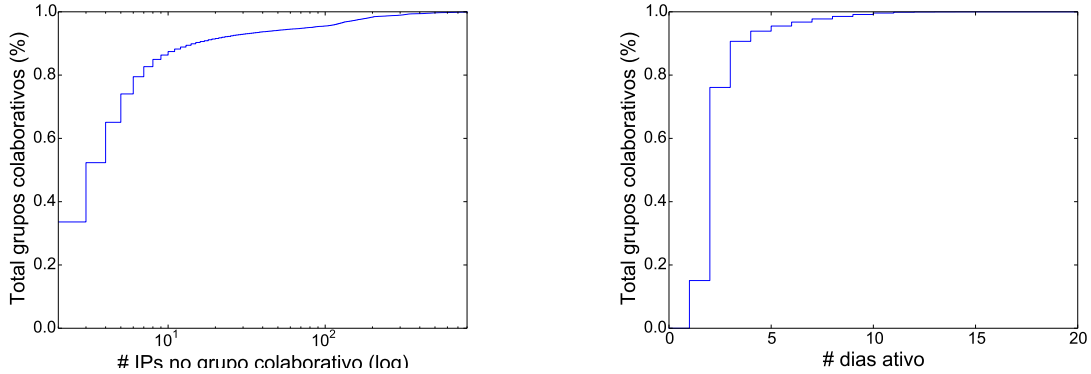
Figura 5.2: Detecção de redes de colaboração usando o índice de influência.

Para identificar endereços IP que podem estar atuando em conjunto durante um período p podemos modelar a influência como um grafo direcionado como se segue. Para cada vértice x, calculamos o índice de influência dele para outro vértice y. O cálculo que utilizamos é o mesmo utilizado por Asur et al. porém consideramos apenas o vértice y para calcular $\sum \mathcal{C}(x)$. Nesse caso, o valor do índice está no intervalo $[0, 1]$ dado que o maior valor que $\sum \mathcal{C}(x)$ pode atingir é a movimentação do vértice x (ou seja, o vértice y participa dos mesmos eventos que o vértice x). Um exemplo dessa matriz é ilustrada na tabela 5.2a. Utilizando um limiar l que define o nível de influência mínimo a partir do qual um vértice x é considerado exercer influência sobre o vértice y, inserimos uma aresta direcionada de x para y caso o nível de influência de x para y for maior que o limiar l estipulado. Um exemplo desse grafo é mostrado na figura 5.2b. A partir

desse grafo, removemos todas as relações unilaterais entre os vértices e extraímos os componentes fortemente conectados com mais de um endereço IP, onde cada um desses componentes indicam uma rede de colaboração (ou *grupo colaborativo*) entre endereços IP no intervalo de tempo p ilustrado pela figura 5.2c.

5.2 Avaliando grupos colaborativos

Nesta seção, buscamos entender o comportamento dos grupos de endereços IP que atuam juntos nos mesmos *SpamBands* em diversos dias ao longo do período analisado. Aplicamos a técnica descrita da seção 5.1, por *honeypot*, com o limiar máximo (l=1,0) para para descobrir os grupos de endereços IP estritamente relacionados em nossa base, ou seja, agrupamos todos os endereços IP que participam exatamente dos mesmos eventos.



(a) Distribuição do tamanho dos grupos colaborativos (escala log).

(b) Distribuição do número de dias ativos dos grupos colaborativos.

Figura 5.3: Visão geral do tamanho e número de dias ativos de cada grupo.

Ao todo, descobrimos 20,033 grupos diferentes ao longo do tempo nos *honeypots*. Esses grupos são responsáveis pelo envio de 30% das mensagens da base. Verificamos que uma parte desses grupos utilizam apenas HTTP/SOCKS (28,83%), outra apenas SMTP (69,89%) e uma pequena parte utiliza ambos protocolos (1,28%) e que 90% deles têm 20 endereços IP ou menos e existe uma pequena porcentagem (4,49%) de grupos com mais de 100 endereços IP conforme mostrado na figura 5.3a. Verificamos que os grupos com menos de 20 endereços IP enviam 49% das mensagens e os grupos com mais de 100 endereços IP enviam 46%. É importante mencionar que não encontramos nenhuma distinção na proporção de protocolos entre eses dois grupos: grupos com 20 endereços IP estão divididos em de 67,14% apenas SMTP, 27,83% ape-

nas HTTP/SOCKS e 5.01% de ambos enquanto grupos com mais de 100 endereços IP estão divididos em 69,51% apenas SMTP, 29,67% apenas HTTP/SOCKS e 0,82% de ambos protocolos.

A figura 5.3b mostra o número de dias ativos dos grupos colaborativos. Pode-se verificar que mais de 90,66% deles apareceram em três dias ou menos e que apenas 0,85% ficaram ativos por dez dias ou mais durante o período analisado. Nós tentamos entender a estabilidade desses grupos, ou seja, dado o período entre o primeiro e último dia que observamos um grupo queremos saber qual a porcentagem de dias que ele ficou ativo. Seja $\mathcal{D}_i$ e $\mathcal{D}_f$, respectivamente, o primeiro e último dia que observamos um grupo e $\mid \mathcal{D}_f - \mathcal{D}_i \mid$ o número de dias entre $\mathcal{D}_i$ e $\mathcal{D}_f$ (inclusive). Seja $\mid \mathcal{D}_a \mid$ o número de dias ativos do grupo. Calculamos a estabilidade $\mathcal{S}$ de um grupo colaborativo como:

$$\mathcal{S} = 1 - \frac{\mid \mathcal{D}_f - \mathcal{D}_i \mid - \mid \mathcal{D}_a \mid}{\mid \mathcal{D}_f - \mathcal{D}_i \mid}$$

Grupos colaborativos com estabilidade $\mathcal{S} = 1,0$ são grupos que ficaram ativos em todos os dias entre o primeiro e último dia em que foram detectados, ou seja, são bastante estáveis. Ou seja, quanto maior esse valor mais constante é um grupo durante o período que ele foi observado e vice-versa. Pode-se observar na figura 5.4 que apenas 17% dos grupos encontrados possuem $\mathcal{S} = 1,0$: eles surgem, enviam suas mensagens e desaparecem no período que avaliamos. Nós também verificamos que 38,15%, 57,99% e 3,86% desses grupos com estabilidade máxima são, respectivamente, apenas HTTP/SOCKS, apenas SMTP e ambos. Também é possível notar que cerca de 60% dos grupos tem uma estabilidade igual ou menor que $\mathcal{E} = 0,3$, indicando grupos pouco estáveis, ou seja,enviam suas mensagens em dias não consecutivos. Nós também avaliamos os protocolos utilizados por esses grupos colaborativos e verificamos que 32,93% e 67,04% são, respectivamente, apenas HTTP/SOCKS e apenas SMTP (não encontramos nenhum grupo nessa categoria que utiliza ambos os protocolos). As próximas seções mostram exemplos de grupos colaborativos.

5.2.1 Furto de contas do PayPal

Nessa seção descrevemos um grupo de colaboração encontrado no *honeypot* US-03 nos dias 27, 28 e 31 de maio de 2016. Esse grupo de colaboração é formado por apenas três endereços IP que aparecem *somente* nos dias citados, nos mesmos *SpamBands*, e enviam mensagens conforme mostrado pela figura 5.5. Verificamos que esses endereços podem ser de *bots* dado que dois endereços IP possuem *country codes* associados à Bolívia e um à Turquia (regiões distantes), além dos três serem de três sistemas autônomos

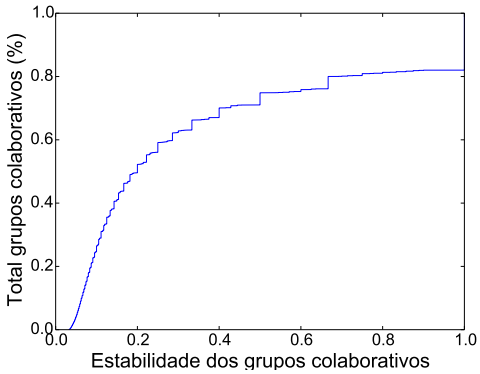


Figura 5.4: Estabilidade dos grupos colaborativos encontrados.

diferentes e utilizarem o protocolo SMTP para o envio de suas mensagens conforme mostrado na tabela 5.1.

Dear Valued Client,

Your PayPal account is limited until we can hear from you.

We have been closely monitoring unusual activities coming from your account. To prevent unauthorized activity, we have limited certain actions such as sending, receiving, and withdrawing money from the account.

In most instances, it is easy to regain access to your account, we just need to get some information from you to confirm your identity.

Please follow the simple steps to restore your account:

- Download the attachment (it contains a form).
- Open and complete the form.

After we review your information, the restrictions will be lifted and you may get back to using your account's full access privileges. We apologize for the inconvenience.

Sincerely,
PayPal Security Team

Attachment: attached248.html

Anexos

Nome	Tamanho
attached248.html	33,7 kB

Figura 5.5: Exemplo de mensagem do PayPal enviada pelo grupo de colaboração detectado.

5.2.2 Propagação de malware

Nessa seção apresentamos um exemplo de grupo colaborativo, composto por dois endereços IP, que enviam mensagens em holândes com *malware* para domínios holandeses. Encontramos esse grupo no *honeypot* AT-01 que atua apenas no dia 1 de abril de 2016 e desaparece após o envio dessa campanha. A figura 5.6 mostra um exemplo dessa mensagem e a tabela 5.2 mostra os atributos desse grupo colaborativo. Inicialmente,

Tabela 5.1: Atributos do grupo colaborativo para envio de campanhas para furto de contas do Paypal.

Atributo	Valor
# mensagens	385
# endereços IP distintos	3
ASes	26210, 6568, 8517
Country Codes	BO, TR
Protocolos (# IP)	SMTP (3)
blacklist (# IPS)	XBL (3), PBL (1)
Idioma das campanhas	inglês

o *spammer* (ou phisher) cria um senso de urgência para o usuário pagar um título que se encontra em anexo ao email. Entretanto, verificando o anexo encontramos que, na verdade, é um executável que está disfarçado com o final ".pdf.exe", indicando que o objetivo do *spammer* não é o pagamento da fatura do título mas a propagação de *malware*. Nós verificamos que ambos endereços IP possuem *country code* do Reino Unido e estão localizados no sistema autônomo 20738 (Host Europe GmbH). Nós verificamos os domínios dos recipientes e encontramos 19.682 domínios únicos sendo todos com ccTLD ".nl", indicando uma campanha específica da Holanda.

Behandeld door : TransIP Domein Registratie
Direct tel. nr. : 082 - 453 75 39

Openstaande vordering, **BELANGRIJK!**

Geachte klant ,

Verwijzen wij u naar de eerder ontvangen herinnering(en). Wij stellen u hierbij
de gelegenheid om het verschuldigde bedrag van 5280,89 met rente binnen 14 dagen te voldoen op
ons IBAN-rekeningnummer NL52ABNA0482467390 t.n.v. St. Derdengeleden Intrum Justitia Nederland B.V.
onder vermelding van het referentienummer.

Blijft betaling uit, dan zijn wij genoodzaakt cliınt te adviseren om over te gaan tot het opstarten van een
gerechtelijke procedure. De kosten die hieruit voortvloeien zullen geheel voor uw rekening komen.
Voor directe betaling en meer informatie over deze vordering gaat u naar onze website
U kunt hiervoor de gegevens gebruiken die op de factuur staan vermeld. Zie bijgevoegde factuur voor meer informatie

Hoogachtend,

Intrum Justitia

De buitengerechtelijke incassokosten kunnen zijn verhoogd met btw in het geval dat de schuldeiser een niet btw-plichtige ondernemer is in de zin van art. 7 en 11 van de Wet op de omzetbelasting 1968

Intrum Justitia Nederland BV Handelend onder de naam Intrum Justitia
Postbos 80046 2552 AB Den Haag H.R. Den Haag 27494742
BTW nr. NL008338640B07 Lid van NVI

Figura 5.6: Exemplo de mensagem em holandês para pagamento de títulos.

Tabela 5.2: Atributos do grupo colaborativo para envio de *phishing* holandês.

Atributo	Valor
# mensagens	1.085
# endereços IP distintos	2
ASes	20738
Country Codes	GB
Protocolos (# IP)	SMTP
blacklist (# IPS)	XBL (2)
Idioma das campanhas	holandês

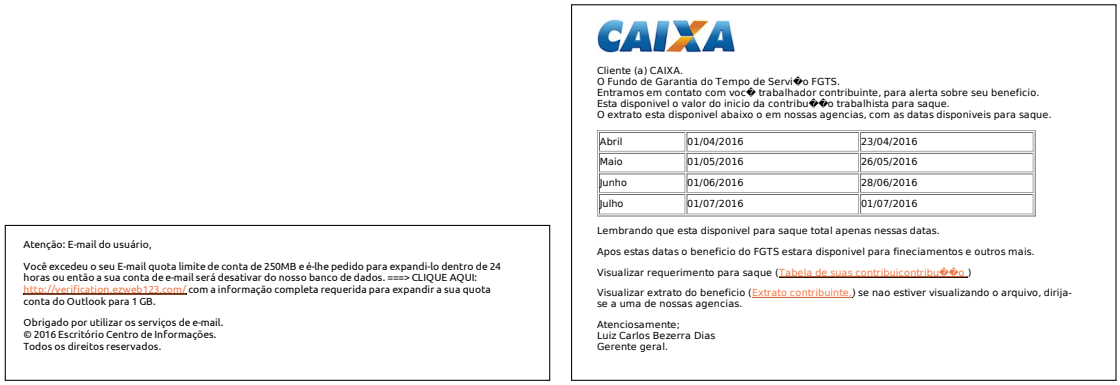
5.2.3 Mensagens de phishing com idioma português

Nesta seção, dedicamos um estudo sobre grupos que colaboram para enviar campanhas brasileiras ao longo do tempo. Nós identificamos que os grupos colaborativos que atuam em campanhas brasileiras aparecem, em geral, apenas em um dia no *honeypot* onde foi detectado e que são campanhas pequenas em seu número de mensagens. Descrevemos cada caso a seguir:

***Phishing* de serviços de email:** Detectamos dois endereços IP do sistema autônomo 12091 (MTNNS-1) com *country code* da Africa do Sul (ZA), no *honeypot* AU-01, que tentam enviar cinco mensagens com intuito de obter informações do usuário conforme mostrado na figura 5.7a. Nós identificamos que esses endereços aparecem, no *honeypot*, apenas no dia 13 de abril de 2016 e suas mensagens são identificadas como mensagens de teste.

***Phishing* do banco CAIXA:** Identificamos no dia 19 de abril de 2016, no *honeypot* US-03, quatro endereços IP colaborando para uma campanha composta de 24 mensagens de *phishing* com tema da Caixa Econômica Federal conforme mostrado na figura 5.7b. Observamos que três desses endereços eram do sistêma autônomo 8075 (Microsoft) e indicam serem de máquinas da plataforma Azure, enquanto um dos endereços é do 20473 (AS-CHOOPA). Observamos que as mensagens são destinadas a 406 domínios diferentes e sendo que 396 deles possuem o ccTLD ".br".

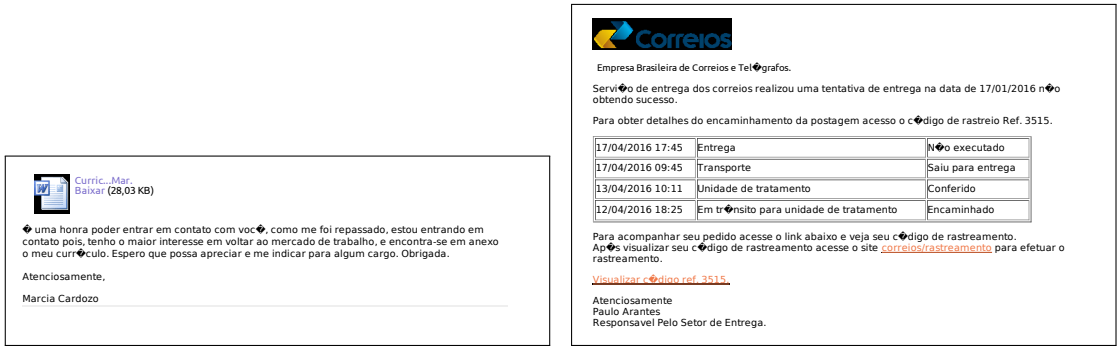
Propagação de *malware* e *phishing* dos Correios: No dia 18 de abril de 2016 no *honeypot* US-03, identificamos três endereços IP colaborando para o envio de 19 mensagens com um objetivo: propagação de *malware*. Nós observamos, entretanto, que o *spammer* utiliza de dois temas distintos para levar o usuário a clicar nas URLs presentes na mensagem: uma ofertando trabalho ao destinatário e outra apresentando



(a) Mensagem requisitando informações de serviço de email **(b)** *phishing* do banco CAIXA.

Figura 5.7: Exemplo de mensagens de *phishing* com temas de serviços de email e do banco CAIXA.

um rastreio de um produto nos Correios. Observamos que um dos endereços IP vem do sistema autônimo 8075 (Microsoft) e os outros dois são do sistema autônomo 20473 (AS-CHOOPA). Também verificamos um total de 253 domínios distintos para onde essas mensagens foram endereçadas, sendo 229 com ccTLD ".br".



(a) *phishing* com tema de ofertas de trabalho **(b)** *phishing* com tema dos Correios

Figura 5.8: Exemplos de mensagens de *phishing* com temas de ofertas de trabalho e dos Correios.

***Phishing* com tema das lojas Americanas:** No dia 4 de maio de 2016 identificamos um *phishing* simulando uma propaganda das lojas Americanas para o dia das mães enviado por um endereço IP de um grupo colaborativo composto por 307 endereços IP distintos. Esse endereço IP pertence ao sistema autônomo 20473 (AS-CHOOPA). Observamos que as mensagens foram endereçadas para 23 domínios distintos sendo 16

deles com ccTLD ".br" e 7 com ccTLD ".com".

Figura 5.9: Exemplo de campanha de *phishing* com tema das lojas Americanas

Capítulo 6

Conclusão e Trabalhos Futuros

Neste trabalho, buscamos entender melhor a infraestrutura utilizada pelos *spammers* para o envio de campanhas de *spam*. Uma campanha é definida como um conjunto de mensagens similares entre si e que possuem o mesmo objetivo como a venda de um determinado produto ou furto de informações de usuários de um banco específico. Para identificar as campanhas de *spam* utilizamos o agrupamento baseado na árvore de padrões frequentes (FP-Tree) e criamos uma nova técnica de identificação de campanhas em FP-Trees. A técnica que apresentamos depende apenas de um único parâmetro e relaciona toda mensagem a uma, e apenas uma, campanha. Mostramos que os endereços IP que participam de apenas uma campanha não são suficientes para determinar a infraestrutura utilizada pelo *spammer* pois este pode utilizar apenas uma parte de uma infraestrutura maior para enviar aquela campanha ou o honeypot pode observar apenas parte do tráfego.

Para detectar a infraestrutura de forma mais ampla utilizada pelo *spammer*, propusemos uma técnica para detectar grupos de endereços IP relacionados pelas campanhas que participam: os *SpamBands*. Aplicamos a técnica em 650 milhões de mensagens de *spam* coletadas em quatorze *honeypots* de baixa interatividade que simulam *proxy* e *relay* abertos. A análise dos *SpamBands* detectados nesse conjunto de dados nos permitiu chegar a conclusões sobre *spammers* que utilizam essas duas formas de envio:

- *SpamBands* compostos por endereços IP que exploram os *honeypots* apenas por *relay* aberto (uso do protocolo SMTP) são a maioria. Mensagens de *phishing* tendem a ser enviadas por esse tipo de *SpamBand*. Conforme indicado na literatura, máquinas que utilizam o protocolo SMTP para se conectar diretamente ao *honeypot* é indicativo de *botnets* uma vez que o *spammer* não está preocu-

pado em esconder sua identidade como acontece em conexões feitas explorando o honeypot como proxy (protocolos HTTP ou SOCKS). Além disso, detectamos que *spams* que estão em idiomas ocidentais (inglês, espanhol, alemão, português, holândes, italiano e francês) são enviados, em quase sua totalidade, por esse tipo de *SpamBand*.

- Na base que utilizamos, observamos que *SpamBands* compostos por endereços IP que exploram os *honeypots* apenas por *proxy* aberto (uso dos protocolos HTTP ou SOCKS) estão bastante relacionados com a disseminação de *spam* em idiomas do oriente (russo, chinês e japonês) e o conteúdo das mensagens são, em sua grande parte, propagandas. Detectamos ainda que esses *SpamBands*, com conteúdo de propagandas, possuem um número muito maior de campanhas em relação a *SpamBands* que exploram o honeypot apenas como *relay* aberto, sugerindo que são possíveis infraestruturas utilizadas como comércio para anúncio de propagandas.

- Mostramos a existência de *SpamBands* que exploram o *honeypot* em ambas vulnerabilidades simuladas indicando *spammers* que utilizam tanto infraestruturas próprias quanto *botnets*. Mostramos um exemplo desse tipo de estrutura que envia propagandas farmacêuticas alemãs e russas.

- Mostramos exemplos de *SpamBands* que revelam que o *spammer* utiliza serviços de nuvem para o envio de *spam*, como é o caso da plataforma da Microsoft Azure. Também identificamos nesses exemplos que *SpamBands* relacionados ao furto de informações bancárias são formados por uma quantidade muito pequena de endereços IP embora um estudo mais específico seja necessário para confirmar essa observação. Mostramos ainda exemplos de campanhas em chinês que buscam ludibriar filtros de *spam* utilizando termos aleatórios nas mensagens.

Como trabalhos futuros, pretendemos realizar outras análises que não realizamos neste trabalho e estender a técnica para outras bases de mensagens de *spam*. A primeira consiste avaliar *SpamBands* que atacam múltiplos *honeypots*. Encontramos indícios que mostram que existe uma sobreposição de endereços IP nos honeypots. Entretanto não exploramos o fato do mesmo *SpamBand* atuar em vários *honeypots*. Acreditamos que isso levaria a detecção de grupos mais abrangentes no caso do mesmo *SpamBand* enviar diferentes campanhas entre diferentes *honeypots*. Entretanto, detectar o mesmo *SpamBand* entre vários *honeypots* é desafiador pois existe uma sobreposição parcial que torna complexo definir o mesmo *SpamBand* entre vários *honeypots*. A segunda análise envolve detectar *SpamBands* no conjunto agregado dos dados de todos os *honeypots*

de todo o período. Essa abordagem é desafiadora pois o tamanho do grafo produzido está na ordem de centenas de milhares de vértices e pode chegar a bilhões de arestas, exigindo tanto algoritmos eficientes quanto amplos recursos de processamento e armazenamento.

Referências Bibliográficas

Almeida, H.; Guedes, D.; Meira, W. & Zaki, M. J. (2011). Is there a best quality metric for graph clusters? Em *Proc. of Conference on Machine Learning and Knowledge Discovery in Databases (ECML)*.

Almomani, A.; Gupta, B.; Atawneh, S.; Meulenberg, A. & Almomani, E. (2013). A survey of phishing email filtering techniques. *IEEE communications surveys & tutorials*, 15(4):2070--2090.

Androutsopoulos, I.; Koutsias, J.; Chandrinos, K. V. & Spyropoulos, C. D. (2000). An experimental comparison of naive bayesian and keyword-based anti-spam filtering with personal e-mail messages. Em *Proc. in Conference on Research and development in Information Retrieval (SIGIR)*.

Asur, S.; Parthasarathy, S. & Ucar, D. (2009). An event-based framework for characterizing the evolutionary behavior of interaction graphs. *ACM Transactions on Knowledge Discovery from Data (TKDD)*.

Attenberg, J.; Weinberger, K.; Dasgupta, A.; Smola, A. & Zinkevich, M. (2009). Collaborative email-spam filtering with the hashing trick. Em *Proc. in Conference on Email and Anti-Spam (CEAS)*.

Bergholz, A.; De Beer, J.; Glahn, S.; Moens, M.-F.; Paaß, G. & Strobel, S. (2010). New filtering approaches for phishing email. *Journal of computer security*, 18(1):7--35.

Blanzieri, E. & Bryl, A. (2008). A survey of learning-based techniques of email spam filtering. *Artificial Intelligence Review*, 29(1):63--92.

Calais, P.; Pires, D. E.; Neto, D. O. G.; Meira Jr, W.; Hoepers, C. & Steding-Jessen, K. (2008). A campaign-based characterization of spamming strategies. Em *Proc. in Conference on Email and Anti-Spam (CEAS)*.

Chandrasekaran, M.; Narayanan, K. & Upadhyaya, S. (2006). Phishing email detection based on structural properties. Em *Proc.of NYS Cyber Security Conference*.

Clayton, R. (2004). Stopping spam by extrusion detection. Em *Proc. in Conference on Email and Anti-Spam (CEAS)*.

Cohen, W. W. (1996). Learning rules that classify e-mail. Em *Proc. of AAAI Spring Symposium on Machine Learning and Information Access*.

Cormack, G. V. (2008). Email spam filtering: A systematic review. *Found. Trends Inf. Retr.*, 1(4):335--455.

Cranor, L. F. & LaMacchia, B. A. (1998). Spam! *Communications of the ACM*, 41(8):74--83.

Cunningham, P.; Nowlan, N.; Delany, S. J. & Haahr, M. (2003). A case-based approach to spam filtering that can track concept drift. Em *Proc. of International Conference On Case-Based Reasoning (ICCBR)*.

Drake, C. E.; Oliver, J. J. & Koontz, E. J. (2004). Anatomy of a phishing email. Em *Proc. in Conference on Email and Anti-Spam (CEAS)*.

Drucker, H.; Wu, D. & Vapnik, V. N. (1999). Support vector machines for spam categorization. *IEEE Transactions on Neural networks*, 10(5):1048--1054.

Fallows, D. (2003). How it is hurting e-mail and degrading life on the internet. *Pew Internet & American Life Project*.

Fawcett, T. (2003). "in vivo"spam filtering: A challenge problem for kdd. *SIGKDD Explor. Newsl.*, 5(2):140--148.

Fazzion, E.; Las-Casas, P. H. B.; Fonseca, O.; Guedes, D.; Meira Jr, W.; Hoepers, C.; Steding-Jessen, K. & Chaves, M. H. (2014). Spambands: uma metodologia para identificaç ao de fontes de spam agindo de forma orquestrada. Em *Proc. of Brazilian Symposium on Information and Computational Systems Security (SBSeg)*.

Fonseca, O. H.; Fazzion, E.; Cunha, I.; Las-Casas, P.; Guedes, D.; Meira, W.; Hoepers, C.; Steding-Jessen, K. & Chaves, M. (2016). Measuring, characterizing, and avoiding spam traffic costs. 20(4):16--24.

Gellens, R. & Klensin, J. C. (2011). rfc 6409.

Geng, G. & Hong, B. (2016). Method for detecting phishing website without depending on samples (google patent).

Gray, A. & Haahr, M. (2004). Personalised, collaborative spam filtering. Em *Proc. in Conference on Email and Anti-Spam (CEAS)*.

Hawley, A. E. (1997). Taking spam out of your cyberspace diet: common law applied to bulk unsolicited advertising via electronic mail. *UMKC Law Review*, 66:381.

Hird, S. (2002). Technical solutions for controlling spam. *Proc. of Australian UNIX and Open Systems User Group (AUUG)*.

Hong, J. (2012). The state of phishing attacks. *Communications of the ACM*, 55(1):74--81.

Ioannidis, J. (2003). Fighting spam by encapsulating policy in email addresses. Em *Proc. of Annual Network and Distributed System Security Symposium (NDSS)*.

Jung, J. & Sit, E. (2004). An empirical study of spam traffic and the use of dns black lists. Em *Proc. of Internet Measurement Conference (ACM IMC)*.

Junod, J. (1997). Servers to spam: drop dead. *Computers & Security*, 7(16):623.

Kanich, C.; Kreibich, C.; Levchenko, K.; Enright, B.; Voelker, G. M.; Paxson, V. & Savage, S. (2008). Spamalytics: An empirical analysis of spam marketing conversion. Em *Proc. of Computer and Communications Security (ACM CCS)*.

Kolcz, A.; Chowdhury, A. & Alspector, J. (2004). The impact of feature selection on signature-driven spam detection. Em *Proc. in Conference on Email and Anti-Spam (CEAS)*.

Konte, M.; Perdisci, R. & Feamster, N. (2015). Aswatch: An as reputation system to expose bulletproof hosting ases. *Proc. of Computer Communication Review (ACM CCR)*.

Kreibich, C.; Kanich, C.; Levchenko, K.; Enright, B.; Voelker, G. M.; Paxson, V. & Savage, S. (2008). On the spam campaign trail. Em *Proc. of USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*.

Kreibich, C.; Kanich, C.; Levchenko, K.; Enright, B.; Voelker, G. M.; Paxson, V. & Savage, S. (2009). Spamcraft: An inside look at spam campaign orchestration. Em *Proc. of USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*.

Kumaraguru, P.; Rhee, Y.; Acquisti, A.; Cranor, L. F.; Hong, J. & Nunge, E. (2007). Protecting people from phishing: the design and evaluation of an embedded training email system. Em *Proc. of Conference on Human Factors in Computing Systems (SIGCHI)*.

Lad, M.; Massey, D.; Pei, D.; Wu, Y.; Zhang, B. & Zhang, L. (2006). Phas: A prefix hijack alert system. Em *USENIX Security*.

Las-Casas, P. H. B.; Fonseca, O.; Fazzion, E.; Hoepers, C.; Steding-Jessen, K.; Chaves, M. H.; Cunha, Í.; Meira Jr, W. & Guedes, D. (2016). Uma metodologia para identificaç ao adaptativa e caracterizaç ao de phishing. Em *Proc. of Brazilian Symposium on Computer Networks and Distributed Systems (SBRC)*.

Las-Casas, P. H. B.; Guedes, D.; Almeida, J. M.; Ziviani, A. & Marques-Neto, H. T. (2013a). Spades: Detecting spammers at the source network. *Computer Networks*, 57(2):526--539.

Las-Casas, P. H. B.; Guedes, D.; Jr., W. M.; Hoepers, C.; Steding-Jessen, K.; Chaves, M. H. P.; Fonseca, O.; Fazzion, E. & Moreira, R. E. A. (2013b). Análise do tráfego de spam coletado ao redor do mundo. Em *Proc. of Brazilian Symposium on Computer Networks and Distributed Systems (SBRC)*.

Leiba, B.; Ossher, J.; Rajan, V.; Segal, R. & Wegman, M. N. (2005). Smtp path analysis. Em *Proc. in Conference on Email and Anti-Spam (CEAS)*.

Levy, E. (2003). The making of a spam zombie army. *IEEE Security & Privacy Magazine*, 1(4):58--59.

Li, F. & Hsieh, M.-H. (2006). An empirical study of clustering behavior of spammers and group-based anti-spam strategies. Em *Proc. in Conference on Email and Anti-Spam (CEAS)*.

Ludl, C.; McAllister, S.; Kirda, E. & Kruegel, C. (2007). On the effectiveness of techniques to detect phishing sites. Em *Proc. of Conference on Detection of Intrusions and Malware and Vulnerability Assessment (DIMVA)*.

Lynam, T. R.; Cormack, G. V. & Cheriton, D. R. (2006). On-line spam filter fusion. Em *Proc. in Conference on Research and development in Information Retrieval (SIGIR)*.

Mason, J. (2002). Filtering spam with spamassassin. Em *HEANet Annual Conference*.

Md Shoeb, A. A.; Mukhopadhyay, D.; Al Noor, S.; Sprague, A. & Warner, G. (2015). Spam campaign cluster detection using redirected urls and randomized sub-domains. Academy of Science and Engineering, USA.

Mitchell, T. (1997). Machine learning. McGraw-Hill Boston, MA.

Moura, G. C.; Sadre, R. & Pras, A. (2011). Internet bad neighborhoods: the spam case. Em *Proc. of International Conference on Network and Service Management (CNSM)*.

Neumann, P. G. & Weinstein, L. (1997). Spam, spam, spam! *Communications of the ACM*, 40(6):112--113.

Pantel, P.; Lin, D. et al. (1998). Spamcop: A spam classification & organization program. Em *Proc. of AAAI Workshop on Learning for Text Categorization*.

Pathak, A.; Hu, Y. C. & Mao, Z. M. (2008). Peeking into spammer behavior from a unique vantage point. Em *Proc. of USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*.

Ramachandran, A. & Feamster, N. (2006). Understanding the network-level behavior of spammers. Em *Proc. of Conference on Data Communication (SIGCOMM)*.

Ramachandran, A.; Feamster, N. & Vempala, S. (2007). Filtering spam with behavioral blacklisting. Em *Proc. of Computer and Communications Security (ACM CCS)*.

Ramzan, Z. & Wüest, C. (2007). Phishing attacks: Analyzing trends in 2006. Em *Proc. in Conference on Email and Anti-Spam (CEAS)*.

Rigoutsos, I. & Huynh, T. (2004). Chung-kwei: a pattern-discovery-based system for the automatic identification of unsolicited e-mail messages (SPAM). Em *Proc. in Conference on Email and Anti-Spam (CEAS)*.

Sahami, M.; Dumais, S.; Heckerman, D. & Horvitz, E. (1998). A bayesian approach to filtering junk e-mail. Em *Learning for Text Categorization: Papers from the 1998 workshop*.

Sakkis, G.; Androutsopoulos, I.; Paliouras, G.; Karkaletsis, V.; Spyropoulos, C. D. & Stamatopoulos, P. (2003). A memory-based approach to anti-spam filtering for mailing lists. *Information retrieval*, 6(1):49--73.

Segal, R.; Crawford, J.; Kephart, J. O. & Leiba, B. (2004). Spamguru: An enterprise anti-spam filtering system. Em *Proc. in Conference on Email and Anti-Spam (CEAS)*.

Shimodaira, H. (2014). Text classification using naive bayes. *Learning and Data Note*.

Sipior, J. C.; Ward, B. T. & Bonner, P. G. (2004). Should spam be on the menu? *Communications of the ACM*, 47(6):59--63.

Skoll, D. (2003). Practical methods for combatting spam. *SYS-CON Media*.

Stone-Gross, B.; Holz, T.; Stringhini, G. & Vigna, G. (2011). The underground economy of spam: A botmaster's perspective of coordinating large-scale spam campaigns. Em *Proc. of USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*.

Stone-Gross, B.; Kruegel, C.; Almeroth, K.; Moser, A. & Kirda, E. (2009). Fire: Finding rogue networks. Em *Proc. of Computer Security Applications Conference*.

van der Merwe, A.; Loock, M. & Dabrowski, M. (2005). Characteristics and responsibilities involved in a phishing attack. Em *Proc. of International Symposium on Information and Communication Technologies*.

Van Wanrooij, W. & Pras, A. (2010). Filtering spam from bad neighborhoods. *International Journal of Network Management*, 20(6):433--444.

Wagner, C.; François, J.; State, R.; Dulaunoy, A.; Engel, T. & Massen, G. (2013). Asmatra: Ranking ass providing transit service to malware hosters. Em *International Symposium on Integrated Network Management*.

Wagner, M. (2002). Spam may overtake e-mail in 2003 (cnn news).

Whittaker, C.; Ryner, B. & Nazif, M. (2010). Large-scale automatic classification of phishing pages. Em *Proc. of Annual Network and Distributed System Security Symposium (NDSS)*.

Wittel, G. L. & Wu, S. F. (2004). On attacking statistical spam filters. Em *Proc. in Conference on Email and Anti-Spam (CEAS)*.

Xie, Y.; Yu, F.; Achan, K.; Panigrahy, R.; Hulten, G. & Osipkov, I. (2008). Spamming botnets: signatures and characteristics. *Proc. of Conference on Data Communication (SIGCOMM)*.

Apéndices

Apêndice A

Estudos de casos de SpamBands

A.1 SpamBand HTTP/SOCKS: Venda de produtos farmacêuticos

Encontramos diversos *SpamBands* HTTP/SOCKS durante o período analisado e trazemos um exemplo de *SpamBand* que tentou enviar mensagens em chinês através de um *honeypot* instalado no Brasil (BR-02) no dia 2016-04-01. Identificamos que a estrutura desse *SpamBand* é uma clique (coeficiente de agrupamento máximo). Nós identificamos que esse *SpamBand* tenta conectar a máquinas de 6 ASes distintos para enviar cerca de 92% de suas mensagens: AS 3462 (HINET), AS 8075 (Microsoft), AS 24506 (Yahoo-TP2), AS 10229 (Yahoo-TW1), AS 15169 (Google) e AS 4780 (SEEDNET). As outras conexões HTTP/SOCKS para o envio do restante das mensagens (8%) são endereçadas para 176 ASes distintos.

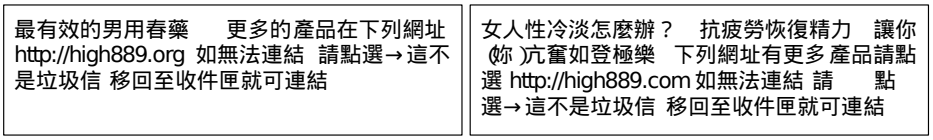


Figura A.1: Exemplo de mensagens de campanhas em chinês enviadas por um *SpamBand* HTTP.

A figura A.1 mostra dois exemplos de mensagens enviadas pela maioria dos endereços IP do *SpamBand* em questão. Essas mensagens têm uma estrutura similar e buscam a venda de produtos farmacêuticos como o viagra, sendo claramente um *SpamBand* de propaganda. O uso dos protocolos HTTP/SOCKS para realizar conexões a outros servidores SMTP indicam que o *spammer* pode estar utilizando uma estrutura

dedicada ao envio. Esse fato é reforçado pela concentração dos endereços IP em um único sistema autônomo (AS 3462), conforme mostrado pela tabela A.1, que pode estar sendo conivente com o abuso do sistema de email. Esse caso mostra como a origem do tráfego enxergado pelos servidores finais de email podem ser distorcidas uma vez que apesar do tráfego ter origem em Taiwan, os servidores finais enxergariam esse tráfego (se o *honeypot* repassasse as mensagens) como originado no Brasil.

Tabela A.1: Atributos do *SpamBand* de venda de produtos farmacêuticos.

ATRIBUTO	VALOR
# mensagens	30.246
# endereços IP distintos	22
# campanhas	18
ASes	3462
Country Codes	TW
Protocolos (# IP)	HTTP (22)
Blacklist (# IPS)	PBL (22)
Idioma das mensagens	chinês
Coef. Agrupamento	1,0

A.2 SpamBand SMTP: Furto de contas da Apple

Nossas análises no *honeypot* instalado no Reino Unido (GB-01) mostram que esse *honeypot* é alvo de pequenos *SpamBands* que utilizam apenas o protocolo SMTP e que são especializados no envio de *phishing*. Nesta seção apresentamos um *SpamBand*, formado por apenas dois endereços IP, que enviam diferentes mensagens para tentar furtar informações de contas na Apple. A figura A.2 mostra dois exemplos dessas mensagens.

Como pode-se observar, as mensagens possuem um texto diferente mas as estruturas são bastante similares: o phisher introduz um argumento para requisitar a informação, como manutenção ou revisão de contas, e interage com o usuário através do link. Pela tabela A.2, observa-se que esses dois endereços IP se localizam no AS 8075 (Microsoft). Por se tratar de uma rede de uma corporação conhecida mundialmente, analisamos os endereços IP através do comando "dig -x" e encontramos que ambos endereços possuem como autoridade de DNS o servidor "prd1.azuredns-cloud.net". Isso sugere que *spammers/phishers* estão alugando serviços de nuvem para o envio de *spam*

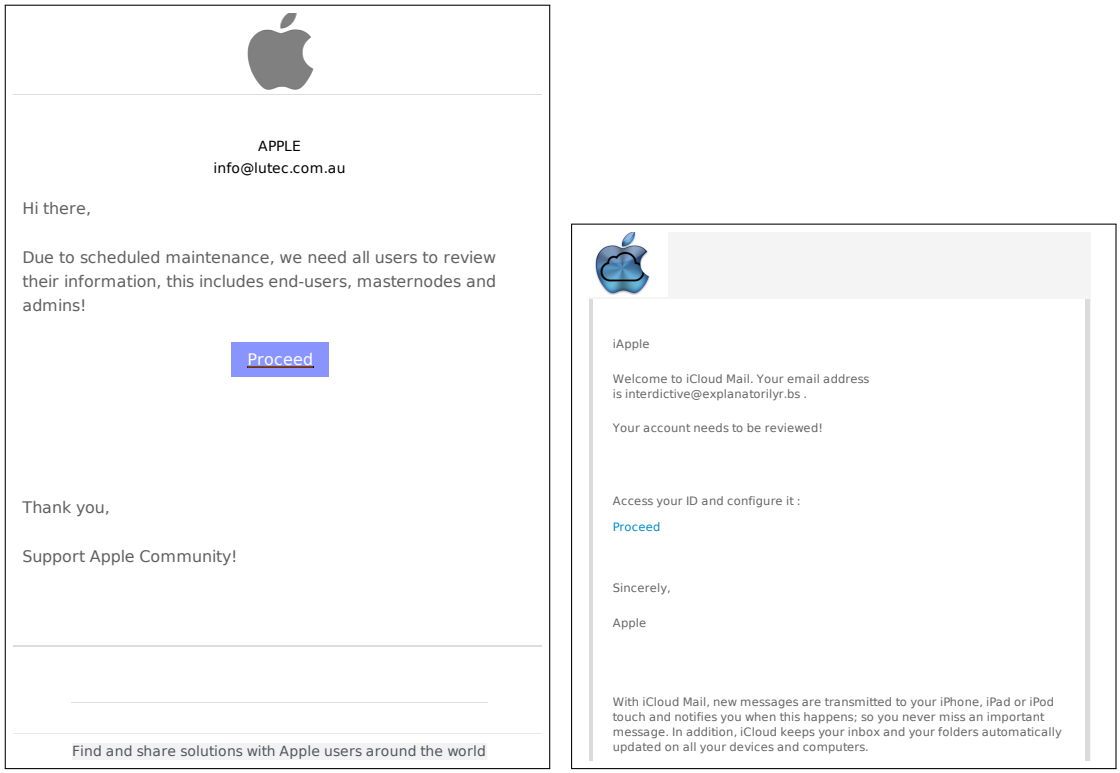


Figura A.2: Exemplo de mensagens de campanhas enviadas com intuito de furtar informações de contas da Apple.

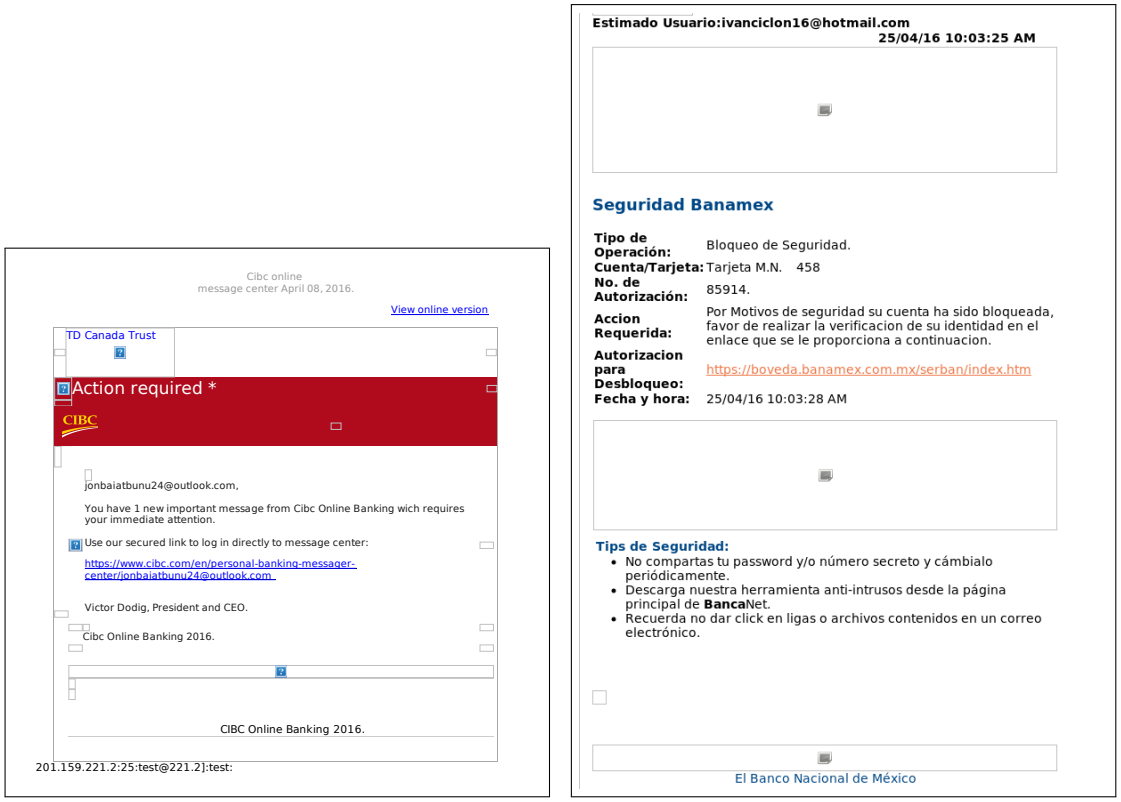
e, por isso, não utilizam protocolos HTTP/SOCKS para esconder suas identidades e abusam diretamente de *relays* abertos.

Tabela A.2: Atributos do *SpamBand* de furto de contas da Apple.

Atributo	Valor
# mensagens	47.842
# endereços IP distintos	2
# campanhas	21
ASes	8075
Country Codes	US
Protocolos (# IP)	SMTP (2)
Blacklist (# IPS)	nenhuma
Idioma das mensagens	inglês

A.3 SpamBands especializados em phishing para bancos

Nessa seção, discutimos exemplos de *SpamBands* de *phishings* relacionados a bancos de diferentes países. Em todos eles encontramos *SpamBands* com endereços IP que utilizam apenas o protocolo SMTP, com exceção do banco mexicano que participa de um *SpamBand* híbrido. Discutimos as caracteríscas dos *SpamBands* de cada banco a seguir.

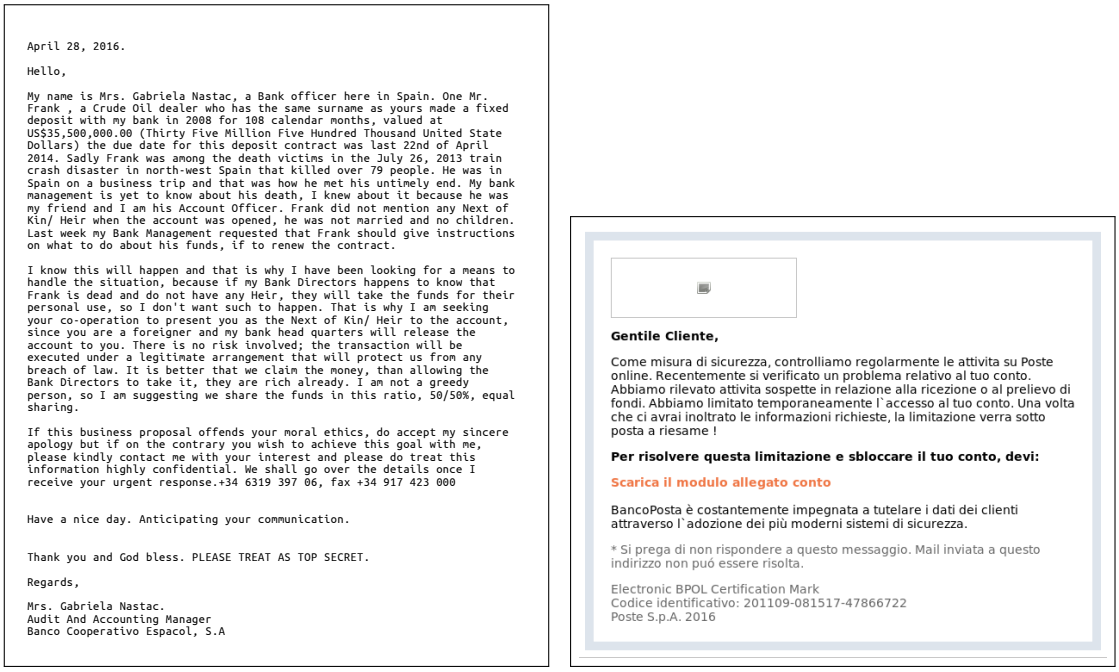


(a) Banco CBIC - Canadá **(b)** Banco Nacional de México - México

Figura A.3: Exemplo de mensagens de campanhas de *phishing* para bancos canadenses e mexicanos.

Banco CBIC: O *SpamBand* que envia a mensagem do banco CBIC da figura A.3 é composto de apenas um endereço IP, localizado no AS 197226 (SPRINT - SDC) e country code polonês. Esse *SpamBand* possui uma única campanha que são as mensagens do banco CBIC e os principais destinos dessas mensagens foram para os domínios "outlook.com" e "hotmail.com". O *honeypot* que encontramos esse *SpamBand* está localizado no Equador (EC-02) e foi encontrado no dia 8 de abril de 2016.

Banco Nacional de México: O *SpamBand* que envia as mensagens do Banco Nacional de México da figura A.3 é híbrido, formado por 139 endereços IP distintos que enviam 160.059 mensagens e composto de diversos idiomas (russo, italiano, espanhol, inglês). Encontramos que a campanha do banco em questão partiu de apenas um endereço IP desse *SpamBand* envolvido em 23 outras campanhas. Esse endereço IP, que utiliza SMTP para o envio, está localizado no AS 197226 (SPRINT - SDC) com *country code* polonês e não está listado nas blacklists XBL ou PBL. Esse *SpamBand* foi localizado no dia 25 de abril de 2016 em um dos *honeypots* instalados no EUA (US-03).



April 28, 2016.

Hello,

My name is Mrs. Gabriela Nastac, a Bank officer here in Spain. One Mr. Frank , a Crude Oil dealer who has the same surname as yours made a fixed deposit with my bank in 2008 for 108 calendar months, valued at US$35,500,000.00 (Thirty Five Million Five Hundred Thousand United State Dollars) the due date for this deposit contract was last 22nd of April 2014. Sadly Frank was among the death victims in the July 26, 2013 train crash disaster in north-west Spain that killed over 79 people. He was in Spain on a business trip and that was how he met his untimely end. My bank management is yet to know about his death, I knew about it because he was my friend and I am his Account Officer. Frank did not mention any Next of Kin/ Heir when the account was opened, he was not married and no children. Last week my Bank Management requested that Frank should give instructions on what to do about his funds, if to renew the contract.

I know this will happen and that is why I have been looking for a means to handle the situation, because if my Bank Directors happens to know that Frank is dead and do not have any Heir, they will take the funds for their personal use, so I don't want such to happen. That is why I am seeking your co-operation to present you as the Next of Kin/ Heir to the account, since you are a foreigner and my bank head quarters will release the account to you. There is no risk involved; the transaction will be executed under a legitimate arrangement that will protect us from any breach of law. It is better that we claim the money, than allowing the Bank Directors to take it, they are rich already. I am not a greedy person, so I am suggesting we share the funds in this ratio, 50/50%, equal sharing.

If this business proposal offends your moral ethics, do accept my sincere apology but if on the contrary you wish to achieve this goal with me, please kindly contact me with your interest and please do treat this information highly confidential. We shall go over the details once I receive your urgent response.+34 6319 397 06, fax +34 917 423 000

Have a nice day. Anticipating your communication.

Thank you and God bless. PLEASE TREAT AS TOP SECRET.

Regards,

Mrs. Gabriela Nastac.
Audit And Accounting Manager
Banco Cooperativo Espacol, S.A

(a) Banco Cooperativo Espanhol - Espanha **(b)** BancoPosta - Itália

Figura A.4: Exemplo de mensagens de campanhas de *phishing* para bancos espanhóis e italianos.

Banco Cooperativo Espanhol: O *SpamBand* envolvido no envio das mensagens do Banco Cooperativo Espanhol da figura A.4 foi encontrado no *honeypot* US-03 no dia 28 de abril de 2016. Esse SpamBand é composto de apenas um endereço IP, localizado no AS 12876 (ONLINE S.A.S.) com *country code* na França. Esse endereço IP utiliza o protocolo SMTP para o envio da única campanha desse *SpamBand*, composta de 5.627 mensagens. Ressaltamos que o endereço IP está listado na XBL e que observamos 28.464 domínios diferentes para onde essas mensagens foram enviadas.

BancoPosta: O *SpamBand* envolvido no envio da mensagem do BancoPosta da figura A.4 foi encontrado no *honeypot* US-03 no dia 30 de abril de 2016. Esse *SpamBand*,

assim como o banco espanhol, é composto de apenas um endereço IP que envia SMTP e participa apenas de uma campanha. Porém, esse endereço IP possui o *country code* dos EUA e foi localizado no AS 7922 (COMCAST). Além disso, esse endereço não está listado na XBL e envia as 6.794 mensagens do *SpamBand* para 15 domínios diferentes, todos com ccTLD ".it".

(a) Bank of America - EUA **(b)** Banco do Brasil - Brasil

Figura A.5: Exemplo de mensagens de campanhas de *phishing* para bancos brasileiros e americanos.

Bank of America: O *SpamBand* envolvido na mensagem do Bank of America da figura A.5 foi detectado no *honeypot* US-02 no dia 18 de maio de 2016. Observamos que o *SpamBand* é composto de apenas um endereço IP localizado no AS 3215 (Orange S.A.) com *country code* francês. Esse endereço IP tenta enviar suas mensagens através do protocolo SMTP e não está listado na XBL. Encontramos que todas as 17.005 mensagens que o *SpamBand* envia para 1.131 domínios diferentes são phishings do Bank of America.

Banco do Brasil: O *SpamBand* envolvido na mensagem do Banco do Brasil da figura A.5 foi detectado no *honeypot* US-03 no dia 3 de abril de 2016. Observamos que esse *SpamBand* possui uma única campanha (a do Banco do Brasil) e é composto de apenas dois endereços IP localizado no AS 8075 (Microsoft) que exploram o *honeypot* como *proxy* aberto (SMTP) e não estão listados na XBL. Esses endereços IP, entretanto, enviam apenas uma mensagem cada com o domínio do *honeypot* que indica que sejam mensagens com o intuito de testar o *honeypot*. Interessante mencionar que não encontramos esses endereços IP em nenhum outro dia em todos os *honeypot* durante o período analisado.

Apêndice B

Sistema desenvolvido para o CERT.br

Esta dissertação culminou no desenvolvimento de um sistema em Python para tratamento de mensagens de *spam* para o CERT.br. Esse tratamento envolve a geração de campanhas de *spam* e dos *SpamBands*. A figura B.1 mostra o funcionamento do sistema. O código principal do sistema está no "koloth.py" que faz a leitura das configurações básicas do sistema, como os *honeypots* a serem processados e a localização dos *mailboxes*, e faz a chamada do código "dayprocess.py" que realiza o processamento de um *honeypot* em um dia. As configurações iniciais determinam o número máximo de chamadas "dayprocess.py" que podem permanecer ativas ao mesmo tempo. Os dias a serem processados para cada *honeypot* são gerados pelo próprio sistema a partir da data inicial padrão no arquivo de configuração e a data mais recente dos *mailboxes* de um *honeypot* no sistema. O sistema faz uma verificação para determinar quais dias já foram processados por cada *honeypot*, realizando apenas o processamento dos dias que ainda não foram processados no intervalo. É possível determinar o intervalo manualmente através do sinalizador "-d <início> <fim>" onde as datas são no formato "yyyy-mm-dd". O código contido no "koloth.py" está registrado no crontab diário para manter as campanhas e *SpamBands* atualizados.

As configurações do sistema estão em um módulo chamado "config". Esse módulo possui alguns arquivos de configuração importantes:

- **config.txt:** configurações essenciais de diretórios como diretório de saída e caminho para os *logfiles* e *mailboxes*.
- **process.txt:** configurações essenciais do processo como número máximo de processos ativos simultaneamente, limiar mínimo de mensagens em uma campanha

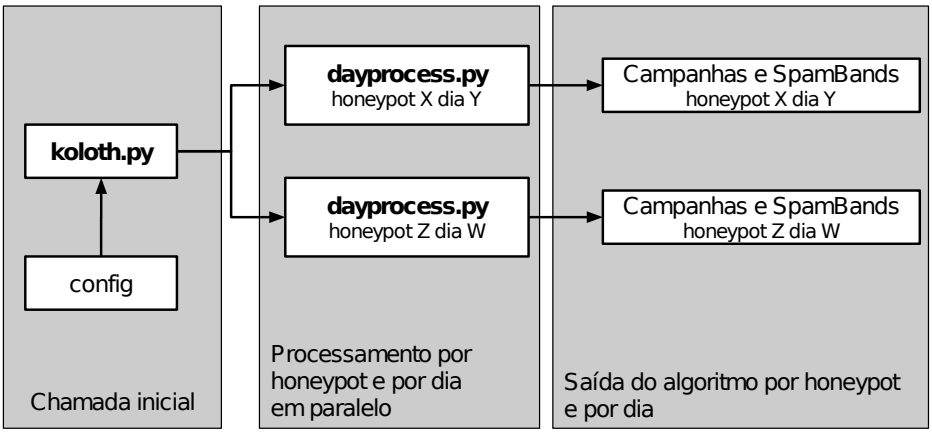


Figura B.1: Esquema do sistema desenvolvido para o CERT.br.

e os limiares do algoritmo de detecção de *SpamBands*.

- **spam.txt e phishing.txt:** exemplos de *spam* e *phishing* utilizados pelo algoritmo para diferenciar mensagens de propaganda de mensagens de *phishing*.

- **targs.txt:** lista de *honeypots* considerados para o processamento.

O sistema possui dependência das seguintes bibliotecas em Python: networkx, matplotlib, langdetect e textblob. Além do módulo "config", o sistema é dividido em outros quatorze módulos que descrevemos a seguir:

- **campaigns:** Módulo que comporta os algoritmos para a detecção de campanhas na FP-Tree.

- **config:** Módulo que comporta as configurações iniciais do sistema como diretórios de leitura e de escrita, *honeypots* a serem processados, mensagens para treinamento do algoritmo de detecção de *phishing*, *stopwords*, etc.

- **examples:** Módulo que comporta os algoritmos necessários para extração de um exemplo por campanha detectada.

- **features:** Módulo que comporta as definições dos atributos considerados para a construção da FP-Tree e os algoritmos para a extração desses atributos dos *mailboxes*.

- **fptree:** Módulo que comporta o algoritmo para construção da árvore de padrões frequentes.

- **log:** Módulo que comporta algoritmos que tratam da geração de relatórios de execução do sistema.
- **logfile:** Módulo que comporta algoritmos para extração de informação dos *logfiles* das mensagens de spam.
- **mapping:** Módulo que comporta algoritmos para realizar a compressão dos atributos extraídos em memória primária.
- **mailbox:** Módulo que comporta algoritmos para a leitura dos *mboxes* e extração de atributos pré-definidos.
- **messages:** Módulo que comporta algoritmos para extração de conteúdo específico do corpo das mensagens como URLs e idioma.
- **process:** Módulo que comporta algoritmos de leitura de informações necessárias a execução do sistema.
- **rsync:** Módulo que comporta algoritmos para determinar a data de sincronização dos *honeypots*.
- **spambands:** Módulo que comporta algoritmos para detecção dos *SpamBands*.
- **statistics:** Módulo que comporta algoritmos que geram relatórios informativos diários da execução do sistema.
- **utils:** Módulo que comporta rotinas de execução como definições para leitura de arquivos específicos, definições do formato de hora, inicialização de algoritmos, etc.

As informações dos *SpamBands* envolvem as conexões do grafo e as informações agregadas como número de mensagens enviadas, protocolos utilizados, informações sobre *blacklists*, domínios dos destinatários e sistemas autônomos envolvidos. As informações das campanhas envolvem a assinatura e as mensagens pertencentes a cada campanha além das informações individuais das mensagens como idioma, URLs, localização no banco de dados de *mailboxes* e uma sumarização contendo um exemplo de mensagem de cada campanha no dia.