Coordenando permissões de postagem em *blogs* por meio de *Web fingerprinting*

Jordan Queiroz[1], **Samantha Correa**[1], **Eduardo Feitosa**[1], **Bruno Gadelha**[1]

[1]Instituto de Computação – Universidade Federal do Amazonas (UFAM)
CEP 69.077-000 – Manaus – AM – Brasil

{jsq,scl,efeitosa,bruno}@icomp.ufam.edu.br

Abstract. *Web fingerprinting is a technique that takes advantage of the Web technologies to extract hardware-related and software-related characteristics in order to generate, with high likelyhood, a key capable of unically identify a user. Web fingerprinting can be used in two ways: benign (for authentication and evade frauds purposes) and malicious (user tracking and vulnerability exploitation purposes). In this wid ork proposed a tool which employs Web fingerprinting to manage user's posting permission in a collaborative system environment.*

Resumo. *Web fingerprinting é uma técnica que consiste em obter características relacionadas ao software e hardware do dispositivo. De posse dessas características, uma chave de identificação pode ser gerada com alta probabilidade de ser única, dessa forma, identificando unicamente o usuário. Sua adoção dar-se a partir de duas perspectivas: benigna (autenticação e evitar fraudes) e maligna (rastrear usuários e explorar vulnerabilidades). Neste trabalho é proposto uma ferramenta que emprega Web fingerprinting para gerenciar as permissões de postagens dos usuários em ambientes de sistemas colaborativos.*

Palavras-chave – *Web fingerprinting*, coordenação, gerenciamento, Modelo 3C de Colaboração, *blog*, Sistemas colaborativos.

1. Introdução

A *Web* é o meio mais usado atualmente para realizar diversas atividades, alguns exemplos são: Transações bancárias, comunicação em mídias sociais, compartilhamento de imagens e vídeos, dentre outras [Khademi et al., 2015]. Todos esses avanços também permitem, cada vez mais, a interação e colaboração entre as pessoas por meio dos sistemas colaborativos.

Embora a *Web* e sistemas colaborativos proporcionam grande interação entre as pessoas, deve-se ressaltar que nem todas elas podem estar dispostas a se portar de forma aceitável uma para com a outra ou com o sistema em si. Por exemplo, *trolls*[1], *spams*[2] e *haters*[3] mostram-se como ameaças que diminuem a experiência de uso e a qualidade da plataforma, seja em termo de conteúdo ou convivência online [Cambria et al., 2010].

[1]Usuários anônimos que espalham mensagens ofensivas e abusivas [Cambria et al., 2010].

[2]Mensagens indesejadas e/ou não solicitadas que são enviadas e recebidas por e-mail, *chats*, blogs, etc; cujo objetivo é fazer anúncios, *phishing*, códigos maliciosos, dentre outros [Chakraborty et al., 2016].

[3]Usuários que demonstram críticas injustas e ódio a outros usuários ou coisas.

Existem trabalhos propostos na literatura que visam combater *spams*, *trolls*, bem como outros usuários e mensagens indesejadas, como por exemplo, Cambria et al. [2010] e Eggendorfer and Keller [2006]. No entanto, não fornecem claramente os meios que podem ser usados para prevenir e/ou remediar essa situação e, quando fornecem, o método não é o mais eficaz (bloqueio de endereço IP, por exemplo). Em todos os casos, é necessário ter o conhecimento de quem é o indivíduo responsável pela ação, como evidenciado nos trabalhos Teodoro et al. [2015] e Eggendorfer and Keller [2006].

Tendo em vista os métodos encontrados na literatura e suas limitações, neste trabalho é proposto o uso de *Web fingerprinting*, que é uma técnica usada para identificar, com alta probabilidade e de forma única, um usuário com base nas características de seu dispositivo [Acar et al., 2014]. Esta técnica pode ser usada para diversos fins, seja benigno (autenticação de usuários, melhoria de usabilidade, evitar fraudes, etc) ou malicioso (rastrear atividades de usuários, explorar vulnerabilidades, dentre outros) [Laperdrix et al., 2016].

Como contribuição, este trabalho investiga a aplicação de *Web fingerprinting* em um sistema colaborativo, neste caso, um *blog*,visando identificar, de forma única, os visitantes e coordenar suas postagens, com o objetivo de evitar a propagação de mensagens indesejadas. Este trabalho considera mensagens indesejadas aquelas oriundas de *trolls*, *haters*, *spams* e de outros indivíduos cujo comportamento é prejudicial para a interação e colaboração entre pessoas.

O trabalho está organizado como segue. Na Seção 2 é dado o *background* sobre *Web fingerprinting* e Modelo 3C de Colaboração; na Seção 3 os trabalhos relacionados são apresentados; na Seção 4 é apresentada a metodologia empregada no desenvolvimento deste trabalho; a Seção 5 apresenta o projeto de implementação do artefato desenvolvido; a Seção 6 apresenta os resultados oriundos dos experimentos e, por fim, a Seção 7 conclui o trabalho.

2. Fundamentação teórica

Esta Seção apresenta os conceitos, definições e explanações relacionados a *Web fingerprinting* e Modelo 3C de Colaboração.

2.1. Modelo 3C de Colaboração

Os sistemas colaborativos provêm meios para que as pessoas possam interagir umas com as outras, geralmente com o objetivo de realizar alguma atividade. O modelo 3C de Colaboração considera a colaboração como uma interação de três dimensões [Fuks et al., 2008] a saber:

1. **Comunicação**: Está relacionada com a troca de mensagens entre as pessoas.
2. **Coordenação**: Está relacionada com o gerenciamento de pessoas, suas atividades e recursos.
3. **Cooperação**: Ação de cooperar, de auxiliar e colaborar, prestando ajuda ou auxílio.

Este trabalho se vale da classificação do Modelo 3C de Colaboração, pois de acordo com Fuks et al. [2008] o mesmo pode ser usado para analisar, representar e também servir de base para o desenvolvimento de *groupware*. Além disso, a relação

entre comunicação, coordenação e cooperação do modelo pode ser usada como guia para analisar aplicações de *groupware*, por exemplo, *chats* e *blogs*, ferramentas que requerem a troca de mensagens (comunicação), políticas de acesso (coordenação) e compartilhamento (cooperação). A Figura 1 ilustra o Modelo 3C de Colaboração.

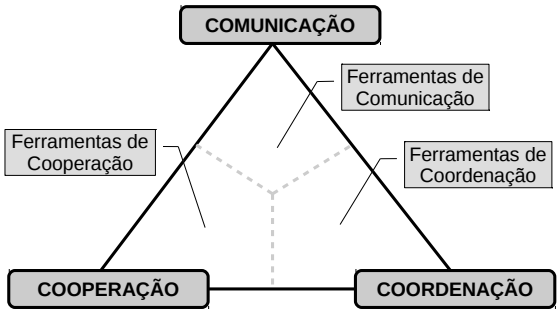

Figura 1. Modelo 3C de colaboração Pimentel et al. [2006]

2.2. *Web fingerprinting*

Web fingerprinting é a técnica que visa identificar, com alta probabilidade, um usuário a partir da geração de uma chave identificadora (*fingerprint*) obtida pela extração de características relacionadas ao *hardware* e *software* de seu dispositivo. A Figura 2 ilustra a dinâmica de um *Web fingerprinting*.

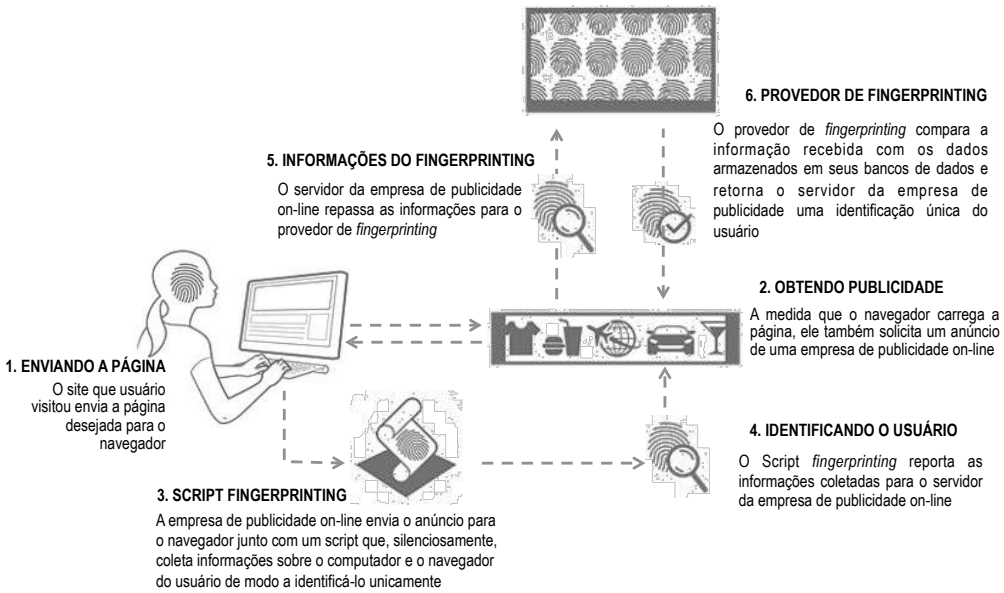

Figura 2. Dinâmica de um *Web fingerprinting* [Saraiva et al., 2014]

Como o foco da pesquisa em desenvolvimento neste trabalho é apresentar a aplicação de *Web fingerprinting* em um ambiente colaborativo para dificultar e até mesmo impedir comportamentos indevidos, ao invés de propor um método, será empregado aquele desenvolvido no trabalho de Queiroz and Feitosa [2016].

O artefato desenvolvido e apresentado neste trabalho consiste em uma ferramenta de suporte à coordenação das atividades de usuários em um sistema de *blog*, visto que o

objetivo é limitar a disseminação de mensagens impróprias em *blogs*, por meio do gerenciamento de atividades relacionadas a comunicação e interação entre usuários.

3. Trabalhos relacionados

Abu-Nimeh and Chen [2010] analisam os comentários postados em *blogs* e os classificam por do classificador SVM (*Support Vector Machine*), de forma a definir se um comentário é ou não um *spam*. Além de usar o classificador, os autores também empregam heurísticas para incrementar a precisão do mesmo na detecção de *spam*. Tal heurística se baseia no endereço de IP ou *e-mail* para mensurar a reputação de quem postou o comentário. Os autores executaram um experimento que consistiu em medir a prevalência de *spam* em comentário de *blogs*. Para isso, foram coletados e analisados 1.048.567 comentários obtidos em *blogs*. Como resultado, os autores concluíram que 75% dos comentários eram *spams*. Além disso, as pessoas que disseminavam *spam* tinham métodos de contornar contramedidas como IP *blacklisting*, por exemplo.

Unger et al. [2013] propõem um *framework* que, através de *Web fingerprinting*, é capaz de monitorar as características de um navegador *Web*. Através desse monitoramento a ferramenta detecta quando há roubo de sessão. Os autores optam por essa estratégia porque atualmente, as autenticações em sites são realizadas apenas uma vez e então um *cookie* de sessão fica armazenado no dispositivo do usuário, sendo que o primeiro pode ser interceptado em uma rede desprotegida. A técnica de *Web fingerprinting* que os autores utilizam se baseia nos padrões do HTML5 e do CSS3, que até o tempo de escrita do trabalho desenvolvido por Unger et al. [2013], possuíam muitas diferenças de implementação em diversos *browsers*, dessa forma, viabilizando os métodos de *Web fingerprinting*. No entanto, a técnica proposta é vulnerável ao ataque *man-in-the-middle*[4] pois, segundo os autores, o algoritmo de criptografia Diffie-Hellman[5] no JavaScript é vulnerável a tal ataque durante a negociação do segredo da chave.

Bursztein et al. [2016] apresentam um método de *device class fingerprinting* capaz de determinar a classe de um dispositivo, sendo esta composta pelo conjunto {*navegador, sistema operacional*}. Diferentemente dos outros trabalhos da literatura, esse método não identifica de forma única um dispositivo. A principal tecnologia utilizada pela proposta é o HTML5 Canvas. A técnica desenvolvida pelos autores cria desafios para os quais o dispositivo precisa dar respostas que são verificadas pelo servidor. Com estas é possível reconhecer a assinatura da classe do dispositivo e, dessa forma, os autores obtém 100% de precisão ao determinar a classe do dispositivo. Ademais, afirmam conseguir detectar tentativas de *login* por força bruta, ataques a lojas de aplicativo (*fake install*, *fake rating*, etc). No entanto, o trabalho não identifica de forma única um dispositivo, mas sim uma classe composta pelo conjunto {*navegador, sistema operacional*}.

4. Metodologia

O objetivo desta pesquisa é aplicar *Web fingerprinting* de forma benigna, isto é, visando beneficiar as pessoas que utilizam a ferramenta (neste caso, um *blog*) que implementa o primeiro. A metologia empregada no desenvolvimento da pesquisa foi o *Design Science Research* que, de acordo com Dresch et al. [2015], consiste na obtenção de artefatos com

[4]Ataque em que os dados trocados entre duas pessoas são interceptados por uma terceira.

[5]Algoritmo específico para a troca de chaves entre dispositivos.

capacidade de resolver problemas do mundo real, mas que também contribuem com a comunidade científica. De forma sucinta, a metodologia supracitada requer as seguintes etapas: Definição do problema a ser resolvido, proposta de solução, desenvolvimento e avaliação do artefato, conclusões e divulgação dos resultados experimentais.

O problema abordado neste trabalho emergiu do fato de que atualmente a interação entre as pessoas está cada vez maior em diversos ambientes, inclusive na *Web*, favorecendo a colaboração entre indivíduos. No entanto, para que o bom convívio possa existir em um ambiente virtual, é necessário evitar situações que possam ameaçá-lo, por exemplo, usuários mal-intencionados cujas atitudes se mostram ofensivas, difamatórias e até mesmo discriminatórias. Dessa forma, ocorreu a seguinte pergunta: ”Como dificultar e até mesmo impedir que as atividades de indivíduos considerados maliciosos possam prejudicar a boa interação e colaboração entre pessoas na *Web*?”

Dessa maneira, observou-se a oportunidade de propor uma solução alternativa que emprega conhecimentos da área de segurança da informação em conjunto com sistemas colaborativos. Para isso, este trabalho conta com o desenvolvimento e avaliação de uma ferramenta que emprega um método de *Web fingerprinting*, que por sua vez é integrada a um *blog*, sendo que este foi escolhido por ser uma das plataformas que permitem a interação e colaboração entre as pessoas. O objetivo do artefato é dificultar e impedir a disseminação de mensagens indesejadas oriundas de usuários maliciosos na *Web*, preservando a boa convivência e colaboração entre indivíduos e a qualidade das informações disponibilizadas em ambientes virtuais *Web* (no caso deste trabalho, um *blog*).

Para validar o artefato, foi realizado um estudo com potenciais usuários de *blog*. A métrica para avaliação e validação é se um dado usuário, uma vez que sua conduta seja classificada manualmente pelo administrador como indevida, poderá interagir novamente com outras pessoas por meio de sua conta autêntica, contas falsas e dispositivos diferentes. É importante notar que a classificação é realizada de forma manual, ou seja, é necessário que o administrador encontre a publicação hostil (comentário exprimindo ódio, por exemplo) e observe o usuário que a realizou e, logo em seguida, prosseguir para a ferramenta e revogar a permissão de publicação do usuário.

5. Projeto e implementação

O artefato desenvolvido neste trabalho está divido em módulos. Cada um deles está listado e descrito como segue. A Figura 3 ilustra a integração da ferramenta com o *blog*.

- **Coletor**: É o método de *Web fingerprinting* responsável por coletar as características relacionadas ao dispositivo do usuário, gerando uma chave única de identificação e transmitindo-a para o **verificador**.
- **Verificador**: Esse módulo recebe do **coletor** a chave de identificação do cliente. De posse da mesma, este módulo verifica se o usuário tem autorização ou não para interagir no *blog*.
- **Banco de dados**: Armazena o identificador único de cada usuário, bem como o seu nome e o *status* da permissão de postagem, sendo que a esta é concedida enquanto o *status* é *1* e revogada quando é *0*.
- **Bloqueador**: Cabe a este módulo capacitar o responsável pelo *blog* ou pela página *Web* de bloquear a permissão de postagem do usuário considerado hostil, preservando a boa interação e colaboração entre usuários.

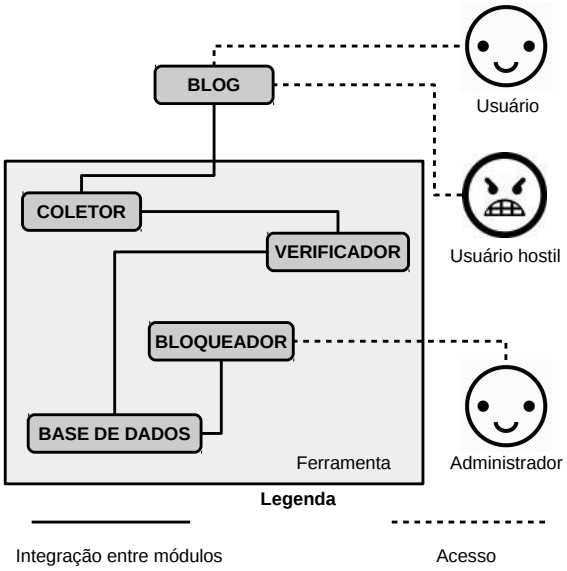

Figura 3. Integração da ferramenta com o *blog*.

Vale ressaltar que o *blog* não está listado, uma vez que este é o objeto de estudo da proposta. O *blog* funciona como um ambiente de interação e colaboração entre as pessoas, proporcionando comunicação entre as mesmas, além de permitir que o administrador coordene as atividades dos usuários segundo as condutas (boas ou ruins) dos mesmos. Todos os módulos foram integrados com o *blog* e este foi instalado em um servidor localizado na universidade, possibilitando que usuários voluntários pudessem participar dos experimentos.

Na Figura 3, os usuários acessam o ***blog***, mais precisamente a página para publicar comentários. Então o **coletor** extrai as características relacionadas ao dispositivo do usuário para gerar o *fingerprint* do mesmo. Após este processo, o coletor envia a chave identificadora para o **verificador**, que por sua vez checa pela existência da mesma. Se o *fingerprint* não existir, então este é adicionado ao **banco de dados**, em uma tabela de novos usuários e é concedida a permissão de postagem para o visitante. Caso contrário, uma das duas situações pode ocorrer: (I) A chave é existente e o visitante tem permissão de postagem, dessa forma, a ferramenta permitirá publicações ou (II) A chave consta no registro do banco, mas o usuário não tem permissão para publicar e, consequentemente, o artefato não dará permissão para enviar mensagens, impedindo a interação com outros usuários. Quando uma postagem que não respeita as políticas de conduta do ambiente for detectado pelo administrador do mesmo, é possível revogar a permissão de publicação de quem enviou a mensagem, por meio do **bloqueador**. Este módulo lista todos os *fingerprints* e seus respectivos nomes de usuários, exibindo a opção de bloquear qualquer pessoa cujas publicações sejam consideradas indevidas. A interface gráfica do módulo bloqueador está ilustrada na Figura 4.

6. Avaliações e resultados experimentais

Após a integração da ferramenta ao *blog*, foi realizado um experimento com usuários voluntários. A experimentação ocorreu no laboratório de informática da universidade com os alunos de uma disciplina, onde cada um deles acessou o *blog* e criou um cadastro

Figura 4. Interface gráfica do módulo bloqueador.

para poder publicar mensagens. Ao todo, o experimento contou com a participação de 17 alunos, sendo que alguns optaram por usar o próprio computador e/ou celular, enquanto os demais optaram por usar os computadores disponíveis no laboratório, sendo estes máquinas clonadas. Optou-se por restringir a permissão de postagens a usuários cadastrados pois foi assumido que, em situações habituais, usuários anônimos podem tirar proveito para publicar mensagens indevidas, no entanto, o foco deste trabalho é impedir que usuários já cadastrados (seja com uma conta autêntica ou falsa) publiquem mensagens indesejadas.

O processo experimental consistiu em diversas etapas compostas por tentativas de publicar mensagens. As fases são descritas como segue e a Figura 5 ilustra, em alto nível, o processo experimental. (I) Criar uma conta no *blog*, realizar a autenticação, encontrar uma publicação e publicar algum comentário em relação a mesma, interagindo com outros indivíduos; (II) Após as primeiras publicações, selecionar dois usuários distintos a partir do módulo bloqueador e revogar a permissão de publicação dos mesmos; (III) Solicitar que algum voluntário capaz de interagir no sistema realize o *login* em um dispositivo cujo alguém tenha sido bloqueado e então tente publicar alguma mensagem; (IV) Bloquear todos os usuários e observar se algum deles conseguem realizar novas publicações; (V) Solicitar a todos os usuários cuja permissão de publicação foi negada que os mesmos tentem postar mensagens a partir de dispositivos em quais não foram bloqueados e (VI) Após uma nova rodada de bloqueios total, solicitar para que todos os voluntários troquem de navegador *Web* e tentem publicar mensagens.

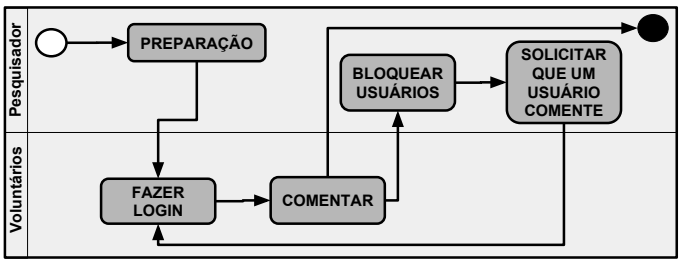

Figura 5. Diagrama em alto nível das etapas do experimento realizado.

6.1. Resultados e discussão

Todos os usuários conseguiram publicar mensagens na primeira etapa, pois tinham permissão para realizar tal atividade. Na segunda etapa, os usuários selecionados para serem bloqueados ficaram impedidos de realizar postagens, enquanto os demais continuaram a interagir uns com os outros no *blog*. Na terceira etapa, não foi permitido que um usuário interagisse com o *blog* a partir de um dispositivo no qual seu dono estava impedido de interagir com o sistema e seus usuários. Na quarta etapa, onde todos os voluntários foram bloqueados, não houve interação alguma. No quinto estágio, com todos os voluntários usando dispositivos diferentes dos anteriores, observou-se interações entre os indivíduos (mesmo que todos tivessem sido bloqueados anteriormente). Na sexta e última etapa, todos foram bloqueados novamente, mas mesmo assim houve interação no *blog* quando um navegador *Web* alternativo era utilizado para acessá-lo.

Os resultados observados na terceira e quinta etapa podem ser preocupantes, devido ao usuário com boas condutas ficar incapacitado de interagir a partir de um dispositivo bloqueado, enquanto um indivíduo com condutas impróprias ser capaz de interagir a partir de um dispositivo não bloqueado. No entanto, vale ressaltar que geralmente as pessoas utilizam seus próprios dispositivos. Dessa maneira, as ocorrências de um usuário benigno logar a partir um dispositivo bloqueado (este pertencente a um usuário hostil) são baixas.

Embora na quinta etapa seja possível que um usuário hostil esteja capacitado de interagir com as outras pessoas no *blog* por meio de um dispositivo diferente daquele bloqueado, é importante notar que o indivíduo pode ter a permissão de publicação revogada no novo dispositivo, dessa forma ficando impedido de interagir por meio dos dois dispositivos. Em um determinado momento, o usuário hostil ficará com todos os dispositivos bloqueados e usar emulação pode ser um trabalho desgastante, dificultando o comportamento indevido.

Com relação a revogação de interação, uma das maneiras de bloquear um indivíduo malicioso de forma eficaz é através de suas credenciais de *login*. No entanto, é possível criar contas falsas e usá-las diversas vezes a partir do mesmo dispositivo, pois este não estará bloqueado. Uma máquina bloqueada, por outro lado, impedirá que um indivíduo interaja por meio de contas falsas. Obviamente técnicas de *fingerprinting* não são as balas de prata para revogar permissões, mas podem ser úteis para complementar os meios convencionais. Um dos exemplos de que *fingerprinting* não resolve todos os problemas é quando um usuário troca de navegador *Web* (como é possível observar na sexta etapa do experimento), gerando uma nova chave de identificação única, mas isso acontece porque as técnicas de *fingerprinting* mais comuns extraem características relativas ao navegador *Web*. Dessa maneira, fica evidente mais vez uma que *fingerprinting* pode ser usado para complementar mecanismos existentes que atuam na prevenção contra condutas impróprias em ambientes colaborativos *online*.

6.1.1. Limitações

Pelo fato do artefato proposto neste trabalho ser um protótipo funcional, o mesmo possui limitações que, dependendo do caso, podem impedir o seu devido funcionamento.

Pelo fato do método de *Web fingerprinting* empregado pelo artefato desenvolvido neste trabalho ser volátil a mudanças no navegador, isso pode ser encarado como uma limitação que pode frustrar em médio prazo a operação de bloqueio. No entanto, vale ressaltar que esta limitação é um dos problemas em aberto na literatura de *Web fingerprinting*, embora existam propostas para tratar essa exceção, estas não estão no contexto deste trabalho.

Por fim, outra limitação é que esse método não funcionará em máquinas clonadas[6] (muito presentes em universidades), pois o *Web fingerprinting* extrai características de *hardware* e *software*. No entanto, por todas serem idênticas, assim que uma for bloqueada ou desbloqueada, todas também serão. Vale ressaltar também que este é um dos problemas em aberto na literatura de *Web fingerprinting*.

7. Conclusões

Este trabalho desenvolveu, apresentou e avaliou um artefato que emprega *Web fingerprinting*, visando beneficiar os indivíduos que utilizam ferramentas onde há interação e colaboração entre as pessoas, proporcionando também ao responsável pela ferramenta a coordenação das atividades dos usuários, para dificultar e até mesmo impedir as atividades de interação daquele considerado como nocivo a boa convivência no ambiente online (no caso deste trabalho, um *blog*).

Observou-se, a partir dos experimentos, que a ferramenta desenvolvida neste trabalho é capaz de bloquear as interações de usuários que violam a conduta de uso, uma vez que a pessoa responsável pelo *blog* decidir bloquear as postagens de um usuário. Pode-se observar também que a ferramenta associa o bloqueio ao dispositivo de um indivíduo (ao contrário de considerar apenas o nome de usuário), dificultando uso de contas falsas como um vetor de ataque. Embora as trocas de dispositivos permitam a um *hater* ou *troll* realizar publicações indevidas, agir dessa forma requer maior quantidade de esforço, o que pode desmotivar a ação.

Por fornecer ao responsável do *blog* o controle sobre as permissões de interações dos usuários, a ferramenta pode ser analisada de acordo com Modelo 3C de Colaboração, mas com o foco na coordenação, que consiste no gerenciamento de pessoas, suas atividades e recursos.

7.1. Trabalhos futuros

Como trabalhos futuros, podem-se destacar:

- Associar o bloqueio não só ao dispositivo de um usuário hostil, mas também à conta do mesmo.
- Identificar um usuário mesmo que seu *fingerprint* tenha sofrido modificação.

8. Agradecimentos

Os autores deste trabalho agradecem à todas as pessoas envolvidas na revisão do texto, sugestões e na participação dos experimentos e também agradecem à instituição de fomento CAPES por viabilizar esta pesquisa.

[6]Máquinas que possuem a mesma configuração de *hardware* e *software*.

Referências

Abu-Nimeh, S. and Chen, T. M. (2010). Proliferation and Detection of Blog Spam. *IEEE Security Privacy*, 8(5):42–47.

Acar, G., Eubank, C., Englehardt, S., Juarez, M., Narayanan, A., and Diaz, C. (2014). The Web Never Forgets: Persistent Tracking Mechanisms in the Wild. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 674–689, Scottsdale, Arizona, USA. ACM New York, NY, USA ©2016.

Bursztein, E., Malyshev, A., Pietraszek, T., and Thomas, K. (2016). Picasso: Lightweight Device Class Fingerprinting for Web Clients. In *Proceedings of the 6th Workshop on Security and Privacy in Smartphones and Mobile Devices*, pages 93–102, Vienna, Austria. ACM New York, NY, USA ©2016.

Cambria, E., Chandra, P., Sharma, A., and Hussain, A. (2010). Do Not Feel The Trolls. *ISWC*.

Chakraborty, M., Pal, S., Pramanik, R., and Ravindranth Chowdary, C. (2016). Recent developments in social spam detection and combating techniques: A survey. *Information Processing & Management*, 52(6):1053–1073.

Dresch, A., Lacerda, D. P., and Júnior, J. A. V. A. (2015). *Design science research: método de pesquisa para avanço da ciência e tecnologia*. Bookman Editora.

Eggendorfer, T. and Keller, J. (2006). Dynamically blocking access to web pages for spammers' harvesters. In *Communication, Network, and Information Security*, pages 205–210.

Fuks, H., Raposo, A., Gerosa, M., Pimental, M., and Lucena, C. (2008). The 3C Collaboration Model. In *Encyclopedia of E-collaboration*, pages 637–644. IGI Global.

Khademi, A. F., Zulkernine, M., and Weldemariam, K. (2015). An Empirical Evaluation of Web-Based Fingerprinting. *IEEE Software*, 32(4):46–52.

Laperdrix, P., Rudametkin, W., and Baudry, B. (2016). Beauty and the Beast: Diverting modern web browsers to build unique browser fingerprints. In *IEEE Symposium on Security and Privacy (SP)*, pages 878–894. IEEE.

Pimentel, M., Gerosa, M. A., Filippo, D., Raposo, A., Fuks, H., José, C., and Lucena, P. D. (2006). Modelo 3C de Colaboração para o desenvolvimento de Sistemas Colaborativos. *Anais do III Simpósio Brasileiro de Sistemas Colaborativos*, pages 58–67.

Queiroz, J. S. and Feitosa, E. L. (2016). LETTY: Uma implementação de Website Fingerprinting. In *Anais do XVI Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais*, volume 1, pages 555–564, Niterói - RJ, Brazil.

Saraiva, A., Feitosa, E., Elleres, P., and Carneiro, G. (2014). Device Fingerprinting: Conceitos e Técnicas, Exemplos e Contramedidas. In Santos, A., Graaf, J., Nogueira, J., and Oliveira, L., editors, *Livro de Minicursos do XIV Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais - SBSeg2014*, pages 49–98, Belo Horizonte - MG. SBC.

Teodoro, J., Gerosa, M. A., Wiese, I. S., and Steinmacher, I. (2015). Quem é quem na lista de discussão? Identificando diferentes e-mails de um mesmo participante. In *Anais do Simpósio Brasileiro de Sistemas Colaborativos*, pages 78–85.

Unger, T., Mulazzani, M., Fruhwirt, D., Huber, M., Schrittwieser, S., and Weippl, E. (2013). SHPF: Enhancing HTTP(S) session security with browser fingerprinting. In *2013 International Conference on Availability, Reliability and Security*, pages 255–261, Regensburg, Germany. IEEE.