Detecção de Phishing utilizando Maching Learnig

Matheus Silva Martins Mota, *Estudante de Graduação, ITA,* Reuben Solomon Katz, *Estudante de Graduação, ITA*

***Abstract*—A crescente demanda de produtos digitalizados possibilitando um sociedade desenvolvida com facilidades web, como cartões virtuais, sites com promoções, aplicações de usando chats, entre outras, também é prejudicial para esta sociedade, pois possibilita o desenvolvimento de táticas de roubo e fraude, como o phishing.**

O phishing é a tentativa fraudulenta de obter informações ou dados confidenciais, como nome de usuários, senhas, números do cartão de crédito, disfarçar-se como uma entidade confidencial em uma comunicação eletrônica.

Apesar das crescentes técnicas de prevenção, principalmente de engenheiros e pesquisadores da área de defesa digital, o phishing continua sendo uma ameaça importante, pois as principais contramedidas em uso ainda são baseadas na lista negra de URLs reativas, que são criadas, principalmente, por denúncias de grandes empresas e iniciativas independentes.

Essa técnica é ineficiente devido à curta vida útil do phishing. Fazer uma abordagem baseada em tempo real de detecção de phishing é mais apropriado, pois tornaria a detecção de phishing mais automatizada.

Neste artigo, apresentamos um melhoramento do PhishingStorm, baseado no artigo PhishStorm: Detecting Phishing with Streaming Analytics.

Neste artigo, apresentamos o PhishStorm, um phishing automatizado de detecção que pode analisar em tempo real qualquer URL para identificar possíveis sites de phishing. O PhishStorm pode interagir com qualquer servidor de email ou proxy HTTP. Argumentamos que os URLs de phishing geralmente têm poucas relações entre a parte da URL que deve ser registrado (domínio de baixo nível) e a parte restante da URL (domínio de nível superior, caminho, consulta). Mostramos neste artigo que evidências experimentais apoiam essa observação e pode ser usado para detectar sites de phishing.

Para esse fim, definimos o novo conceito de relacionamento intra-URL e avalie-o usando recursos extraídos de palavras que compõem um URL com base em consultar dados dos mecanismos de pesquisa do Google. Esses recursos são então usados na classificação baseada em aprendizado de máquina para detectar URLs de phishing de um conjunto de dados real. Nossa técnica é avaliada em 800 URLs legítimos de phishing. Discutimos no artigo padrões de implementação eficientes que permitem análises em tempo real usando arquiteturas de Big Data, como STORM.

***Index Terms*—Detecção de Phishing, Classificação de URL, Big Data, STORM**

I. Introdução

O phishing é um dos métodos de ataque mais antigos, já que o uso da engenharia social corresponde à metade do trabalho para obter informações confidenciais. Além disso, ele é um dos mais lucrativos. Em 2003, Anti-Phishing Working Group (APWG) divulgou um relatório que aponta que cerca de US$1,2 bilhão em cripto moedas já foi roubado utilizando técnicas de phishing.

Várias técnicas são usadas para realizar ataques de phising, variando de subterfúgios técnicos (envenenamento de cache DNS, falsificação de e-mail, aquisição de servidores da Web) até engenharia social.

Além disso, vários objetivos são buscados: dados, dinheiro ou credenciais, roubo de sites falsos, download de malware, apesar dessa diversidade, um recurso comum é o uso de URLs ofuscadas para redirecionar usuários para sites falso ou downloads drive-by.

Atrair usuários da Internet fazendo-os clicar em links maliciosos que parecem confiáveis é uma tarefa fácil por causa da disseminação credulidade e desconhecimento. Para lidar com essa ameaça a melhor estratégia é impedir a conexão com sites de phishing por identificação de URLs de phishing. Outras técnicas como derrubar sites de phishing provou ser difícil e ineficiente devido principalmente à curta vida útil do site e o uso do fluxo DNS. Web de phishing. A vida útil curta do local torna o prolongado processo de reação da lista negra baseada em relatórios de usuários ineficientes. Além disso, o uso de diferentes variações nos URLs para o mesmo phishing complica a tarefa da lista negra, pois as listas negras devem fornecer uma combinação perfeita para uma URL. Portanto, em tempo real a detecção maliciosa de URL é uma técnica melhor para derrotar o phishing.

Neste artigo, propomos um URL automatizado em tempo real para sistema de classificação de phishing para proteger os usuários contra phishing. O método subjacente tem como objetivo a identificação de URLs de phishing que se baseiam em domínios registrados (maliciosos ou não) que não estão relacionados aos seus destinos marca. Para iludir suas vítimas, os phishers misturam muitas palavras-chave (marca famosa, palavras atraentes) no restante partes do URL. A maioria dos usuários da Internet não está ciente da hierarquia de DNS. Ver palavras como paypal, ebay ou visa em qualquer nível de um URL fará com que se sintam confiantes de que o link do invasor leva ao site oficial dessas marcas.

A partir da observação de URLs de phishing, afirmamos que existem poucas relações entre o domínio registrado e o resto da URL. No entanto, as palavras que compõem o resto da URL (domínio de baixo nível, caminho, consulta) geralmente tem muitas inter-relações. Portanto, nossa abordagem avalia a relação de palavras que compõem uma URL e destaca as diferenças entre URLs legítimos e de phishing. Anteriormente existente soluções que não são adequadas para avaliar semelhança ou parentesco de palavras para o vocabulário da Internet. Essas ferramentas, provenientes do processamento de linguagem natural. Geralmente, não há entradas para nomes de domínio e a maioria dos palavras que compõem um URL.

Matheus Silva Martins Mota Aluno de Graduação do Instituto Tecnológico de Aeronáutica 2018, São José dos Campos, SP, 1228-462, Brasil e-mail: matheussilvamartins1714@gmail.com.

Reuben Solomon Katz Aluno de Graduação do instituto Tecnológico de Aeronáutica since 2018, São José dos Campos, SP, 1228-462, Brasil e-mail: reubenskatz@gmail.com.

Aproveitamos a consulta do mecanismo de pesquisa dados do Google para calcular essa relação. Com base nisso, definimos o termo de relacionamento intra-URL. Métodos de computação de recursos eficientes, aproveitando o são utilizadas técnicas de análise de streaming e estrutura de dados com espaço eficiente. Isso reduz o atraso na detecção de phishing URLs e permitir aplicativos mais amplos, como email de phishing ou HTTP filtragem de tráfego. Extraímos 12 recursos de um único URL que são introduzidos em algoritmos de aprendizado de máquina para identificar URLs de phishing. Nossa técnica é avaliada com base na verdade 800 URLs. Por fim, uma pontuação de phishing é calculada para cada URL único com base no classificador Random Forest. Para resumir as principais contribuições deste artigo:

- Introduzimos o conceito de relacionamento intra-URL que descreve o relacionamento entre um domínio registrado e o palavras que compõem o restante de um URL.

- Aproveitamos os dados de consulta do mecanismo de pesquisa para estabelecer relações entre palavras e mostramos que isso é mais adequado vocabulário da Internet do que os métodos existentes.

- Propomos novos recursos com base na relação entre URLs e construir uma abordagem baseada em aprendizado de máquina para distinguir entre URLs de phishing e não phishing. Este artigo é uma extensão de PhishStorm: Detecting Phishing with Streaming Analytics com os seguintes contribuições adicionais:

- Verificação de phishing com método de classificação de Naive Bayes reduzir o atraso nos recursos de relacionamento intra-URL

- Um processo detalhado de engenharia de recursos é executado.

II. Técnicas de Phishing

Os phishers geralmente tentam atrair suas vítimas para clicar em URLs não autorizados que apontam para sites de phishing ou downloads drive-by. Diferentes técnicas de ofuscação de URL são usadas com o objetivo de ocultando o host real e, principalmente, o domínio registrado, o apenas parte do URL que não pode ser definido livremente. Se alguém quiser usar um domínio mydomain.tld e derivar vários URLs dele: url1.meudominio.tld, url2.meudominio.tld / arquivo, ele primeiro registra o domínio mydomain.tld em um domínio registrador, garantindo que ele não possa ser registrado por ninguém outro. Supondo que um phisher queira prender os usuários do PayPal, ele deve use um domain.tld que não seja paypal.com, pois esse domínio é já registrado pelo PayPal Inc. O phisher deve se registrar um nome de domínio mydomain.tld e tente enganar as pessoas misturando rótulos como paypal no restante do URL: login.meudominio.tld / paypal. Um domínio registrado consiste em duas partes: um nível principal domínio e um sufixo público. Um sufixo público (ou ps) é um domínio sufixo de nome sob o qual um usuário da Internet pode registrar um nome. Pode ser apenas um domínio de nível superior, como .com, .org ou um combinação de domínios de nível como .co.uk ou .blogspot.com. Um domínio de nível principal (ou mld) é o domínio de nível que precede um sufixo público. Um domínio registrado é então: mld.ps. Por exemplo em www.paypal.com/login, com é o ps e paypal é o mld. As diferentes técnicas de ofuscação consistem em misturar o nome do domínio original ou as palavras-chave de phishing a parte restante do URL. Essas palavras-chave são geralmente a marca direcionada, serviços relacionados da marca e outros palavras atraentes, como seguro, login, proteção, etc. Suponha que um URL formado por um nome de host com diferentes domínio de nível (ld), um caminho (caminho) e uma consulta (chave = valor): http://5ld.4ld.3ld.mld.ps/path1/path2/path3?key1=value1 chave2 = valor2. A ofuscação geralmente consiste em misturar palavras-chave no caminho, na consulta e no domínio de baixo nível do nome do host (5ld.4ld.3ld). A seguir, apresentamos o técnicas de ofuscação de URL mais usadas:

- Tipo I: ofuscação de URL com outro domínio: o mld.ps é um nome de domínio real, geralmente registrado pelo phisher, enquanto o domínio original sendo phishing faz parte do caminho, o consultor ou o domínio de nível superior.
- Tipo II: ofuscação de URL com palavras-chave: Novamente, o mld.ps é um nome de domínio real, e a marca sendo phishing e palavras relacionadas fazem parte do caminho, da consulta ou do domínio de nível superior.
- Tipo III: Typosquatting ou domínios longos: o mld.ps da URL está sendo o domínio. phishing, mas com erros ortográficos, com letras ou palavras ausentes ou adicionadas, ou o domínio é pronunciado da mesma maneira que o original, mas escrito de forma diferente. O canal da marca direcionada também pode ser combinado com outras palavras para criar um domínio não registrado.
- Tipo IV: ofuscação de URL com endereço IP: o nome do host da URL é substituído por um endereço IP e o phishing da marca faz parte do caminho ou da consulta.
- Tipo V: Ofuscação com encurtador de URL: um serviço de encurtamento de URL é usado para ocultar o nome do host real. Esses URLs não têm significado e são usados principalmente em serviços de segmentação por ataques de phishing que usam esse tipo de URL curto, como o Twitter.

Focamos na identificação dos quatro primeiros tipos de URL técnicas de ofuscação, já que nossa técnica depende de recursos naturais processamento de linguagem, que claramente não é adequado para URLs. O recurso comum desses URLs ofuscados é que a marca e alguns termos relacionados estão incluídos no caminho, a consulta e o domínio de baixo nível. Estes termos estão relacionados como estes têm relacionamentos com a marca alvo e não têm relação óbvia com o mld.ps usado para phishing. este é o oposto do que acontece para um URL legítimo, em que todas as partes do URL estão normalmente relacionadas. Para revelar isso diferença uma análise de parentesco da parte diferente de uma URL é realizado.

III. Análise da relação intraURI

Para a análise da relação entre URIs é necessário quantificar uma relação entre os termos do URI. Se analisarmos uma URI, podemos detectar quatro componentes principais:

- Esquema (HTTP, HTTPS, ftp, irc)
- Domínio
- *Path*
- *Query*

Esquema	Domínio	*Path*	*Query*
https	mpm.payback.de	redi	lid=5927652727930224970
https	ductportails.firebaseapp.com		

TABLE I
TABELA DE EXEMPLO DE URI E SUA DIVISÃO.

No nosso caso, usamos apenas esquemas do tipo HTTP e HTTPS, os quais tem mais chance de dar ataque de phishing por serem mais adeptos à engenharia social e darem mais segurnaça ao usuário, dessa forma, analisamos apenas URLs(URIs com o esquema HTTP ou HTTPS).

As partes que são mais adequados para a nossa análise são o domínio e o path, o maior motivo disso é que a *query* é um parâmetro passado pelo usuário do site (em geral), portanto, não fazendo parte da engenharia social relacionado ao ataque de phishing. Além disso, o esquema também não é adequado para análise por não estar vinculado à engenharia social.

O domínio pode ser dividido em duas partes, o domínio principal(*e.g. mpm.payback*) e o domínio de topo(*e.g. .de*).

A. Extração das palavras dos URLs

Inicialmente, separou-se as URLs como especificado nos componentes em I . Em seguinte, separou-se o o domínio em domínio principal (referenciado como *dp*) e domínio de topo (referenciado como *dt*). Após essa separação, separou-se o *dp* em palavras legíveis (*e.g. duct, portails, firebase, app*) por meio da lei de zipf com um dicionário de custo de palavras, removeu se '_', *www.*, '.' entre palavras, '-' entre palavras e por fim, números - por não estarem relacionados ao nosso conceito de relação intraURL.

A lei de zipf é uma lei empírica o qual formula de maneira matemática-estatística que a distribuição de dados pode ser aproximado pela distribuição zipfiana.

Após isso, pegou-se cada palavra do path e repetiu-se o mesmo processo que se fez com o *dp*, obtendo assim várias palavras para o domínio e várias para o path. Os resultados dessa lei se mostram satisfatórios com a necessidade adiquirda, tendo problema apenas com palavras que não são da língua inglesa ou com siglas. Tome como exemplo o URL *https://kuchkhasbate.info/wp-includes/SimplePie/HTTP/signin/myaccount/success*), ao se dividir obtém se *ku, ch, khas, bate*(a palavra original é do hindu, sendo escrito como *kuch khas bate*) do *dp*, *my, wp, simple, success, sign, in, pie, account, includes, https* do *path*.

Afim de utilizar a relação intraURL vamos definir dois conjuntos:

1) DP_{url} (lê-se domínio principal da URL)
2) RE_{url} (lê-se resto da URL)

Por exemplo, o URL *http://52.170.238.217/prime/mobile* tem como resultado:

- DP_{url} = {52.170.238.217}
- RE_{url} = {prime, mobile}

B. Ferramentas de palavras relacionadas

Perceba que a palavra mobile tem como palavras relacionadas XXX. Perceba que, como explicado anteriormente,phishing é uma técnica de engenharia social, portanto, utiliza a 'linguagem da internet' para enganar usuários a entrar em seu site. Levando isso em conta definimos dois novos conjuntos denominados REL_{DP} (lê-se relacionados ao domínio principal) e REL_{RE} (lê-se relacionado ao resto da URL). *E.g.*, o *dp* da URL no exemplo, após os filtros pode ser definido como $DP_{url} = \phi$ tendo, portanto, $REL_{DP} = \phi$. Quanto ao conjunto RE_{url} = {prime, mobile}, temos REL_{RE} = {altice, mobile, iphone, cod, mario, kart, amazon, prime, twitch}, o que fornece $RE_{url} \cap REL_{RE}$ = {prime, mobile} .

Para determinar as palavras relacionadas, utilizou-se ferramentas de pesquisa. Perceba que, a melhor forma de encontrar palavras no 'dicionário da internet' é utilizando ferramentas de pesquisas de mostram pesquisar relacionadas. Nesse caso, a ferramenta mais adequada é o **Google Trends**.

O **Google Trends** é um site da Google o qual analisa a popularidade de pesquisas na ferramenta *Google Search* (ferramenta da própria empresa). Uma das funcionalidades dessa ferramenta é a de pesquisas relacionadas à pesquisa real, levando isso em conta, é possível encontrar os conjuntos RE_{url} e DP_{url}.

Atualmente, a empresa *Google* não criou uma API oficial para a análise de dados, portanto, utilzou-se um rastreador web (*crawler*) para navegar a página e obter as pesquisas relacionadas.

IV. Análise dos Resultados

A. Forma de Quantificação

Existem várias forma de quantificar conjuntos, a principal é o índice de Jaccard. Define-se como índice de Jaccard como:

$$J = \frac{|A \cap B|}{|A \cup B|}$$

Para isso, encontra-se seis tipos de índice de Jaccard:

Perceba que cada índice de Jaccard é mais expressivo para cada tipo de técnica de phsihing. O índice $J_{REL_{RE} \cap RE}$ é muito útil para técnicas de phishing do tipo 4 e tipo 2, $J_{REL_{RE} \cap RE}$ para a técnica 1, $J_{REL_{DP} \cap DP}$ para a técnica 3, enquanto os índices não citados são úteis para todos os tipos exceto o tipo 4.

$J_{REL_{RE} \cap RE}$	Índice de Jaccard para REL_{RE} e RE
$J_{REL_{RE} \cap REL_{DP}}$	Índice de Jaccard para REL_{RE} e REL_{DP}
$J_{REL_{RE} \cap DP}$	Índice de Jaccard para REL_{RE} e DP
$J_{REL_{DP} \cap RE}$	Índice de Jaccard para RE e REL_{DP}
$J_{REL_{DP} \cap DP}$	Índice de Jaccard para DP e REL_{DP}
$J_{DP \cap RE}$	Índice de Jaccard para RE e DP

TABLE II
TABELA DE ÍNDICES DE JACCARD.

B. Classificação de URL e Discussão

Perceba que nosso problema é de classificação supervisionada. Esse tipo de problema possui uma variedade de tipos de algoritmos para sua resolução, dessa forma decidiu utilizar-se as seguintes:

- Randmon Forest
- SVM
- Naive Bayes

A vantagem do Randmon Forest é a sua alta acurácia em relação aos outros algoritmos atuais, além de conseguir lidar com vários inputs e ter fácil acesso ao funcionamento dele e os motivos de classificação. No nosso trabalho, esse obteve o melhor resultado.

Logo em seguida, Naive Bayes tem o segundo melhor resultado, isso resulta do fato de estarmos utilizando um dataset pequeno. Além disso, tem a vantagem de convergir rapidamente ser a condição de independência de Naive Bayes for verdadeira.

Quanto ao SVM , percebe-se que esse possui grande vantagem ao se discutir sobre independência de inputs e ser eficiente na memória, entretanto não possui nenhuma vantagem específica para esse problema.

Random Forest		
Acurácia	Falso Negativo	Falso Positivo
57%	12.5%	33%

TABLE III
TABELA DE RESULTADOS.

SVM		
Acurácia	Falso Negativo	Falso Positivo
48.5 %	100 %	0 %

TABLE IV
TABELA DE RESULTADOS.

Naive Bayes		
Acurácia	Falso Negativo	Falso Positivo
48.5%	83.1%	18.3%

TABLE V
TABELA DE RESULTADOS.

V. CONCLUSÃO

Perceba que os nossos resultados destacam-se pela pela baia acurácia, e alta taixa de falso negativo para SVM e Naive Bayes. Isso se deve pelo fato de de termos apenas cerca de 300 *links* no nosso dataset.

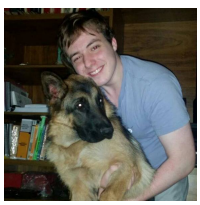
AGRADECIMENTOS

Gostaríamos de agradecer ao Professor Doutor Felipe Verri pelas suas aulas os quais foram de extrema importância para o nosso desenvolvimento na área e também aos outros atendentes das aulas tomadas no curso de CE-299 por compartilhar a experiência com a gente.

REFERENCES

[1] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.
[2] K. Nigam, A. McCallum, S. Thrum and T Mitchell, *Learning to Classify Text from Labeled and Unlabeled Documents*, Technical Report Carnegie Mellon. =0
[3] K. P. Bennett, C. Campbell *Support Vector Machines: Hype or Hallelujah?*, Technical Report Bristol University. =0
[4] G. Kirby *ZIPF'S LAW* . =0
[5] M. S. Granovetter, "The strength of weak ties," The American Journal of Sociology, vol. 78, no. 6, pp. 1360–1380, 1973.

Matheus Silva Martins Mota Entrou no Instituto Tecnológico da Aeronáutica em 2018. Fez esse artigo como trabalho para as aulas de CE-299.



Reuben Solomon Katz Entrou no Instituto Tecnológico da Aeronáutica em 2018. Fez esse artigo como trabalho para as aulas de CE-299.