MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
DEPARTAMENTO DE CIÊNCIA E TECNOLOGIA
INSTITUTO MILITAR DE ENGENHARIA
CURSO DE MESTRADO EM SISTEMAS E COMPUTAÇÃO

VANESSA QUADROS GONDIM LEITE

GERAÇÃO DE *DATASET* A PARTIR DA CRIAÇÃO DE UMA *SOCIAL BOTNET*

Rio de Janeiro
2018

INSTITUTO MILITAR DE ENGENHARIA

VANESSA QUADROS GONDIM LEITE

GERAÇÃO DE *DATASET* A PARTIR DA CRIAÇÃO DE UMA *SOCIAL BOTNET*

Dissertação de Mestrado apresentada ao Curso de Mestrado em Sistemas e Computação do Instituto Militar de Engenharia, como requisito parcial para a obtenção do título de Mestre em Ciências em Sistemas e Computação.

Orientador: Prof. Ronaldo Moreira Salles - Ph.D.

Rio de Janeiro
2018

INSTITUTO MILITAR DE ENGENHARIA
Praça General Tibúrcio, 80 - Praia Vermelha
Rio de Janeiro - RJ CEP 22290-270

004.69 Leite, Vanessa Quadros Gondim
S586e Geração de *dataset* a partir da criação de uma *social botnet* / Vanessa Quadros Gondim Leite, orientada por Ronaldo Moreira Salles - Rio de Janeiro: Instituto Militar de Engenharia, 2018.

81p.: il.

Dissertação (mestrado) - Instituto Militar de Engenharia, Rio de Janeiro, 2018.

1. Curso de Sistemas e Computação - teses e dissertações. 1. social botnet. 2. socialbot. 3. dataset. 4. security information. 5. segurança da informação. 6. mídias sociais. 7. OSN. 8. Facebook. I. Salles, Ronaldo Moreira. II. Título. III. Instituto Militar de Engenharia.

INSTITUTO MILITAR DE ENGENHARIA

VANESSA QUADROS GONDIM LEITE

GERAÇÃO DE *DATASET* A PARTIR DA CRIAÇÃO DE UMA *SOCIAL BOTNET*

Dissertação de Mestrado apresentada ao Curso de Mestrado em Sistemas e Computação do Instituto Militar de Engenharia, como requisito parcial para a obtenção do título de Mestre em Ciências em Sistemas e Computação.

Orientador: Prof. Ronaldo Moreira Salles - Ph.D.

Aprovada em 3 de Maio de 2018 pela seguinte Banca Examinadora:

Prof. Ronaldo Moreira Salles - Ph.D. do IME - Presidente

Prof[a]. Raquel Coelho Gomes Pinto - D.Sc. do IME

Prof. Sidney Cunha de Lucena - D.Sc. da Unirio

Rio de Janeiro
2018

À minha família e ao Instituto Militar de Engenharia, alicerces da minha formação e aperfeiçoamento.

AGRADECIMENTOS

Primeiramente, agradeço a minha família pelo suporte irrestrito nas minhas escolhas ou ações.

Agradeço ao Instituto Militar de Engenharia e a todos os professores que me acompanharam durante o mestrado.

Ao Ronaldo Moreira Salles, meu orientador, pelos direcionamentos.

A todos que contribuiram para os experimentos.

Aos autores referenciados nesta obra.

Um agradecimento especial: ao meu pai Ricardo, à minha irmã Patrícia, à minha vó Elisa pela força e incentivo; aos meus amigos Érick, Felipe, Major Arias e Mário, que me acompanharam em diversos momentos durante o mestrado; ao Gabriel L., ao Filipe Braida e ao Xiao, pela disponibilidade e pelo apoio dado.

Aos demais amigos, pela paciência e compreensão diante a minha ausência.

A todos os outros, que direta ou indiretamente, contribuíram para a conclusão desta dissertação.

E finalmente agradeço a Deus, por todos os aprendizados e oportunidades que Ele me proporcionou até hoje.

"Quem só acredita não é quem sempre alcança. Quem alcança é quem corre atrás dos seus objetivos. "

JUÇARA QUADROS

SUMÁRIO

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE SIGLAS

EUA	Estados Unidos da América
ONS	*Online Network Social*
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
API	*Application Programming Interface*
FIS	*Facebook Immune System*
DACS	*Design and Analysis of Communication Systems*
CAIDA	*Cooperative Association for Internet Data Analysis*
PREDICT	*Protected Repository for the Defense of Infrastructure Against Cyber Threats*
WITS	*Waikato Internet Traffic Storage*
MOM	*Message Oriented Middleware*
SDK	*Software Development Kit*
URI	*Uniform Resource Identifier*
HTTP	*Hypertext Transfer Protoco*
URL	*Uniform Resource Locator*
SMS	*Short Message Service*
JEE	*Java Platform, Enterprise Edition*
JMS	*Java Message Service*
IA	Inteligência Artificial

LISTA DE ABREVIATURAS

ABREVIATURAS

SbN - Social Botnet

RESUMO

O Facebook é uma das maiores mídias sociais e está cada vez mais ameaçado por *socialbots*, que são perfis controlados por software capazes de realizar ações malignas ou benignas. Isso envidência a necessidade de criação de contramedidas para os cenários em que o uso dos *socialbots* e do Facebook seja direcionado para fins maliciosos. Entretanto, a detecção de comportamentos anômalos para o desenvolvimento de contramedidas é um desafio, principalmente por causa da ausência de dados descritos, completos e disponívei. Com base nisto, o objetivo desta pesquisa é criar uma arquitetura conceitual que contemple desde a etapa de criação de uma *social botnet* até a geração dos dados. Além disso, através da implementação desta arquitetura, atinge-se o outra finalidade deste estudo, que é gerar um dataset com ações dos socialbots e interações entre eles e humanos no Facebook.

ABSTRACT

Facebook is one of the largest social media and is increasingly threatened by socialbots, which are software-controlled profiles capable of performing malignant or benign actions. This envisages the need to create countermeasures for scenarios where use of socialbots and Facebook is targeted for malicious purposes. However, the detection of anomalous behaviors for the development of countermeasures is a challenge, mainly because of the lack of data described, complete and available. Based on this, the objective of this research is to create a conceptual architecture that contemplates from the stage of creation of a social botnet until the generation of the data. In addition, through the implementation of this architecture, we achieve the other purpose of this study, which is to generate a dataset with socialbots actions and interactions between them and humans on Facebook.

1 INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO E MOTIVAÇÃO

Com a popularização da Internet, alguns serviços de comunicação e de entretenimento começaram a apresentar maior relevância e uso, como as *Online Social Networks* (OSNs) (HE et al., 2015). Um exemplo deste crescimento na utilização é observado em relação ao Facebook[1], que atingiu pela primeira vez a meta de 2 bilhões de usuários em 2017 (REPORTS, 2018). O aumento no uso das mídias sociais fica ainda mais evidente no Brasil, líder em relação ao tempo gasto em redes sociais, com média de 60% a mais que os demais países (ALEX, 2015).

O crescimento na quantidade de usuários das OSNs progride devido ao surgimento de novos meios de acesso, como os aplicativos de celulares, visto que quase 9 milhões de internautas no Brasil acessam a Internet exclusivamente de seus *tablets* e celulares (ALEX, 2015). Isso influencia inclusive um *ranking* feito em 2017, no qual Youtube[2] e Facebook estão presentes na lista dos 10 primeiros sites mundialmente visitados (AMAZON, 2017).

Devido à grande popularização, as mídias sociais começaram a desempenhar novas funções, como a execução de *marketing* social (KIMURAI et al., 2008). Um dos meios de execução deste tipo de divulgação é a partir de interação humana na própria página da empresa na mídia social ou de *socialbots*, que são softwares de automação que controlam perfis na mídia social e apresentam capacidade de executar atividades *online* (BOSHMAH et al., 2011; DAPP, 2017a).

Assim, através da criação de *socialbots* ou robôs sociais infere-se que o conteúdo das redes sociais começou a ser gerado também por usuários não legítimos (ou falsos). O Facebook, por exemplo, estima que existem mais de 83 milhões de contas falsas (CELLAN-JONES, 2016). Com isso, as OSNs passaram a ser exploradas pelos *socialbots* para diferentes fins em contextos também distintos, como:

- aumento de popularidade, associado ao marketing próprio ou de um produto; (PAULO, 2016);

[1] https://www.facebook.com/
[2] https://www.youtube.com/

- divulgação de campanha política, como na eleição do Obama nos Estados Unidos da América (EUA) (POSTER, 2016) e eleição do Aécio e Dilma no Brasil (DAPP, 2017b; FABIO, 2014);

A fim de influir pessoas em grande escala, através do uso de *socialbots*, que são uma massa de perfis capazes de adquirir informações que nem sempre são de acesso a todos, é possível otimizar o uso de recursos para explorar a OSN (KRISHNA, 2015). Isso gera um impacto na área de segurança da informação em função dos seus três princípios básicos (confidencialidade, disponibilidade e integridade) (WHITMAN; MATTORD, 2011) que podem ser afetados pelas ações dos *socialbots*. Essas são atividades que um robô social pode executar em um OSN e que são similares às ações realizadas por um usuário legítimo.

Existem casos em que eles são usados para fins benéficos, comuns à sociedade. Um exemplo pontual do uso deles neste contexto de realizar o bem social foi a mobilização para o auxílio às vítimas das enchentes na Zona Oeste do Rio de Janeiro (SILVIA, 2016). Entretanto, as mídias sociais podem ser exploradas para fins também maliciosos. Em Globo (2014) é apresentado um caso em que um indivíduo levou apenas 10 dias para coletar no Facebook informações necessárias para sequestrar uma criança.

Em função das diversas situações onde pode ocorrer o uso indevido das OSNs, observa-se a importância de explorar e aplicar a segurança da informação neste contexto, sobretudo por ser um campo de pesquisa que enfrenta desafios em busca de mitigar comportamentos anômalos, gerados por usuários comuns ou maliciosos (GARCIA-TEODORO et al., 2009).

Exemplificam-se como anomalia em uma rede o comportamento gerado e observado a partir de uma *botnet*. Essa é um conjunto de computadores infectados e controlados por *hackers* que funcionam como uma ferramenta utilizada para diversas ações maliciosas, como disseminar *malwares* (vírus, *worms*, entre outros) e efetuar ataques distribuídos de negação de serviço (FENG et al., 2011).

1.2 *SOCIALBOT*

Socialbot é um software de automação preparado para interagir com seres humanos nas mídias sociais, inclusive é capaz de imitar e de influênciar o comportamentos dos usuários legítimos. Pode ser categorizado a partir da intenção para qual foi criado: benigna ou maligna. No primeiro grupo pertencem, por exemplo, os respondedores automáticos, bastante utilizados por marcas e empresas de atendimento ao cliente. (FERRARA et al., 2014) Já no segundo, consideram-se os robôs sociais com objetivo de enganar, explorar e manipular o discurso das redes sociais através de *spam*, códigos maliciosos, difamação,

entre outros exemplos. Isso pode resultar em vários níveis de danos para a sociedade (FERRARA et al., 2014; DAPP, 2017a).

O impacto dos *socialbots* não está restrito somente às OSNs, já que atinge também a sociedade. Em Bollen et al. (2011) DAPP (2017a) é indicada a capacidade de um robô social influenciar na estabilidade do mercado. Bollen et al. (2011) mostram que existem evidências de que os operadores de mercado financeiro estão atentos e reagem às informações das mídias sociais.

Ainda neste mesmo contexto econômico, Ferrara et al. (2014) descreveram uma campanha de sucesso recente criada com *socialbot*, no qual resultou em um aumento de 200 vezes no valor de mercado de uma empresa de tecnologia chamada de Cynk, tornando o valor da empresa equivalente a US$ 5 bilhões.

Outra função de um *socialbot* é ser utilizado com o objetivo de alcançar uma posição influente a atingir mais pessoas com o seu conteúdo publicado. Esta influência compromete as estruturas que expressam a relação social entre perfis de uma rede social com uma grande quantidade de usuários (BOSHMAF et al., 2013).

De acordo com Ferrara et al. (2014), o novo desafio trazido por estes robôs sociais é o fato de sugerirem a falsa impressão de que alguma informação, independentemente de sua veracidade, é altamente popular e endossada por muitos, exercendo uma influência contra a qual ainda não há um mecanismo de defesa. Inclusive, em DAPP (2017a) é discutido o fato dos *socialbots* representarem risco à democracia, pelos seguintes aspectos:

- alteração do rumo das políticas públicas;
- criação da falsa sensação de amplo apoio a certa ideia proposta ou figura pública;
- disseminação de rumores, notícias falsas e teorias conspiratórias, que geram desinformação e poluição de conteúdo;

Em função dos exemplos supracitados, observa-se que os *socialbots* podem gerar danos severos, o que sinaliza a importância da identificação dos mesmos, com o propósito de diferenciar quais situações são reais e quais são manipuladas (DAPP, 2017a), a fim de evitar principalmente danos aos usuários legítimos ou a uma entidade.

1.3 *SOCIAL BOTNET*

Uma *social botnet* (SbN) é gerenciada através do seu controlador, também denominado formalmente como um *botmaster*. Ele pode ser representado como um indivíduo ou sistema mal-intencionado com acesso a uma ou mais contas da OSN, sendo esse o modo pelo

qual interage com a mídia social. As atividades que um controlador pode executar através de uma *social botnet* são delimitadas pelo conjunto de ações que o OSN disponibiliza para qualquer usuário verdadeiro. (COMPAGNO et al., 2015)

Normalmente, o termo *social botnet* está associado a um conjunto de robôs sociais para fins maliciosos, devido aos casos famosos presentes na literatura. Tem-se como exemplo, Koobface (THOMAS; NICOL, 2010; TANNER et al., 2010), que está ativo ainda hoje e é considerado um dos *socialbots* mais bem sucedidos, em função da sua capacidade de infectar diferentes sistemas operacionais (Linux, Windows, MacOS) e de capturar as credenciais dos usuários do Facebook e de outras mídias sociais. É válido ressaltar que não necessariamente um *socialbot* infecta um dispositivo de usuário, já que podem ser feitos com base na criação de perfis falsos, cujo comportamento é próprio ou clonado de algum perfil verdadeiro (TIWARI, 2017).

O método utilizado para obter tais credencias explora os *cookies* dos *browsers*, já que é um modo desta *social botnet* maliciosa atingir outros perfis a partir de mensagens com *malware* enviadas para os amigos do perfil comprometido (THOMAS; NICOL, 2010) (JARRAD, 2008). Entretanto, a partir do conceito de que uma SbN (*social botnet*) é um conjunto de *socialbot*, considera-se a possibilidade da mesma não estar exclusivamente associada às ações maliciosas, como nos casos em que é usada para *social marketing* (KIMURAI et al., 2008). Assim, a *social botnet* contruída nesta pesquisa não contemplará atividades maliciosas por questões éticas.

1.4 CARACTERIZAÇÃO DO PROBLEMA

As OSNs e os *socialbots* presentes nas mesmas podem ser utilizados para diversas finalidades, sejam positivas ou negativas. Assim, nota-se a importância da criação de contramedidas para os cenários em que o uso de ambos seja direcionado para fins maliciosos, passíveis de causar danos a algum indivíduo ou alguma entidade.

Entretanto, a detecção de comportamentos anômalos é um desafio. Isso porque, para a análise ou detecção de anomalias, muitos artigos envolvem soluções com metodologias estatísticas (BOSHMAF et al., 2013) ou de aprendizagem (DAVIS et al., 2016; TIWARI, 2017), o que apriori, exigem dados descritos para a análise e composição do treinamento (ABT; BAIER, 2014). Uma opção no contexto de análise ou detecção de anomalias nas OSNs, seria que as mídias sociais fornecessem dados que pudessem ser utilizados para fins de pesquisa. Entretanto, muitas companhias, como o Facebook, são relutantes quando o assunto em questão é o compartilhamento de seus dados para estudos (CATANESE et al.,

2011).

Existe também a opção de uso de informação fornecidas por repositórios públicos. No entanto, como a maioria dos repositórios públicos de dados oferecem amostras com informações anonimizadas e sem descrição, os pesquisadores geralmente coletam individualmente os seus *datasets* em ambientes onde possuem acesso e algum controle, pois assim é um meio de garantir o conhecimento necessário sobre as informações relacionadas à sua pesquisa (ABT; BAIER, 2014). Há também cenários em que o estudo acaba tendo um direcionamento em função da limitação dos atributos presentes na base de dados fornecida, pelo fato deste *dataset* não contemplar todas as propriedades escolhidas inicialmente para serem utilizadas na pesquisa, conforme visto em Kumar e Reddy (2012).

Contudo, a criação de um *dataset* próprio especificamente para o desenvolvimento de uma solução para detecção de comportamentos anômalos e a não publicação das informações utilizadas para execução do estudo proposto, comprometem a veracidade dos dados contidos no mesmo e afeta comparabilidade entre as pesquisas de uma mesma área (ABT; BAIER, 2014).

Ainda em relação à detecção, no contexto de OSN, outro problema observado é o constante aprimoramento dos *socialbots*, já que buscam novos mecanismos de evasão cada vez mais complicados e avançados para confundir as ferramentas de detecção (JI et al., 2016; BOSHMAF et al., 2013). Desta forma, acredita-se que a criação e a implementação de uma arquitetura conceitual de uma *social botnet* são formas de melhor entender como uma SbN se infiltra e de avaliar o impacto gerado. A efetividade da arquitetura construída junto à metodologia para criação dos *socialbots* contribuem na identificação dos pontos onde os mecanismos de detecção, sejam de pesquisas ou da própria OSN, precisam ser aprimorados.

1.5 OBJETIVO DO TRABALHO

A partir dos problemas observados, foi definido como objetivo principal deste trabalho: criação de *dataset* classificado para pesquisas de *social botnet* em mídias sociais;

Através deste *dataset* classificado, as pesquisas que atuam no mesmo contexto, não só poderão utilizá-lo para treinar seu método de detecção de *socialbots* e *social botnets*, como também terão uma forma de comparar os resultados e de garantir a imparcialidade do estudo. Em consequência destes objetivos principais, outros secundários surgiram: definição de uma arquitetura conceitual capaz de servir como base para a geração de *dataset* através de uma *social botnet*;proposta de uma metodologia para a criação de

social botnet;

A criação do *dataset* classificado necessita do mapeamento e armazenamento as atividades dos *socialbots*. Por conta disso, identificou-se a necessidade de criar e implementar a arquitetura conceitual apresentada como objetivo secundário.

Em função da efetividade da *social botnet* implicar na geração do *dataset*, os mecanismos de evasão para evitar sua detecção devem ser considerados. Portanto, para aprimorá-la, também é proposta uma metologia para criação de *socialbots*. Com o propósito de elaborar um *dataset* com informações mais relevantes, foi desenvolvido um sistema de automatização, capaz de agrupar um conjunto de ações como pertencentes a um determinado cenário.

1.6 CONTRIBUIÇÕES

De acordo com o que foi demonstrado como problemática e objetivo, compreendem-se como contribuição deste trabalho os seguintes itens:

a) Disponibilizar um conjunto de dados para:

   - treinamento de algoritmos de detecção de *social botnet*;
   - definição de um denominador comum para comparações entre os algoritmos;

b) Propor uma arquitetura conceitual para geração de *dataset* no contexto de OSN, capaz de ser mantida ao longo dos anos por outros pesquisadores;

c) Viabilizar uma metodologia para criação de *socialbot* junto a análise da efetividade da mesma. Esta metodologia também pode ser utilizada como meio de avaliação de ferramentas de segurança presentes nas mídias sociais, como por exemplo o FIS (*Facebook Immune System*). Além disso, a mesma pode ser aplicada para outras redes sociais e é independente da arquitetura proposta;

d) Fornecer uma metodologia para a descrição de um *dataset* relacionado à SbN em uma *Online Social Network*;

e) Indicar os comportamentos relevantes dos *socialbots* a serem considerados nas pesquisas com o contexto de detecção de *socialbots*;

1.7 ORGANIZAÇÃO DO TEXTO

Este trabalho encontra-se organizado da seguinte forma:

O Capítulo 2 apresenta os trabalhos relacionados ao problema estudado e suas principais características. Como o trabalho abrange diversas áreas, os trabalhos relacionados serão divididos de acordo com os principais temas: Detecção de *Social Botnet*, Criação de *Social Botnet* e *Socialbot* e Disponibilização de *datasets* rotulados na área de Segurança da Informação.

O Capítulo 3 descreve a metodologia proposta para a geração dos *socialbots* e a categorização do *dataset* gerado. Desta forma, serão apresentados os motivos da distribuição em cenários com diferentes níveis.

O Capítulo 4 detalha a arquitetura conceitual e sua implementação, bem como as ferramentas utilizadas para viabilizar o desenvolvimento da solução proposta.

O Capítulo 5 demonstra a etapa de execução dos experimentos e os resultados obtidos. Neste caso, obtêm-se informações sobre as ações dos *socialbots* e como o Facebook reage através de uma análise. E também, compara os atributos existentes no *dataset* gerado com os analisados durante a evolução desta pesquisa. Além disso, indica os comportamentos observados no Facebook e nos provedores de email durante os experimentos.

Finalmente, no Capítulo 6, são realizadas as considerações finais sobre a dissertação juntamente com os trabalhos futuros. Também são apresentadas as contribuições gerais, viabilizadas no decorrer do trabalho. Além disso, são apresentadas as dificuldades encontradas para a execução do mesmo, de modo que possa servir de direcionamento para os pesquisadores com interesse na mesma área deste estudo.

2 TRABALHOS RELACIONADOS

Durante a etapa de definição do tema a ser abordado na dissertação, foi verificada a relevância dos estudos que retratam os *socialbots* ou *social botnet* no contexto das mídias sociais, sobretudo em relação ao Facebook, que é a mídia social estudada neste trabalho.

Em uma rápida busca no IEEEXplore[3], por exemplo, para verificar a quantidade de artigos encontrados através da pesquisa com os textos "*socialbot*", "*socialbot* facebook"e "*socialbot* twitter"no período entre 2010 e 2017, foi possível observar que em relação ao primeiro item, foram encontrados 136 artigos, enquanto no segundo e no terceiro foram observados 20 e 55 respectivamente.

Nas buscas onde o período entre 2010 e 2017 foram mantidos e alterou-se apenas o campo de pesquisa para "socialbotnet"e, posteriormente, "socialbotnet facebook", foram listados 59 artigos na primeira opção, enquanto na segunda somente 10. Destes 10 artigos ((THOMAS; NICOL, 2010), (COMPAGNO et al., 2015), (GHANADI; ABADI, 2014), (ZHANG et al., 2017), (AL-DAYIL; DAHSHAN, 2016), (CHEN et al., 2013), (TYAGI; AGHILA, 2012), (AGRAWAL; VELUSAMY, 2016), (WATTERS et al., 2013) e (TANNER et al., 2010)), somente os três primeiros discorrem realmente no escopo do Facebook.

A fim de compreender melhor a área de estudo, houve uma tentativa de encontrar os motivos plausíveis capazes de justificar a quantidade restrita de artigos que atuam no contexto do Facebook. Assim, em Douglas (2017) observou-se uma discussão sobre a abertura das APIs das OSNs, no qual é apresentado que a API do Twitter não tem restrições tão rigosas quanto as do Facebook.

Mesmo com as dificuldades de se trabalhar com o Facebook, ele foi escolhido para ser explorado neste estudo por ter 2 bilhões de usuários (REPORTS, 2018) e por ser a mídia social mais utiliza no mundo (STATISTA, 2018), o que sugere a sua importância e o alto impacto que as ações maliciosas podem gerar.

Como este estudo apresenta atuação sobre Facebook, os trabalhos relacionados foram categorizados a partir dos principais tópicos relacionados com esta pesquisa, em função de não ter sido observado na literatura pesquisas que contemplem simulatenamente todas as frentes presentes nessa.

[3]http://ieeexplore.ieee.org/Xplore/home.jsp

2.1 DETECÇÃO DE *SOCIAL BOTNETS* OU *SOCIALBOTS*

Boshmaf et al. (2013) analisam as *social botnets* e discorrem sobre a detecção das mesmas pelo Facebook. Os autores buscam avaliar a taxa de infiltração dos *socialbots* nesta mídia social. Durante o estudo, as vulnerabilidades das OSNs são discutidas e exploradas, em função dos autores construírem a sua própria SbN no Facebook para avaliação da taxa de infiltração. As vulnerabilidades listadas neste artigo foram importantes para esta dissertação, pois contribuiram parcialmente na etapa da construção dos perfis dos robôs sociais.

Davis et al. (2016) discorrem sobre o BotOrNot, um serviço publicamente disponível e amplamente conhecido com milhares de recursos para avaliar contas do Twitter. Esta análise é feita através da comparação de semelhança deste perfil com as características conhecidas dos *socialbots*. Para classificar uma conta sendo de um robô social ou de uma pessoa, a ferramenta foi treinada com ambas as classes. Para treinamento dos *socialbots*, eles precisaram utilizar uma lista de Lee et al. (2011) que os identifica.

Já Ji et al. (2014) propõem uma nova abordagem de detecção de comportamento *socialbot* no *host* final. Como os binários sociais ou os códigos-fonte não são facilmente acessíveis, primeiramente eles criaram um nova *social botnet*, conhecida como Wbbot, analisaram sua arquitetura e comportamento.

Posteriormente, observaram o comportamento das máquinas diante da manipulação de SbNs provenientes de sites públicos, de modo que as informações coletadas pudessem ser utilizadas como um meio de avaliação de metodologia proposta. Em função dos resultados obtidos, uma das conclusões foi que a maioria dos robôs sociais imitam atividades do usuário legítimo (JI et al., 2014).

Kumar e Reddy (2012) usam técnicas de classificação como *Support Vector Machine*, *Nave Bayes* e árvores de decisão para classificar os perfis como falsos ou genuínos. Por apresentarem um método de detecção automática, acreditam na facilidade da sua aplicação no contexto das OSNs, em função da classificação manual dos perfis ser custosa.

Durante o estudo descrevem também o FIS (*Facebook Immune System*), ferramenta do Facebook que verifica em tempo real cada clique e todas as operações de leitura e gravação realizadas através dele. Essa realiza cerca de 25 bilhões de avaliações diárias e até 620.000 verificações por minuto no pico, como ocorreu em maio de 2011. Para aplicar as técnicas de detecção, os autores de Kumar e Reddy (2012) utilizam base de dados para treinamento.

No artigo Ghanadi e Abadi (2014) é apresentado o SocialClymene, um sistema de

reputação negativa baseado em *PageRank*, capaz de identificar as *botnets* Stego. Essas usam imagens compartilhadas em uma rede social para enviar os comandos do *botmaster* e receber as informações roubadas de usuários infectados. Com base na avaliação da reputação negativa, a ferramenta desenvolvida em Ghanadi e Abadi (2014) analisa as imagens compartilhadas por usuários de redes sociais e calcula uma pontuação de reputação negativa para cada um, a partir do histórico de participação deles em atividades de grupos suspeitos.

Tiwari (2017) considera que a engenharia social é a principal causa de ameaças a qualquer OSN, sobretudo porque os perfis falsos buscam tornar-se amigos de usuários legítimos com um objetivo final de obter acesso às informações privilegiadas. São apresentadas quatro tipos de ameaças básicas previstas nas OSNs, enumeradas como clássica, moderna, combinadas e visando crianças. Embora haja a categorização, somente é exemplificado a ameaça clássica, onde perfil falso é criado usando o detalhes pessoais de um usuário legítimo ou com dados de um usuário não real.

Em função das ameaças destacadas em Tiwari (2017), é explicitado que as OSNs se preocupam com isso, pois possuem soluções internas para detecção de atividades falsas e *spam*, tal como o FIS no Facebook. Desta forma, os autores de Tiwari (2017), discutem métodos úteis de aprendizagem de máquinas para a criação de perfis, revisam métodos de identificação dos *socialbots* concomitantemente com a análise da *Online Social Network* sob uma perspectiva multi-agente.

Como os perfis falsos aprimoram suas técnicas de engenharia social e são capazes de realizar atividades com o objetivo de influenciar na popularidade de uma pessoa ou de um produto em divulgação, Tiwari (2017) identifica que a detecção baseada em conteúdo é um passo de aperfeiçomento da pesquisa, a fim de demonstrar melhor desempenho em termos de falsos positivos e falsos negativos.

2.2 CRIAÇÃO DE *SOCIALBOTS* E *SOCIAL BOTNETS*

Os autores Jin et al. (2013) executam ataques à privacidade baseados em amigos em comum em uma rede social. Assim, descobriram que a partir deles, um usuário malicioso é capaz de lançar ataques de privacidade que identificam amigos e vizinhos distantes de um usuário específicio. Também analisam as várias estruturas de ataque passíveis de utilização para a construção de estratégias para explorá-lo.

Uma análise das vulnerabilidades de uma OSN para a criação de *social botnet* é realizada em Boshmaf et al. (2013). Os perfis usados são capazes de executar dois tipos

de operações genéricas em qualquer OSN: interação social, que são operações que usadas para ler e escrever conteúdo social; as operações na estrutura social, que são utilizadas para alterar o grafo social.

Para os perfis realizarem tais ações, os autores criam uma ferramenta que mantém o *login* das contas ativos constantemente. Eles se apoiam principalmente nestas duas abordagens: a API OSN e modelos de solicitação HTTP. Isso porque somente com a API do Facebook, não foi possível executar algumas atividades (BOSHMAF et al., 2013).

Após a criação dos *socialbots* e da arquitetura proposta em Boshmaf et al. (2013), os autores executaram os testes por oito semanas, alcançando uma taxa de infiltração de 80% de sucesso de contas falsas no Facebook. Também é observado que a simulação mostra que um invasor usando apenas um nó atacante pode identificar mais de 60% dos amigos de um usuário.

Apesar do resultado obtido, a criação dos *socialbots* com capacidade de publicar conteúdos de usuários legítimos selecionados aleatoriamente, a fim de imitar o comportamento humano, facilita na detecção desses, visto que nem sempre os interesses ou as ações de um *socialbot* são coerentes. Inclusive porque suas publicações podem ser contraditórias.

Já Compagno et al. (2015) empregam uma abordagem com uso de *malware* para construção de *bots* no Facebook e GooglePlus. Desta forma, para tornar estes robôs ativos na *botnet* Elisa, eles devem ser utilizados por uma vítima ao interagir com alguma OSN. Esta *botnet* usa esteganografia para esconder seus comandos dentro das mensagens das vítimas, tornando assim o conteúdo publicado mais confiável para os amigos das mesmas.

Zhang et al. (2017) utilizam contas legítimas no Twitter como *socialbots*. Para a construção da *social botnet*, eles compraram contas no Twitter e desenvolveram o *botmaster* em Java, utilizando o protocolo OAuth e a própria API desta mídia social. Assim, foi possível executar todas as operações do Twitter em nome de todos os robôs sociais de modo que esses apresentassem comportamentos similares ao usuários legítimos.

Em He et al. (2015) é previsto que um *rootkit* de automação de teste web (WTAR) seja um meio para a concepção de *socialbots* maliciosos. Eles implementaram estes robôs sociais em algumas mídias sociais (Facebook, Twitter e Weibo) e validaram a ameaça. Para isso, analisaram os comportamentos dos protótipos em um ambiente de laboratório e na Internet. Também acompanharam os relatórios dos antivírus amplamente utilizados.

Além dos *socialbots* de He et al. (2015) serem baseados em WTAR, independente da rede social, também podem imitar comportamento humano através de um automatizador de testes. Entretanto, a pesquisa mostrou limitação, visto que ao utilizar um *malware*

como base da pesquisa e caso a máquina fosse desligada, a inicialização deveria ser manual.

Como o comportamento humano e o conteúdo gerado são simulados por uma ferramenta de automatização de testes sem a preocupação de uma alta quantidade de requisições ativar algum mecanismo de segurança da OSN, caso seja mapeado o tempo de execução das ações, é possível identificar que essas não foram feitas por um humano. Portanto, acredita-se que uma possível abordagem que contemple uma metodologia para a criação dos *socialbots* torne o comportamento mais próximo ao de um humano.

Mesmo as mídias sociais aprimorando suas ferramentas de detecção de comportamentos anômalos e os *socialbots* evoluindo de modo que sejam cada vez mais parecidos com os humanos, entre os trabalhos analisados no contexto de criação de *socialbots* e de *social botnets*,não foi observada uma definição de metodologia que abordasse uma lógica para a construção do perfil dos robôs sociais, quando os mesmos eram construídos pelos pesquisadores. A indicação de como foi construído este processo ressalta os pontos nos quais as pesquisas relacionadas à detecção de *social botnet* são recomendadas a considerar.

2.3 *DATASETS* NA ÁREA DE SEGURANÇA DA INFORMAÇÃO

Em Abt e Baier (2014) é retratada a problemática de escassez de dados rotulados em *databasets* públicos, normalmente utilizados na área de Segurança para aprendizado de máquina supervisionado em busca de investigar padrão de comportamento. Os autores analisaram alguns *datasets* e especificaram seus problemas. Entre os avaliados, estão:

- *Cooperative Association for Internet Data Analysis* (CAIDA) (CAIDA, 2018),*Protected Repository for the Defense of InfrastructureAgainst Cyber Threats* (PREDICT) (PREDICT, 2018), MOME (*Monitoring and Measurement database*) (MOME, 2018) e WITS (*Waikato Internet Traffic Storage*) (WITS, 2018), apresentam dificuldade de mapear a relação entre os dados devido ao anonimato das informações, como o dos IPs (ABT; BAIER, 2014);

- DARPA (DARPA, 2018), cujos dados são antigos e não retratam fenômenos das redes atuais; (ABT; BAIER, 2014);

- repositório de dados formados para estudo específico de um grupo, como o "*The Simpleweb Traffic Traces Data Repository*", criado pela *Design and Analysis of Communication Systems* (DACS)(ABT; BAIER, 2014);

A fim de entender como a comunidade de segurança lida com esta falta de rótulos,

Abt e Baier (2014) realizaram um estudo sistemático dos artigos presentes nas principais conferências de segurança no período compreendido entre 2009 e 2013.

Após a análise, concluiu-se que 70% dos artigos revisados dependem de conjuntos de dados gerados manualmente, apenas 10% dos artigos estudados liberam os dados criados, 44% deles tendem a utilizar conjuntos de dados externos da indústria (não são liberados também).

Estes índices sinalizam que a comunidade de Segurança da Informação está enfrentando um problema relacionado à falta de dados rotulados. A fim de ser capaz de resolver este problema, uma das sugestões do autor é o compartilhamento dos *datasets* descritos (ABT; BAIER, 2014), uma das contribuções esperadas desta pesquisa.

Kumar e Reddy (2012) é outro exemplo de artigo que evidencia a limitação dos *datasets* na área da Segurança da Informação. Isso porque, seus autores precisavam de uma base de dados rotulada para aplicar as técnicas de detecção com base em treinamento. Entrentato, as informações presentes no *dataset* fornecido pelo Laboratório Barracuda eram restritas por não apresentarem o conteúdo das ações, somente a quantidade de vezes que algumas atividades ocorreram (por exemplo, número de curtidas ou de amigos), então os atributos definidos inicialmente tiveram que ser alterados, o que de certo modo pode ter impactado nos resultados obtidos.

Nos experimentos de Jin et al. (2013) foi necessário o uso de *dataset* para o ataque baseado em amigos. Assim, utilizou-se um conjunto de dados com vários atributos sintéticos de usuários do Facebook. Isso porque o *dataset* original, cuja fonte não é indicada, continha apenas as redes de amigos. Estes atributos sintéticos foram embasados em diferentes *clusters* de usuários gerados a partir do grau de um perfil, através de diferentes abordagens de agrupamento. Com base no cenário apresentado, acredita-se que a existência de uma base de dados com atributos bem definidos auxiliaria no estudo.

Em Ghanadi e Abadi (2014) utilizaram-se dois conjuntos de dados, um de relacionamento sociais entre usuários de Facebook (MCAULEY; LESKOVEC, 2012) e outro de imagens do Flicker. Uma limitação do *dataset* é sua construção, em função de ser constituído apenas de "círculos"(ou "listas de amigos") do Facebook (MCAULEY; LESKOVEC, 2012). A mesma restrição de conteúdo é vista na base de dados gerada durante a construção do artigo Catanese et al. (2011), no qual a mesma é construída utlizando a perspectiva de grafos.

Já Tyagi e Aghila (2012) demonstram a eficácia e as vantagens de explorar uma *botnet social* para a distribuição de spam e manipulação por meios digitais, através de experimentos reais no Twitter. Também são realizadas simulações orientadas por rastreamento

para comparar o desempenho da distribuição de spam entre os métodos independentes e os utilizados por uma *botnet*. Mesmo com as simulações mapeadas, não é relatada a construção ou a disponibilização de um *trace* com estas informações.

Ji et al. (2014) avaliam o desempenho do sistema desenvolvido durante a pesquisa através dos dados gerados por *socialbots* com *traces* reais. Para coletar a informação necessária para avaliar a metodologia de detecção, os autores tiveram acesso aos binários e código fontes dos robôs sociais de modo que pudessem executá-los, para capturar as requisições que seriam utilizadas para avaliação do método de detecção proposto. Embora os autores tivessem demonstrado interesse no compartilhamento do *dataset*, os atributos coletados não foram divulgados, somente são indicadas as taxas de identificação.

3 METODOLOGIA PROPOSTA

Este capítulo discorre sobre a metodologia proposta para construir uma base de dados classificada, constituída pelas ações dos *socialbots* no Facebook. Como mostrado no Capítulo 2, os trabalhos relacionados existentes não definem uma lógica associada ao comportamento dos perfis dos *socialbots* quando observados como um usuário do Facebook, e essa é uma etapa importante para infiltração da *social botnet* na mídia social, este estudo mostra a abordagem a ser utilizada.

Assim, este capítulo tem como objetivo discorrer sobre as etapas de construção de uma *social botnet* até a geração e categorização do *dataset*, demonstrando as dificuldades e os meios de realização de cada uma das etapas. Desta forma, a seguir, serão expostas as metodologias desenvolvidas para:

- Criação das contas dos *socialbots*;
- Definição dos perfis dos *socialbots*;
- Geração do *dataset*;

3.1 CRIAÇÃO DAS CONTAS DOS *SOCIALBOTS*

Ao criar uma nova conta de usuário no Facebook, um endereço de email ou número de celular são necessários primeiro para validar e, em seguida, ativar a conta através de um email ou inserção de um código recebido por mensagem para a confirmação da criação da conta. Após a confirmação, seu proprietário a ativa e define suas preferências seguindo uma URL de ativação enviada por e-mail ou por mensagem. Consequentemente, para criar os *socialbots*, é preciso superar estes obstáculos. Em relação aos emails, existem as seguintes soluções possíveis:

- uso de emails temporários;
- utilização de provedores comuns que não limitam o número de contas de email criadas por sessão de navegação ou endereço IP (por exemplo, MailRu[4]);

[4]https://e.mail.ru/login

- aplicação de provedores, por exemplo ProntonMail[5], que não solicitam celulares para atestar a criação da conta, pedindo assim apenas as credenciais do usuário e um email para recuperação, sendo o email um item não obrigatório;

- emprego de fornecedores clássicos que solicitam celular para confirmar a criação de conta, como Yahoo ou Gmail;

As duas primeiras abordagens foram também descritas em Boshmaf et al. (2013), e ainda funcionam até hoje. A terceira foi analisada durante o estudo e foi vista sua efetividade durante a execução do mesmo. Enquanto a última foi empregada para dificultar a detecção em função do amplo uso destes provedores.

Como Boshmaf et al. (2013) exemplifica apenas 10MinuteEmail[6] e MailRu sendo uma solução temporária e um provedor comum sem limitação na quantidade de contas respectivamente, para definir qual solução de email temporário seria aplicada, na Tabela 3.1 foram avaliadas algumas ferramentas.

TAB. 3.1: Avaliação do efetividade de uso de soluções de emails no Facebook.

Ferramenta	Atuação sobre o Facebook
https://temp-mail.org/	Não funciona
https://tempail.com/	Funciona
https://app.inboxbear.com/	Não funciona
https://pt.emailfake.com/	Não funciona
https://www.guerrillamail.com/	Não funciona
https://10minutemail.com/	Não funciona
https://www.mohmal.com/	Funciona
https://www.crazymailing.com/	Não Funciona

Também houve a tentativa de utilizar números de celular temporários. Esses são números não associados a nenhum indivíduo específico, obtidos através de sites que fornecem o serviço de recebimento de mensagem. Algum sites, como Receive SMS Online [7] e Receive Free SMS Online[8], fornecem este tipo de serviço. Entretanto, quando os celulares temporários foram utilizados para criar contas, o Facebook os identificou facilmente e os classificou como inválidos. A única solução de celular temporário testada que funcionou bem com o Facebook foi o MyTrashMobile [9]. Esta ferramenta disponibiliza números de

[5]https://protonmail.com/
[6]https://www.10minutemail.com/10MinuteMail/index.html?dswid=8568
[7]http://receive-sms-online.com/
[8]http://receivefreesms.com/
[9]https://pt.mytrashmobile.com

telefone individuais para os usuários, ou seja, mais ninguém os poderá utilizar. Esse é o diferencial da ferramenta em relação às demais utilizadas.

Uma alternativa foi usar números celulares já existentes, acessíveis e com donos reais, onde havia a possibilidade de confirmar a conta. Este passo foi fundamental para facilitar o acesso dos *socialbots* ao usuários legítimos, pois nos casos em que o número havia sido usado em uma conta verdadeira, ao atribuí-lo como pertencente a um perfil falso, o Facebook automaticamente indicava este perfil do robô social a alguns amigos da conta original ou sugeria que o mesmo adicionasse tais amigos.

Esta recomendação realizada pelo Facebook contribui para a execução de engenharia social. Já que facilita a inserção dos *socialbots* nesta mídia, pois ao indicá-los para usuários legítimos, fornece uma sensação maior de credibilidade. E, consequentemente, quando um usuário verdadeiro adiciona o perfil do *socialbot*, isso contribui para que os amigos do indivíduo real aceitem a solicitação de amizade ou adicionem o perfil falso, pois fornece maior confiabilidade a esse robô social.

Em relação à ativação da conta do Facebook, observou-se que mesma não era impeditiva para o uso do perfil, pois o usuário dono desta conta conseguia adicionar outros usuários e realizar as demais ações normalmente. Então não houve um preocupação em automatizar este processo.

3.2 DEFINIÇÃO DOS PERFIS DOS *SOCIALBOTS*

Com o objetivo de inserir com maior facilidade os *socialbots* no Facebook, buscou-se criar uma lógica para a criação dos perfis, algo que não foi observado nos trabalhos relacionados a este tema, e levantar os perfis dos usuários brasileiros na Internet, sobretudo nas mídias sociais. Desta forma, foram utilizadas como base as informações presentes em Alex (2015), com os seguintes dados estatísticos segmentados por:

a) Idade: A maior parte dos usuários das mídias sociais estão na faixa etária entre os 15 e 44 anos (ALEX, 2015), conforme a Tabela 3.2.

b) Sexo: O uso das mídias sociais por homens é praticamente igual à taxa de mulheres, já que eles representam 50,5% dos usuários no Brasil (ALEX, 2015).

c) Localidade: A região Sudeste corresponde à quase metade dos usuários do país atuantes nas Mídias Sociais (ALEX, 2015). Em função disso, levando em consideração os dados presentes na Tabela 3.3, as vinte cidades mais populosas do Brasil

TAB. 3.2: Distribuição de usuários das Mídias sociais no Brasil a partir da faixa etária. (ALEX, 2015)

Faixa Etária	Taxa
menos de 15 anos	17%
15 até 24 anos	22,4%
25 até 34 anos	23,2%
35 até 44 anos	20,9%
45 até 55 anos	11,6%
mais de 55 anos	4,9%

(EXAME, 2014) foram distribuídas aleatoriamente entre as regiões as quais pertencem.

TAB. 3.3: Distribuição de usuários das Mídias sociais com base nas regiões do Brasil. (ALEX, 2015)

Região	Taxa
Norte	4,2%
Nordeste	15,9%
Centro-Oeste	9,9%
Sudeste	49,7%
Sul	20,2%

A geração dos atributos dos perfis falsos foi realizada de modo independente para cada um dos critérios apresentados acima. Isso ocorre porque existe uma dificuldade inerente de encontrar os dados unificados, *e.g.* a porcentagem das idades com relação à cada estado.

Desta forma, para a criação das contas falsas, o responsável por controlar os *socialbots* define as credenciais do Facebook. Os dados destes usuários, como nome, foram criados com base em IBGE (2010), no qual mostra os vinte nomes femininos e masculinos mais comuns categorizados pelas décadas. Já em relação aos sobrenomes, foi usado PROCOB (2015), onde lista os trinta sobrenomes mais comuns no país, que em alguns casos foram combinados de forma aleatória entre si.

Os emails foram criados utilizando como base o nome ou possível apelido daquele nome, separado por ".", com nomes do meio abreviados, último sobrenome escrito por inteiro e os dois últimos dígitos do ano de nascimento. Como pode ocorrer do email já existir, quando este cenário ocorre, repete-se os dois números até não existir mais conflito.

Já em relação à distribuição por mês de nascimento no qual o indíviduo nasceu, como a base de dados de DATASUS (2015) não aponta dados anteriores à 1994, foram escolhidas

as informações mais recentes. Desta forma, foram calculadas as porcentagens com base no total de 3.017.668 brasileiros nascidos em 2015, agrupados por mês do nascimento. O resultado deste cálculo pode ser observado na Tabela 3.4.

TAB. 3.4: Porcentagens dos brasileiros nascidos em 2015 e agrupados por mês de nascimento

Mês	Taxa de Porcentagem	
Janeiro	8,4%	
Fevereiro	8%	
Março	9,2%	
Abril	8,7%	
Maio	8,9 %	
Junho	8,4%	
Julho	8,4%	
Agosto	8,1%	
Setembro	8,3% Outubro	8%
Novembro	7,6%	
Dezembro	8%	

Observou-se quea maior parte nasceu em Março, seguido de Maio e Abril. Em contrapartida, Novembro foi o mês com menor taxa de nascimento. Os meses de Janeiro, Junho e Julho ficaram empatados. O mesmo ocorreu em Fevereiro, Outubro e Dezembro.

Após o cadastro, em alguns perfis foi incluída uma foto pessoal, antecedentes e interesses. Os antecedentes e interesses foram definidos a partir de gostos populares ou regionais. Com base em Thomas e Nicol (2010), a fim de disfarçar-se como uma conta legítima, os perfis mais aprimorados se juntam a vários grupos sociais a partir de palavras-chave que representem referências populares, como grupos que discutem sobre alguma vertente política.

3.3 GERAÇÃO DO *DATASET*

A escassez de *datasets*, principalmente os que apresentam os dados completos e classificados, limita as pesquisas em segurança da informação. Por exemplo, os autores de Kumar e Reddy (2012) precisaram alterar os parâmetros inicialmente definidos no estudo em função da quantidade restrita de atributos do *dataset* que utilizaram para detectar *social botnet*.

Desta forma, a base de dados gerada nesta dissertação busca incluir informações presentes no *dataset* utilizado em Kumar e Reddy (2012), como amigos de um mesmo gênero ou número de curtidas, e também as ações que até então não tinham sido mapeadas

(postagem, comentário, entre outras), a fim de que o pesquisador que utilizar o *dataset* possa manipulá-lo de modo que consiga extrair as informações necessárias e relevantes para sua pesquisa. O modo como as ações serão disparadas será melhor apresentado no Capítulo 4, que discorre sobre a arquitetura proposta e implementada.

Como o *dataset* gerado pode ser aplicado no contexto de identificação de uma *social botnet*, foi desenvolvido um meio para facilitar o processo de aprendizado dos algoritmos de detecção. Esse consiste principalmente em identificar as ações de um conjunto de *socialbots* no momento de execução dos testes como sendo pertencentes a um determinado cenário. Entende-se neste estudo que um cenário é constituído por um conjunto de ações de um ou mais *socialbots* em uma mídia social. Em seguida, estes cenários são classificados de acordo com os níveis de dificuldade de detecção dos *socialbots* que atuam neles.

A aplicação do conceito "cenário"nesta pesquisa foi uma abordagem criada em função da dificuldade de encontrar trabalhos relacionados que agrupam os dados semanticamente. Esta dificuldade ocorre principalmente por conta da limitação na quantidade de *datasets* disponíveis. Além disso, a falta de descrição das informações contidas nas bases de dados existentes, sobretudo no contexto do Facebook, implicam que não é utilizada uma abordagem por categorização em cenários. Entretanto, existe uma técnica de engenharia de software, descrita em Bai et al. (2002), que explica este processo de classificação. Nesta técnica, é realizada uma modelagem baseada em cenários, que são capazes de capturar a funcionalidade do sistema em vários níveis de abstração. Foi com base nos conceitos descritos pelos autores de Bai et al. (2002) que o termo "cenário"foi adaptado e definido neste estudo.

Foi necessária executar tal adaptação porque no modelo exposto pelos autores de Bai et al. (2002), os cenários são organizados hierarquicamente e descrevem as funcionalidades do sistema em vários níveis de abstração, tais como: grupos de cenários, constituitídos por um ou mais cenários; sub-cenários, que são a decomposição de um cenário, representando uma subfunção fornecida pelo software; entre outros. Além disso, este modelo indica diferentes perspectivas (vista funcional, visualização de dados e vista de uso) para o sistema, nos quais cada uma busca entender o relacionamento dos seus respectivos componentes entre si (BAI et al., 2002).

Entretanto, a forma proposta pelos Bai et al. (2002) não é adequada ao problema de descrição do *dataset* por atuarem em escopos bastante distintos. Em consequência disso, o termo cenário exprimido neste estudo denota outra abordagem, já que ele é constituido das ações dos *socialbots*.

De modo semelhante ao artigo Bai et al. (2002), esta pesquisa considera também as

interações e os relacionamentos, mas estes itens apontados são aplicados entre os *socialbots*. É válido ressaltar que a forma como a definição do termo "cenário" foi estabelecida, embasou o processo de categorização por cenários, que será descrito a seguir.

3.3.1 CATEGORIZAÇÃO POR CENÁRIOS

Para discorrer sobre a categorização por cenários, antes será necessário formalizar alguns conceitos. Os cenários representam um conjunto de uma ou mais ações, nos quais elas podem ou não reproduzir as atividades de um usuário legítimo. Isso porque nem sempre um *socialbot* imita um perfil verdadeiro, já que a quantidade de atividades que ele é capaz de fazer está associado ao seu nível de aprimoramento. Quando não ocorre uma reprodução de ações de algum perfil, o robô social manifesta uma identidade e conteúdos de ações próprios, com suas preferências, tal como área de trabalho ou gosto musical.

As ações representam as funcionalidades que um usuário consegue executar no Facebook como: curtir, compartilhar ou postar algum conteúdo (texto, foto ou link); comentar ou responder algum comentário; adicionar ou recusar amigos; adicionar fotos; entre outros.

Nos casos onde as ações retratam a replicação das atividades de um determinado perfil, elas são feitas de forma completamente automática, através da inserção de um arquivo de entrada que contém informações, como contéudo publicado e data de execução, relacionadas às ações realizadas pelos perfis do Facebook.

Em contrapartida, há a possibilidade das ações serem executadas de forma semi-automática, *i.e.* tanto os procedimentos são realizados quanto os conteúdos são definidos manualmente pelo controlador da *social botnet*. A publicação destas operações continua sendo efetuada de modo automático em todos os *socialbots* selecionados (ativos) para aquele contexto. Um detalhamento maior sobre como tecnicamente as ações são executadas será descrito no Capítulo 4. Já em relação aos cenários, esses foram definidos a partir dos seguintes critérios:

- as atividades que os *socialbots* realizam na mídia social. Essas são similares às ações dos usuários legítimos;
- tipos de perfis (*socialbots* ou perfis legítimos) com quem os componentes da *social botnet* interagem;
- comportamento dos *socialbots* que compõe um determinado cenário. Um exemplo de comportamento é o tempo levado para aceitar solicitações de amizade;

Conforme já explicitado, este trabalho tem como contribuição a disponibilização de uma arquitetura conceitual para geração de *dataset* no contexto de OSN, que poderá ser aplicado para treinamento de algoritmos de detecção de *social botnet*. A fim de facilitar o processo de aprendizagem, o conteúdo do *dataset* foi categorizado por cenários. Esses, por sua vez, foram classificados em função dos níveis de dificuldade na identificação dos *socialbots* que o compunham. Assim, para classificá-los, foram utilizados cinco níveis de abstração: muito fácil, fácil, médio, difícil e muito difícil.

As métricas foram definidas com base em Hingston (2010), onde os autores projetaram e organizaram a BotPrize, uma competição de programação. Durante a disputa, os competidores desenvolvem *bots* que são avaliados em função do seu nível de aprimoramento, ou seja, o objetivo é os *bots* não conseguirem ser distinguidos dos jogadores humanos. Esta disputa é uma versão de *bot* do Teste de Turing. Entretanto, ao invés de ser um teste de inteligência do computador como o teste proposto por Alan Turing (TURING, 2009), o BotPrize é uma prova da capacidade do robô parecer um ser humano.

Fundamentados em Hingston (2010), onde também é indicado que a inteligência artificial de nível humano é necessária para treinamento realista usando simulação, os níveis de dificuldade são determinados a partir da porcentagem de erro na classificação dos *socialbots* pertecentes a um determinado cenário. Esta classificação deverá ser realizada por um grupo de pessoas que se consideram usuárias assíduas do Facebook. Entende-se que um erro ocorre quando o grupo de pessoas selecionadas para identificar os *socialbots* criados neste estudo o classificam como um perfil legítitmo.

Desta forma, os níveis de abstração são aplicados aos cenários na etapa do experimento que resulta no *dataset*, a partir da porcentagem de erro na detecção por humanos dos robôs sociais que os constituem. De acordo com a porcentagem de erro, os cenários são classificados da seguinte forma:

TAB. 3.5: Classificação dos cenários por níveis de dificuldade e taxa de erro

Níveis de Dificuldade	Taxa de erro
Muito difícil	Entre 81% e 100%
Difícil	Entre 61% e 80%
Médio	Entre 41% e 60%
Fácil	Entre 21% e 40%
Muito fácil	Entre 0% e 20%

Um ponto a ser destacado sobre a classificação realizada por humanos é que esta abordagem não é escalável. Entretanto, como não foi observada na área acadêmica alguma

ferramenta efetiva para classificar perfis de Facebook como falsos ou verdadeiros, esta abordagem não pôde ser utilizada.

Com o objetivo de determinar como seria realizado o processo de classificação, foi feito um estudo prévio, independente dos experimentos, sem o armazenamento das ações. Para isso, foram utilizados cem perfis falsos apenas com capacidade de compartilhar um evento publicado no perfil de um usuário verdadeiro.

Ao solicitar para um grupo de vinte usuários do Facebook classificá-los, os quais quinze se descreviam como usuários assíduos do Facebook e cinco utilizavam a ferramenta raramente, observou-se que após quinze a vinte perfis, os participantes classificavam qualquer uma das contas, mesmo sem avaliação, como falsas, pois o teste ficava cansativo.

Em função de não ter sido mesclado com uma listagem de usuários legítimos, isso impactou nos resultado, pois a taxa de detecção foi de 75%, cujo o grupo formado por usuários constantes do Facebook identificou com 100% de acerto as contas falsas. Contudo, como a taxa de erro entre até os vinte primeiros perfis a serem avaliados por usuários bastantes atuantes no Facebook foi nula, confirmou-se que utilizar esta abordagem contribui para a classificação dos cenários.

A partir do que foi observado, constentou-se que, para aprimorar a etapa de classificação: além de misturar perfis legítimos com os dos *socialbots*, era também necessário colocar uma quantidade total da amostra, formada pelos dois tipos de perfis, inferior à quinze. Como um mesmo *socialbot* pode atuar em diferentes cenários, a cada finalização de algum experimento, esta etapa de classificação deverá ser executada.

Além disso, como os comportamentos dos *socialbots* contribuem para facilitar ou dificultar na identificação dos mesmos, foi definido que o grau de dificuldade máximo será atribuído ao cenário onde os *socialbots* são mais aprimorados, fazendo com que taxa de erro na detecção superior à 80%.

3.3.1.1 DEFINIÇÃO DE COMPORTAMENTO DOS *SOCIALBOTS*

O comportamento que os *socialbots* manifestam contribui para dificultar ou facilitar a detecção dos mesmos. Por conta disso, foi realizado um estudo, em busca de mapear previamente estes costumes. Tiwari (2017) exemplificam-se práticas que auxiliam na identificação dos perfis falsos, como o uso de imagens existentes em algum local na Web. Isso contribui na identificação porque existem sites, por exemplo o https://www.tineye.com/, capazes de mapear o local onde aquela imagem aparece *online*. Desta forma, este estudo

utiliza imagens disponíveis em repositórios públicos, *e.g. Public Domain Photos*[10] e também fotos privadas com autorização prévia do dono, ambas selecionadas manualmente.

Existem outras pesquisas que contribuem na definição das possíveis ações a serem realizadas pelos *socialbots*. Exemplifica-se isso através do levantamento feito pela empresa de segurança Barracuda Networks, que mostra que 43% de usuários falsos nunca atualizam as informações nem postam no mural e 97% dos perfis falsos se identificam como mulheres bonitas e superpopulares (LABS, 2012). É válido indicar que a porcentagem associada às mulheres não é aplicada nesta pesquisa, devido à lógica utilizada na etapa de definição dos perfis dos *socialbots*, descritas na Seção 3.1.

Tiwari (2017) lista diferentes tipos de ameaças básicas previstas nas OSNs e discorre sobre a ameça clássica. O autor considera o ataque de clonagem da identidade como um exemplo deste tipo de ameaça. Como o nome deste ataque sugere, nele um perfil falso é criado usando o detalhes pessoais de um usuário honesto, ou seja, o perfil falso mascara o usuário real. Outra alternativa é criar um perfil fictício para o qual o usuário real não existe.

Existe também o caso de criar perfis com as mesmas características dos perfis criados no estudo prévio descrito nesta seção. As contas de Facebook criadas utilizaram nomes aleatórios ou compostos por caracteres especiais ou símbolos; sem fotos ou com fotos de animais ou paisagens; sem amigos ou com poucos amigos.

Uma outra forma de introduzir os *socialbots* no Facebook é através de inclusão de antecedentes e interesses nos seus perfis (THOMAS; NICOL, 2010). Várias informações relacionadas aos usuários, como os comportamentos (duração de viagens, entre outros), o nível de educação (ensino médio, ensino super e pós graduação) e a descrição do trabalho podem ser obtidas através da manipulação da ferramenta de négocios desta mídia social (BUSINESS, 2015).

No relatório disponibilizado pelo próprio Facebook, é possível observar os principais assuntos em conversas ao longo de 2017. Tais assuntos são representados como "*topics to watch*"e "*hot topics*". O primeiro são tópicos de conversas que cresceram consistentemente ao longo de 12 meses. Já o segundo, são os assuntos de conversa onde o volume de citação é desproporcionalmente alto em um determinado período. (IQ, 2017)

Por meio de tais informações e também dos tópicos mais populares (*hot topics*), os *profiles* mais aprimorados podem infiltrar-se em vários grupos sociais. Em função dos diversos aspectos listados sobre as questões comportamentais dos perfis falsos, as mesmas ações serão aplicadas na fase de experimento, que consiste na criação do *dataset* através do

[10]http://www.public-domain-photos.com/

armazenamento das atividades executadas pelos *socialbots* e seus respectivos conteúdos.

4 CRIAÇÃO E IMPLEMENTAÇÃO DA ARQUITETURA CONCEITUAL

Este trabalho possui como um dos seus objetivos a criação de uma arquitetura conceitual que abrange desde a etapa de construção de uma *social botnet*, o que inclui a criação dos *socialbots* que a constituem, até a geração e armazenamento das ações executadas pelos robôs sociais.

Desta forma, definiu-se uma arquitetura conceitual em busca de permitir a manutenção do *dataset* por outros pesquisadores. Isso porque as tecnologias utilizadas foram descritas somente durante a implementação da arquitetura conceitual, ou seja, a arquitetura conceitual proposta permite o uso de quaisquer ferramentas.

Como a arquitetura conceitual foi definida independente da tecnologia utilizada para sua implementção, foi possível principalmente propor uma abordagem que busca melhorar o cenário de escassez de dados, sobretudo dos classificados ou rotulados, na área de Segurança da Informação. Como esta área é bastante abrangente, a arquitetura proposta está associada ao contexto de *social botnets* no Facebook.

Outro aspecto a ser discutido neste capítulo é a etapa de implementação desta arquitetura conceitual, por se tratar de um passo necessário para a disponinilização da base de dados descrita, uma das constribuições neste estudo. Desta forma, neste capítulo é apresentada a arquitetura conceitual e a sua implementação.

4.1 ARQUITETURA CONCEITUAL

Para construir a arquitetura conceitual capaz de gerar uma base de dados constituída das atividades e dos conteúdos gerados pelos *socialbots*, foi realizada uma divisão semântica entre os seus componentes, que reflete os papéis que cada um deles desempenha, conforme pode ser observado na Figura 4.1.

O primeiro item (ou retângulo), está relacionado a todo contexto que envolve a criação de perfis, ou seja, contempla as atividades de: criação de emails, contas do Facebook, execução das metodologias desenvolvidas para a criação das contas e definição dos perfis dos *socialbots*.

Como esta etapa exige um trabalho diretamente proporcional ao tamanho da *social botnet* a ser criada, sugere-se o uso de automatizadores capazes de preencher campos de página web, para que a criação de contas seja de um modo totalmente ou parcialmente

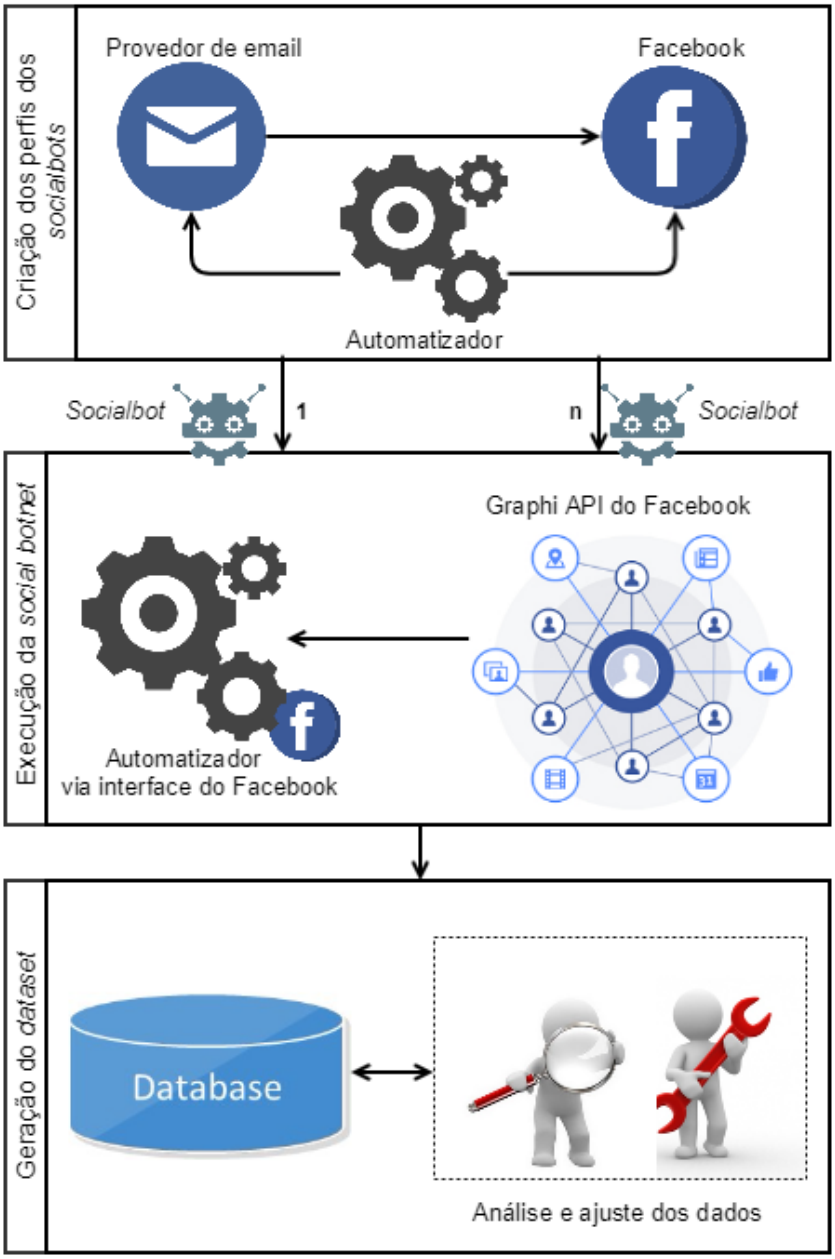

FIG. 4.1: Representação da Arquitetura Conceitual.

automático. A automação completa, por exemplo, ocorre quando o provedor de email utilizado não exige algum tipo de confirmação ou a implementação da arquitetura criada resolve qualquer tipo de validação que possa surgir.

Existem também situações onde a automação é parcial, em função de existir algum tipo de confirmação associada à legitimidade do usuário, como ocorre quando há o envio de mensagem para o celular do mesmo, na expectativa que ele insira um código na página web, confirmando assim que não é um robô.

Quando ocorre este tipo de validação, é necessário construir uma solução aprimorada e genérica, capaz de resolver isso independente do perfil, o que pode ser muito custoso em alguns casos, sobretudo se houver a possibilidade de usar provedores de email que não exigem tal validação.

Ainda há a automatização relacionada à mídia social a ser explorada, que no caso deste estudo, é o Facebook. Esta OSN realiza diferentes tipos de confirmação de um perfil, de acordo com o fornecedor de email. No caso do Outlook, Yahoo e Gmail, o processo é através de uma URL (*Uniform Resource Locator*) a ser acessada por email. Com outros provedores, como Prontmail, é enviado um código que deve ser digitado pelo usuário.

Já quando a conta no Facebook é criada utilizando um número celular, também ocorre o envio de um código por SMS (*Short Message Service*) para ser digitada na página web. Entretanto, como esta confirmação não era impeditiva para o uso do perfil, pois o usuário conseguia adicionar os usuários e realizar as demais ações normalmente, não houve a automatização desta etapa do processo de criação de perfis no Facebook.

A partir das opções de automatização mostradas, este estudo aborda solução automática, quando o fornecedor de email não exige validação, e parcialmente automática, quando o provedor exige inserção de códigos enviados por mensagem ou resolução de *captchas*.

Já em relação ao item "Execução da *social botnet*", após o levantamento dos trabalhos relacionados, foi concluído que um *socialbot* consiste em dois componentes principais: um perfil de uma OSN e o *software* que automatiza a forma como ele interage com os outros usuários. A quantidade de *socialbots* que compõem uma *social botnet* pode variar de 1 até n, sen. Desta forma, esta etapa representa a execução da *social botnet* através do gerenciamento centralizado e automatização das ações dos *socialbots*.

Inicialmente, etapa da arquitetura conceitual relacionada à exploração pela *social botnet* resumia-se a uma aplicação web que funcionaria sobre a API do Facebook, denominada Graph API[11] , com a comunicação entre ambas as partes através de requisições do protocolo HTTP. Estas requisições utilizariam uma linguagem reconhecida pelo *browser*, como JavaScript. O uso desta linguagem ainda representa uma facilidade, já que possibilita o uso de funções existentes no *Software Development Kit*[12] (SDK) do próprio Facebook.

O fato desta etapa estar configurada de acordo com a Figura 4.2 torna possível atender aos requisitos de adicionar um usuário do Facebook para ser gerenciado pelo sistema,

[11] http://developers.facebook.com/docs/graph-api?locale=pt_BR
[12] https://developers.facebook.com/docs/javascript

postar, comentar, responder comentários e curtir conteúdos da *timeline* do usuário gerenciado pela aplicação. Os conteúdos a serem utilizados nas ações podem ser gerados de modo manual ou automático.

No primeiro caso, o conteúdo é gerado manualmente pelo controlador da *social botnet*, que aplica os comportamentos dos *socialbots* mapeados na Subseção 3.3.1.1 do capítulo 3. Já o segundo, é realizado através do agendamento das execuções das atividades inclusas num arquivo de entrada. No caso deste estudo, o arquivo a ser utilizado foi fornecido pelo Ferreira (2018), que o gerava através do criação de um aplicativo usado por usuários do Facebook. Portanto, a implementação da arquitetura conceitual satisfaz também a geração de conteúdo automático.

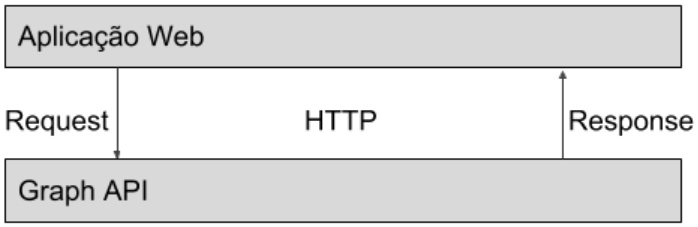

FIG. 4.2: Fase inicial da implementação da Arquitetura Conceitual.

No entanto, com o aprimoramento dos *socialbots* através de inclusão de novas atividades a serem exploradas no próprio Facebook, houve a necessidade de implementar funcionalidades não atendidas (embora não seja indicada esta limitação) pela Graph API, como por exemplo: postar, comentar e curtir conteúdos em lugares diferentes à *timeline* do próprio usuário gerenciado pela aplicação; adicionar, excluir e aceitar solicitação de amizade; e agendar postagens baseadas em arquivos de entrada fornecidos ao sistema.

Para atender estes requisitos, houve a necessidade de acrescentar uma outra camada na etapa de execução de ações da *social botnet*. Esta camada atualmente também é responsável por executar várias das funcionalidades antes feitas pela Graph API, fazendo com que esta API fosse utilizada apenas para obtenção de dados relativos ao usuário, como foto do perfil, ID do usuário no sistema do Facebook e e-mail. Assim, a segunda parte da arquitetura conceitual, com a camada nova (delineada pelo retângulo azul), consiste na Figura 4.3.

Esta camada foi adicionada de modo a permitir que as demais partes do projeto pudessem evoluir independentemente umas das outras. Optou-se, portanto, por seguir uma arquitetura orientada a serviços, desenvolvendo uma API REST. Essa é uma interface que irá prover acesso às ações que um usuário pode executar no Facebook, seguindo os

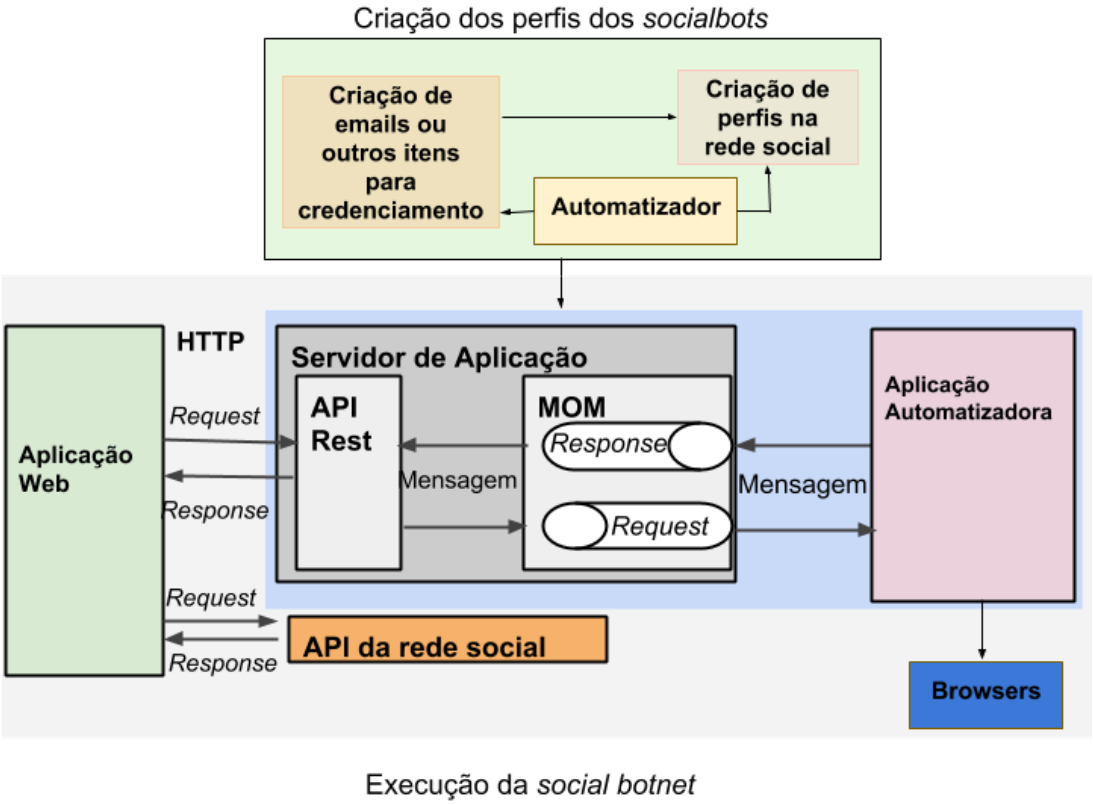

FIG. 4.3: Estruturação da etapa de execução da *social botnet* da Arquitetura Conceitual.

padrões do HTTP definidos em Fielding (2000), no qual um recurso é identificado através: da URI (*Uniform Resource Identifier*), que é o identificador; e os métodos, como *get*, *post* ou *delete*, do HTTP (*Hypertext Transfer Protoco*), que indicam o que será feito com estes recursos.

Nesta arquitetura, a API REST é a responsável por receber as requisições HTTP da interface web, enviá-las a um MOM (*Message Oriented Middleware*). Ele é capaz de receber e armanzenar a mensagem enviada de um lugar e de repassá-la para um outro lugar. Desta forma, ele envia o pedido à uma aplicação automatizadora responsável por processar o pedido e respondê-lo de forma adequada. Este automatizador é quem acessa o browser para executar as ações dos *socialbots*.

Portanto, através desta arquitetura, é possível que o controlador da *social botnet* defina os comandos (ações a serem executadas pelos *socialbots*) utilizando a Aplicação Web. É por meio de uma requisição HTTP que o servidor de aplicação entrega esta solicitação à aplicação automatizadora, para que essa seja responsável por executar as ações dos robôs sociais no browser.

Outro aspecto desta arquitetura é que a mesma não apresenta acoplamento de nenhuma das partes do sistema, para que caso seja necessário qualquer tipo de mudança no

futuro, ela poderá ocorrer sem que as demais partes sejam afetadas, como por exemplo, uma troca de tecnologias utilizadas no sistema.

Por fim, a última parte consiste na geração do *dataset* classificado, uma das constribuições deste estudo. Através desta etapa, ao mapear todas as ações dos *socialbots* para um determinado cenário, a aplicação automatizadora é responsável por armazená-las. Assim, esta base de dados poderá ser disponibilizada para as demais pesquisas que atuem num contexto similar.

4.2 IMPLEMENTAÇÃO

Para viabilizar a implementação da arquitetura conceitual proposta, foram utilizadas algumas ferramentas que serão apresentadas a seguir. A Figura 4.4 ilustra a proposta de implementação da arquitetura conceitual.

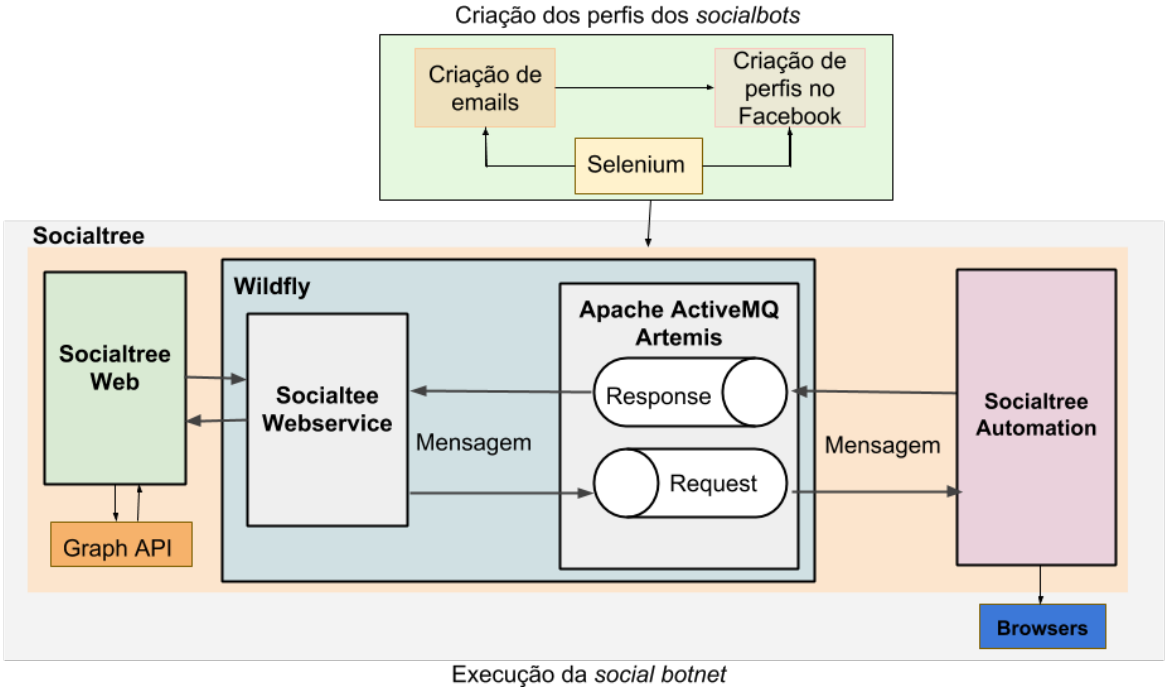

FIG. 4.4: Proposta de implementação da arquitetura conceitual.

Conforme mostra a Figura 4.4, o processo de implementação é composto de duas principais fases: a primeira é formada pela etapa de "criação dos perfis dos *socialbots*", já a segunda junta os demais componentes da arquitetura conceitual ("execução da *social botnet*"e "geração do *dataset*").

A primeira etapa da implementação foi formada pelo Selenium[13]. Ele é um conjunto de ferramentas utilizado para automatizar navegadores web, usado principalmente para

[13]http://www.seleniumhq.org/

automatizar testes de aplicações web. Entretanto, ele não se limita a isso, já que pode ser usado para automatizar qualquer tipo de tarefa efetuada em um navegador (PROJECT, 2012).

O Selenium tem o apoio de alguns dos maiores fornecedores de navegadores, que buscam tornar esta ferramenta uma parte nativa de seus respectivos navegadores (RAFAEL, 2015). É também a tecnologia principal em inúmeras outras ferramentas de automação de navegador, APIs e frameworks, possuindo bibliotecas para linguagens como Java, C#, Ruby, Python e JavaScript (RAFAEL, 2015).

Através do uso desta ferramenta, foi possível automatizar a etapa de criação das contas de email ou dos perfis do Facebook. Entretanto, em situações onde ocorriam algum tipo de validação, o processo referente à validação teve que ser feito manualmente, como no caso das verificações que ocorrem após a criação de emails providos por fornecedores clássicos (Gmail, Yahoo ou Outlook) ou de perfis de usuários no Facebook.

Já a segunda etapa da implementação consiste na criação da Socialtree, responsável por realizar as atividades da *socialbot*, gerar *datasets* e agrupar as ações em cenários. Esta aplicação é feita em Java, JavaScript, HTML, CSS e diversas ferramentas que serão descritas no decorrer deste capítulo. Como é composta por uma estrutura mais complexa, os seus componentes de *frontend* (Socialtree Web) e *backend* (Socialtree Webservice e Socialtree Automation) serão descritos separadamente nas seções a seguir.

4.2.1 SOCIALTREE WEB

A composição arquitetural da aplicação web foi montada de acordo com as necessidades impostas a partir das possíveis atividades a serem realizadas por um usuário no Facebook. Desta forma, o Socialtree Web, representa a interface de gerenciamento dos *socialbots* indicada na figura 4.5,

Por meio do Socialtree Web é possível: ativar/desativar perfis; incluir/excluir as contas dos robôs sociais da ferramenta de gerenciamento; informar qual ação será executada e o conteúdo que será aplicado durante a execução da mesma; e selecionar o cenário associado a uma determinada ação. Através deste componente é possível fazer os perfis dos *socialbots* executarem as atividades presentes na Tabela 4.1.

O Socialtree Web é composto por várias camadas, representadas pelas bibliotecas das ferramentas, que se comunicam entre si através de chamadas de funções JavaScript. Este componente do Socialtree foi feito também utilizando JavaScript, HTML e CSS. Segue abaixo o detalhamento e função que cada uma das ferramentas que compõe esta interface

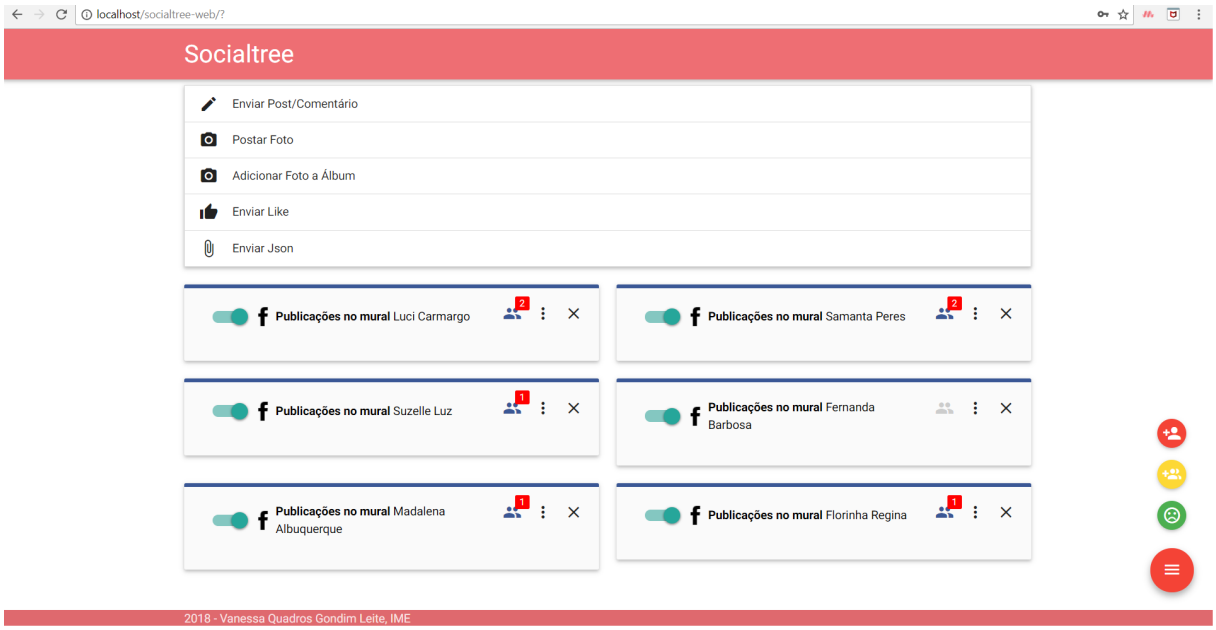

FIG. 4.5: Interface de gerenciamento do Socialtree Web.

TAB. 4.1: Atividades dos perfis dos *socialbots* gerenciadas pelo Socialtree Web.

Atividades	No próprio perfil	Em Perfis de Terceiros
Postar conteúdo	Sim	Sim
Curtir conteúdo	Sim	Sim
Comentar conteúdo	Sim	Sim
Responder comentário	Sim	Sim
Adicionar foto em um álbum	Sim	Não se aplica
Adicionar ou excluir amigos	Sim	Não se aplica
Aceitar ou recusar solicitação de amizade	Sim	Não se aplica

web:

- **Servidor Apache**[14]**:** Utilizado para armazenar e publicar o sistema, ou seja, ele hospeda o Socialtree Web, de modo que ele possa ser acessado através de uma URL.

  Este projeto é um esforço colaborativo de desenvolvimento de software destinado a criar uma implementação de código fonte robusto, de nível comercial, com recursos completos e disponível gratuitamente de um servidor HTTP (Web) (APACHE, 1997).

  O servidor HTTP Apache foi lançado em 1995 e tem sido o servidor web mais popular na Internet desde abril de 1996 (APACHE, 1997), sendo esses os principais motivos de sua escolha.

[14]https://httpd.apache.org/

- **Backendless**[15]**:** é um serviço web que disponibiliza *backend* como serviço (baas), armazenamento de arquivos e API REST de acesso aos dados do *backend*. A Plataforma Backendless é fortemente integrada, projetada para agilizar e acelerar o processo de desenvolvimento de aplicativos. Ela, como um todo, estabelece uma abordagem unificada para a construção de aplicações de um modo rápido, o que faz com que o custo de desenvolvimento seja reduzido e com a alta confiabilidade.

  No Socialtree Web, a biblioteca do Backendless foi usada no *frontend* para consumir os dados do banco. Foi necessário fazer isso para recuperar do banco de dados os cenários e a suas respectivas descrições. Ambos são definidos no Socialtree Web à medida que os experimentos são realizados. Desta forma, um conjunto de atividades executadas tornam-se pertencentes a um determinado cenário. Além disso, a criação de um cenário é realizada pela interface do Socialtree Web, conforme mostra a figura 4.6. Como consequência, o nome e a descrição de um cenário são armazenados no Backendless.

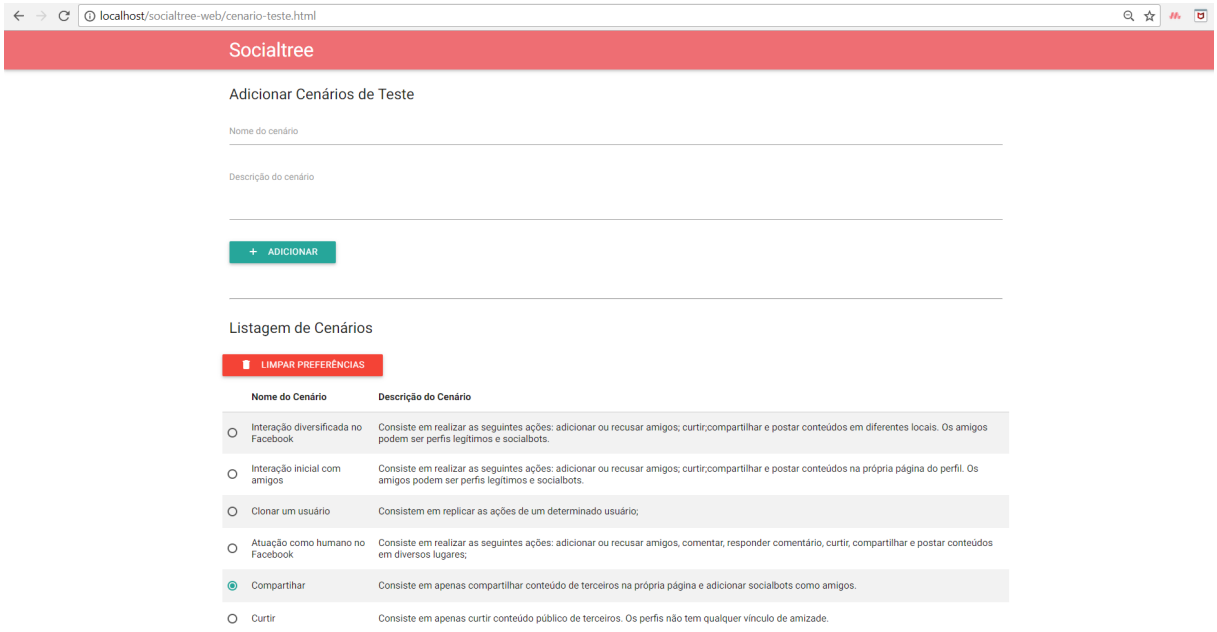

FIG. 4.6: Interface do Socialtree Web para seleção de cenário.

- **Pusher**[16]: é um serviço web que disponibiliza software como serviço (saas), dentre os quais, um sistema de canais de comunicação com *websockets*. Foi escolhido por ter uma API de acesso às suas funcionalidades tanto na linguagem JavaScript quanto na linguagem Java.

[15]https://backendless.com/
[16]https://pusher.com/

O Pusher foi utilizado para possibilitar o envio de mensagens e comunicação assíncrona do Socialtree Web com o Socialtree Automation, servindo como solução para notificações de recebimento de solicitações de amizade. Isso porque o Socialtree Automation, que ainda será descrito mais a frente, faz a verificação de novas solicitações de amizade.

Como o Automation é parte do *backend* do Socialtree e, em função dele precisar enviar mensagem para o Socialtree Web, foi necessário criar um canal para tal comunicação, sendo o Pusher o responsável por desempenhar esta finalidade. Para acontecer esta comunicação, o Socialtree Web utiliza uma biblioteca do Pusher para receber mensagem do *backend*.

- **Materialize**[17]**:** é um framework *front-end* moderno e responsivo baseado em *Material Design*[18]. Foi utilizado como base para a identidade visual do Socialtree Web e foi escolhido por ser fácil de integrar e rápido de utilizar.

- **SDK JavaScript Facebook:** Como explicado anteriormente na seção que descreve a arquitetura conceitual, o SDK JavaScript do Facebook consiste em um conjunto de funções em JavaScript que possibilita dentre outras coisas, acesso fácil ao *login* do Facebook e à Graph API do Facebook. Foi utilizado para obter informações, como e-mail e foto do perfil, do usuário que for inserido no sistema Socialtree para ter o seu perfil gerenciado pela ferramenta;

A partir das ferramentas supracitadas, o Socialtree Web apresentou a construção em camadas representada pela Figura 4.7.

4.2.2 SOCIALTREE WEBSERVICE

O sistema Socialtree Webservice é feito em Java e é responsável por fazer a intermediação entre a interface web e o Socialtree Automation, o verdadeiro responsável por fazer a implementação e automatizar as ações dos *socialbots* não disponíveis através da Graph API. Para o webservice disponibilizar uma API Rest, pois é através dela que é permitido o acesso às ações que um usuário realiza no Facebook, são utilizadas seguintes ferramentas:

- **Wildfly**[19]**:** é um servidor de aplicação que permite *deploy* de aplicações do padrão JEE (*Java Platform, Enterprise Edition*), disponibilizando implementações de cada

[17]http://materializecss.com/
[18]https://material.io/
[19]http://wildfly.org/

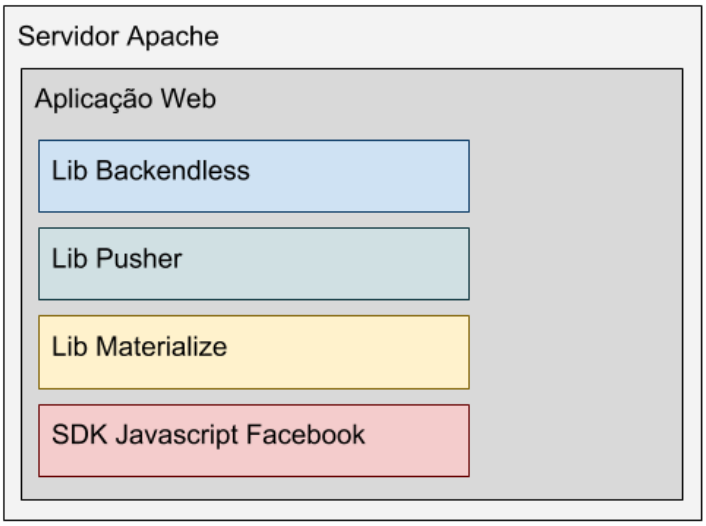

FIG. 4.7: Representação do Socialtree Web em camadas.

uma das especificações desse padrão. Inclusive, ele contém por padrão uma implementação interna de um MOM (Apache ActiveMQ Artemis), uma vez que há um JMS (*Java Message Service*). O JMS é uma especificação ou conjunto de regras do Java EE, para definir como deve ser tratada a questão de envio e recebimento de mensagens (mensageria).

Portanto, o Wildfly foi escolhido por se tratar de uma solução robusta e conhecida e por disponibilizar todo o ambiente necessário para o desenvolvimento da aplicação em questão.

- **Apache ActiveMQ Artemis[20]:** É uma implementação de *Broker* MOM para a especificação JMS (*Java Message Service*) do Java EE. Está contido dentro Wildfly e é utilizado por este servidor de aplicação para mensageria entre sistemas.

  Portanto, o Apache Artemis é o responsável por: gerenciar os dois canais de comunicação, um para envio das mensagens do Webservice para o Automation, e outro de resposta, que envia mensagens do Automation para o Webservice; receber, armazenar e entregar as mensagens vindas da API Rest do sistema. Tanto a criação quanto o gerenciamento dos canais ocorre através da interface de administração do Wildfly.

  A principal função do Apache Artemis neste estudo é integrar a aplicação Socialtree Webservice com a Socialtree Automation, buscando garantir a manutenção das

[20]https://activemq.apache.org/artemis/

mensagens, mesmo quando o Socialtree Automation não estivesse disponível ou o Apache Artemis fosse desligado.

É válido ressaltar que embora o Pusher também sirva como canal de comunicação, ele e o Apache Artemis são utilizados em contextos distintos. Já que o primeiro é uma canal entre o Socialtree Automation e o Web, já que ele envia notificação do *backend* para o *frontend*. O segundo serve como canal de mensageria entre o Socialtree Webservice e o Automation.

4.2.3 SOCIALTREE *AUTOMATION*

Este componente é o último a ser abordado em relação à implementação da arquitetura conceitual. Foi feito em Java e é o responsável por automatizar as ações com o navegador e executá-las nos perfis do Facebook. Neste componente está toda a lógica responsável por realizar as atividades no Facebook.

Nesta etapa, inicialmente, são criadas 5 instâncias (janelas) de navegadores que esperam a inserção de usuários a serem gerenciados pela interface web. Cada uma destas instâncias é utilizada por um único usuário enquanto o mesmo se mantiver ativo na ferramenta e é liberada quando o mesmo deixar de ser. Um *socialbot* se mantém ativo quando ele está habilitado no Socialtree Web para realizar as atividades no Facebook.

À medida que novos usuários são gerenciados, ao atingir a quantidade de 5 *socialbots* logados, o sistema se encarrega de criar novas instâncias, a fim de preparar e já deixar disponível as janelas do browser a serem utilizadas pelos robôs sociais a serem inseridos. Além disso, cada uma destas instâncias é acessada através de um semáforo binário que permite apenas uma ação ser executada por vez em cada uma delas de modo a manter a consistência dos dados.

Cada ação executada na aplicação Socialtree Web efetua uma requisição para a API Rest (Socialtree Webservice), que por sua vez envia o pedido para o MOM Apache ActiveMQ Artemis. Já o Socialtree Automation, através do Apache Camel, recupera estas mensagens enviadas ao MOM e fica responsável por realizar:

- o roteamento da mensagem ao método de processamento da ação correspondente de acordo com o conteúdo da mensagem (*content-based router*). Esse é identificado através de um cabeçalho definido na mensagem enviada pela API Rest no formato chave-valor. A chave é definida como "*action*"e o valor refere-se ao nome da ação que se deseja executar. O conteúdo das ações é enviado no corpo da mensagem;

- processar e automatizar no navegador;
- persistir as informações relativas a ação executada;
- gerar a resposta para a API, que por sua vez notifica a interface web;

Além disso, também é de responsabilidade do Socialtree Automation realizar a cada um minuto a verificação nos navegadores abertos em relação: à recepção de respostas de pedidos de amizade feitos pelos perfis dos *socialbots* gerenciados; à solicitação de amizades por terceiros a estes perfis; à qualquer outra notificação, recebida pelos perfis gerenciados pela interface web (Socialtree Web).

Estas notificações são resultantes de: convite para curtir páginas ou ir para eventos; respostas aos comentários em *posts* (textuais ou de fotos) feitos na página (*feed*) do próprio perfil ou em grupos específicos; reações (curtir, entre outras), menções e respostas a comentários onde foi feita alguma citação. Elas são persistidas na tabela "*notify_interaction*", que é descrita no capítulo 5. Para implementar esta gama de funcionalidades, o Socialtree Automation utiliza as ferramentas a seguir:

- **Selenium:** foi escolhido para automatizar tarefas no Facebook não cobertas pela API da rede social, tais como, adicionar ou remover amigos e interagir (curtir, comentar ou compartilhar) conteúdo de terceiros.
- **Quartz** [21]**:** é uma biblioteca de agendamento de tarefas em código aberto, que pode ser integrada em aplicativos Java. Por tarefas, entende-se qualquer ação descrita por um código em Java, por exemplo, a atualização de um banco de dados, envio de e-mails ou geração de log de sistemas, nos quais tais tarefas são executadas de acordo com o agendamento feito pelo programador, podendo ser num exato momento temporal ou em períodos cíclicos.

  O Quartz foi selecionado por possibilitar o agendamento de tarefas futuras, como efetuar uma publicação com base em dados fornecidos previamente por um arquivo de entrada. Este arquivo de entrada é colocado pelo controlador da *social botnet* na interface web (Socialtree Web) através de upload e é enviado ao Socialtree Webservice. Toda requisição que o Webservice recebe é enviada para o Apache Artemis.

  Como o Apache Artemis atua como canal de mensageria, o Apache Camel (uma das ferramentas do Socialtree Automation) o consome. Por sua vez, a lógica desenvolvida e implementada no Socialtree Automation é a responsável por agendar uma

[21] http://www.quartz-scheduler.org/

ação através do uso do Quartz. Assim, quando chegar a data agendada, o Quartz aciona o Socialtree WebService, para que a ação seja executada.

- **Apache Camel**[22]**:** é um *framework* de integração de código-fonte aberto que torna a integração de sistemas simples. Utilizando um integrador é possível se dedicar em como interoperar com o sistema, ao invés de se preocupar em como o sistema que está sendo integrado realmente. O Apache Camel provê simplicidade e abstrações gerenciáveis para sistemas complexos e, devido a estas características, é considerado um excelente *framework* de integração.

  Além disso, o Apache Camel torna a integração nos projetos de software produtiva pois, por ser robusto, facilita o desenvolvimento ao realizar controles internos ao reduzir a quantidade de linha de código e o uso de bibliotecas externas. Isso porque seu núcleo é um motor de roteamento que funciona a partir de uma DSL (*Domain Specific Language*) e tem como base os padrões de integração (*Enterprise Integration Patterns* - EIPS). Isso permite de forma simples e rápida que o desenvolvedor defina a fonte dos dados, as rotas pelas quais os dados passarão e pelas quais serão processados, e, por fim, o destino para onde os mesmos dados irão.

  Devido aos motivos supracitados, o Apache Camel foi apontado para prover de forma fácil uma integração entre o MOM Apache ActiveMQ Artemis e o Socialtree Automation, já que é o Apache Camel quem faz o consumo das mensagens do Apache Artemis. Além disso, o Apache Camel direciona as mensagens para suas rotas. Entende-se por rota a lógica criada durante este estudo para o Apache Camel desempenhar uma função bem definida, ou seja, a lógica desenvolvida para realização de ações específicas no Facebook.

  Durante o direcionamento, o Apache Camel considera o tipo de mensagem que recebeu do Artemis. Esta diferenciação entre as mensagens tambem foi algo criado neste estudo e foi realizada através do cabeçalho, onde é indicado o tipo de ação no Facebook a ser realizada.

- **Backendless:** escolhido para realizar a persistência dos dados (ações dos usuários), das notificações e dos cenários gerados pela aplicação Socialtree, por sua facilidade de utilização e integração, foco em produção ágil e disponibilizar bibliotecas em JavaScript e Java, que foram as linguagens utilizadas no desenvolvimento.

[22]http://camel.apache.org/

- **Pusher:** fornece uma biblioteca para o Socialtree Automation enviar mensagem para o Socialtree Web, sinalizando que existe uma notificação de solicitação de amizade. Isso é feito pra que o controlador da *social botnet* possa ver a solicitação de amizade. Desta forma, é possivel aceitar a amizade num processo mais otimizado.

5 EXPERIMENTOS

Para a geração do *dataset* classificado, as metodologias apresentadas no capítulo 3, desenvolvidas para a criação das contas dos *socialbots*, para a definição dos perfis dos *socialbots* e para a geração do *dataset*, foram aplicadas nesta parte do estudo. Com isso, foi exposta a descrição dos experimentos, responsável por apresentar a organização dos experimentos e como é empregue a metodologia do capítulo 3. Posteriormente, indicou-se a execução dos experimentos.

Outra parte explicitada é a comparação entre o *dataset* gerado neste estudo e os demais *datasets* com informações referentes à esta mídia social. Além disso, também é possível observar as características do Facebook analisadas durante os experimentos.

5.1 DESCRIÇÃO DOS EXPERIMENTOS

Uns dos objetivos desta pesquisa são a criação de uma *social botnet* e o armazenamento das atividades e conteúdo gerados por ela, de modo que isso resulte na geração do *dataset*. Para gerá-lo, é executado um conjunto de experimentos, que é o resultado da execução de ações no Facebook pelos *socialbots* através dos perfis falsos.

Com base nisso, os experimentos foram divididos em etapas principalmente para validar as ações a serem realizadas no Facebook, também descritas na Tabela 4.1 do capítulo 4. As etapas foram definidas da seguinte forma: validação da *social botnet*, que não gera o *dataset* a ser disponibilizado e nem categoriza as ações dos *socialbots*; e geração de *dataset*, que aplica a categorização por cenários.

Devido a esta separação do experimento, as etapas utilizaram perfis falsos distintos. Isso porque na geração do *dataset* seria interessante que houvesse o armazenamento desde a execução das primeiras ações para que os dados armazenados fossem os mais completos possíveis e representassem a realidade dos perfis dos *socialbot*. Entretanto, não havia esta restrição para a validação da *social botnet*, já que esta etapa consistia em avaliar a efetividade da infiltração da *social botnet*, ou seja, não apresentava como foco o armazenamento das ações dos *socialbots*.

Ainda em relação às etapas do experimento, na validação da *social botnet*, foram criados cerca de 20 perfis sem a aplicação de alguma metodologia. Em contrapartida, para executar os experimentos que resultam na geração do *dataset*, foram feitos 60 *socialbots*,

sendo que 10 foram construídos sem o uso de uma metodologia específica e com características que facilitam bastante a sua identificação, *e.g.* caracter especial no nome e fotos de animais ou paisagens, e 50 foram elaborados com uma lógica. É válido ressaltar que na segunda etapa do experimento, houve a preocupação de manter ativa a mesma quantidade inicial de perfis, *i.e.* se os 60 perfis criados fossem bloqueados, seriam recriados 60 perfis com a mesma metodologia aplicada nos anteriores para que os experimentos fossem ser realizados e para que pudesse ser feita a identificação dos *socialbots* pelo grupo de pessoas selecionadas para realizar esta atividade.

O uso de lógica nos 50 perfis teve como base a seção 3.2, que consiste em definir os perfis dos *socialbots* para facilitar a sua inserção no Facebook, e a subseção 3.3.1.1, que exemplifica comportamento que os robôs sociais manifestam que dificultam ou facilitam a detecção dos mesmos. Dentre os robôs sociais criados com uma semântica, eles foram separados em 26 homens e 24 mulheres, distribuídos da seguinte forma:

- Localização: Distribuição representada na Tabela 5.1.

TAB. 5.1: Distribuição por região no experimento.

Região	Homem	Mulher
Norte	1	1
Nordeste	4	4
Centro-Oeste	3	2
Sudeste	13	12
Sul	5	5

- Faixa Etária: Distribuição representada na Tabela 5.2.

TAB. 5.2: Distribuição por faixa etária no experimento.

Faixa Etária	Homem	Mulher
menos de 15 anos	4	4
15 até 24 anos	6	5
25 até 34 anos	7	6
35 até 44 anos	5	5
45 até 55 anos	3	3
mais de 55 anos	1	1

- Meses de aniversário: Distribuição representada na Tabela 5.3

Como os dados estatísticos utilizados para gerar as tabelas supracitadas apresentavam casas decimais e eram utilizados para determinar quantidade de perfis dos *socialbots*,

TAB. 5.3: Distribuição por meses de aniversário no experimento.

Mês	Homem	Mulher
Janeiro	2	2
Fevereiro	2	2
Março	3	3
Abril	3	2
Maio	3	2
Junho	2	2
Julho	2	2
Agosto	2	2
Setembro	2	2
Outubro	2	2
Novembro	1	1
Dezembro	2	2

foi realizado arredontamento das casas decimais para que nas tabelas existissem somente números inteiros. Devido ao arredondamento, existiu a preocupação de respeitar a semântica, *i.e.* manter a representatividade dos itens de cada tabela, mantendo nos campos com maiores porcentagens uma maior quantidade de perfis e nos itens com menores porcentagens, uma menor quantidade de perfis. Assim, por exemplo, Novembro e Março continuaram sendo o meses com menor e maior quantidade de pessoas respectivamente.

Em relação à cidade e ao estado, a distribuição foi realizada com base na quantidade de pessoas por região. Após esta etapa da criação dos perfis dos *socialbots*, para as ações realizadas por eles pudessem ser categorizadas de um modo que o resultado final fosse a geração do *dataset* classificado, foi implementada a metodologia apresentada na subseção 3.3.1. Desta maneira, os cenários foram definidos em função dos tipos de ações em que um ou mais *socialbots* poderiam executar no Facebook e dos vínculos de amizade que apresentavam:

- Cenário "Curtir":
  - Ação: curtir conteúdo público de terceiros;
  - Tipos de amigos: sem amigos;
  - Características dos perfis: sem foto; nomes podem ser escritos normalmente, ou representados através de apelidos ou com caracteres especiais;
- Cenário "Compartilhar":

  - Ações: compartilhar conteúdo público de terceiros na própria página e adicionar amigos;
  - Tipos de amigos: *socialbots*;
  - Características dos perfis: com foto sendo imagens aleatórias retiradas de repositórios públicos;

- Cenário "Interação inicial com amigos":
  - Ações: adicionar ou recusar amigos, curtir, compartilhar e postar conteúdos na própria página do perfil;
  - Tipos de amigos: perfis legítimos e *socialbots*;
  - Características dos perfis: erro de digitação ao postar conteúdo; uso de fotos de pessoas que deram permissão para o seu uso ou retiradas de repositórios públicos; conteúdo postado somente em sua *timeline*;

- Cenário "Interação diversificada no Facebook":
  - Ações: adicionar ou recusar amigos, curtir, compartilhar e postar conteúdos em diversos lugares;
  - Tipos de amigos: perfis legítimos e *socialbots*;
  - Características dos perfis: eles começam a interagir melhor com os usuários do Facebook de um modo geral, ao disponibilizar conteúdo em diversos lugares;

- Cenário "Atuação como humano no Facebook":
  - Ações: adicionar ou recusar amigos, comentar, responder comentário, curtir, compartilhar e postar conteúdos em diferentes locais;
  - Tipos de amigos: perfis legítimos e *socialbots*;
  - Características dos perfis: eles começam a agir como usuários legítimos; postam conteúdos diversos, sobretudo os relacionados à temas atuais; comentam e respondem comentário;

- Cenário "Clonar um usuário":
  - Ações: replicação das atividades de um determinado perfil legítimo a partir de um agendamento;
  - Tipos de amigos: perfis legítimos e *socialbots*;

– Características dos perfis: suas ações são agendadas e limitadas às ações referentes ao usuário a ser replicado;

5.2 EXPERIMENTO

Os experimentos foram divididos em duas etapas: validação da *social botnet* e de geração do *dataset*. A primeira foi executada durante o ano 2017 e com duas rodadas de testes, onde a duração em dias não ultrapassava duas semanas. Existiu esta quantidade de períodos não contínuos porque a cada ajuste na arquitetura implementada, os testes eram realizados novamente.

Durante o período de validação, foram utilizados cerca de 20 perfis falsos. Observou-se que os perfis mais atraentes, sobretudo mulheres, apresentavam mais solicitações de amizade e interações. Por exemplo, em um perfil feminino com faixa etária inferior à 30 anos, 101 pedidos de solicitação de amizade foram aceitos em 1 dia e em apenas 4 minutos, 83 aceites de pedidos foram realizados sem qualquer tipo de bloqueio pelo Facebook. Além disso, este mesmo perfil apresentava cerca de 200 perfis adicionados em cerca de 3 dias e também foi sugerido um encontro por um usuário, onde um usuário aparentemente legítimo repassou o seu número de celular. Este comportamento pode ser observado na figura 5.1.

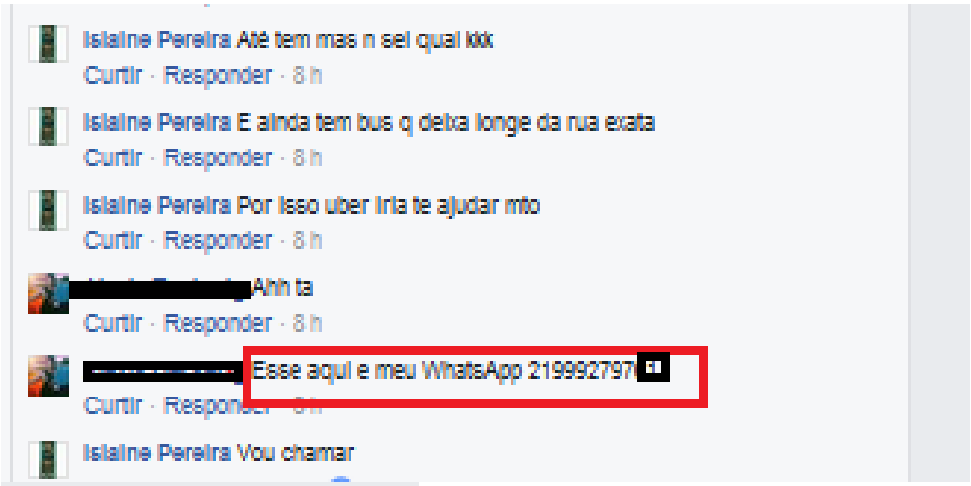

FIG. 5.1: Demonstração de informação fornecida por um usuário a um *socialbot*.

Na Figura 5.1, um usuário com a identificação ofuscada na cor preta envia dados pessoais para o *socialbots*. Um outro usuário identificado na cor vermelha interage também com este mesmo robô social.

Além disso, um perfil que executava ações no Facebook, criado com email temporário e sem foto, permaneceu ativo por mais de um ano. Isso foi observado porque o mesmo

permaneceu acessível após a primeira etapa dos experimentos. Portanto, através da execução das ações dos *socialbots* foi possível observar a efetividade da infiltração da *social botnet* construída neste estudo no Facebook, independente da geração do *dataset*, pois foi possível criar e manter a quantidade definida de perfis dos *socialbots* para a realização dos experimentos que resultam no *dataset*.

Já a segunda etapa, que consistia na geração do *dataset* através das ações dos *socialbots*, durou cerca de uma semana e ocorreu em 2018. Os testes que geraram o *dataset* final, ou seja, o que será disponibilizado. Os experimentos mantiveram-se utilizando constantemente 60 perfis, *i.e.*, qualquer perfil que fosse bloqueado, a mesma metodologia aplicada nele era mantida no perfil a substituí-lo. Nesta fase, foram realizados os testes a partir de seis cenários distintos: "Curtir", "Compartilhar", "Interação inicial com amigos","Interação diversificada no Facebook","Atuação como humano no Facebook"e "Clonar um usuário". Neste contexto, um experimento corresponde a um cenário.

Desta forma, estes experimentos foram organizados de acordo com a Tabela 5.4, onde: a primeira coluna representa os experimentos atribuídos aos cenários; a segunda indica a quantidade máxima de *socialbots* ativos na ferramenta durante a execução do experimento; e a terceira indica o modo de criação dos *socialbots* para o experimento, ou seja, se houve emprego de alguma metodologia descrita no capítulo 3 ou não.

TAB. 5.4: Organização dos experimentos.

Experimento	Quantidade máxima de *socialbots* ativos	Modo de construção
Curtir	10	Sem metodologia descrita
Compartilhar	10	Com metodologia descrita
Interação inicial com amigos	10	Com metodologia descrita
Interação diversificada no Facebook	10	Com metodologia descrita
Atuação como humano no Facebook	10	Com metodologia descrita
Clonar um usuário	10	Com metodologia descrita

Durante a segunda etapa do experimento, observou-se que até mesmo os perfis sem fotos e com variedade de ações bastante limitada, recebiam solicitações de amizade. Embora tais perfis não pudessem aceitá-las por este tipo de atividade não estar inclusa no cenário em que o mesmo está enquadrado. Após a execução da segunda etapa do experimento, obteve-se como resultado a geração do *dataset*. Conforme explicitado no capítulo

3, a geração deste *dataset* inclui também a classificação do mesmo através do uso de cinco níveis de abstração: muito fácil, fácil, médio, difícil e muito difícil.

Para realizar a classificação, foram escolhidas 10 pessoas que se consideram usuárias assíduas do Facebook, numa faixa etária de 20 e 60 anos. Esta quantidade foi baseada em McAuley e Leskovec (2012), onde foi realizada uma pesquisa com esta mesma quantidade de pessoas, onde elas deveriam identificar manualmente todos os círculos aos quais seus amigos pertenciam.

À medida que os testes referentes a um cenário eram finalizados, este grupo de pessoas classificava uma quantidade de 15 perfis como legítimos ou falsos, sendo que 10 eram os perfis utilizados pelos *socialbots* e 5 eram de usuários verdadeiros. Com os valores obtidos, buscou-se encontrar métricas calculadas individualmente para cada um dos cenários.

Após analisar se a identificação de cada um dos perfis dos *socialbots* que compunham cada um destes cenários estava errada, comparavam-se as respostas com o gabarito, para assim obter um valor e, referente à quantidade de respostas erradas em um cenário específico. Este cálculo foi feito para cada uma das catalogações dos *socialbots*. Posteriormente, para encontrar a média de erro das categorizações dos *socialbots*, foi realizado o somatório com base na quantidade p de pessoas responsáveis por identificar os perfis falsos. O resultado foi utilizado para encontrar a função $M(c)$, que representa média de erro para cada um dos cenários, através da fórmula

$$M(c) = \frac{\sum_{p=1}^{n} e}{P(c)} \tag{5.1}$$

.

Através da média, é possível obter a função $F(c)$, utilizada na classificação de cada cenário c, descrita por

$$F(c) = \frac{M(c)}{S(c)} \tag{5.2}$$

sendo $M(c)$ a média de erro para cada um dos cenários e s a quantidade total de *socialbots* pertencentes ao cenário em questão. Desta forma, ao calcular os valores de $F(c)$ para cada um dos cenários, é possível classificá-los de acordo com a Tabela 5.5. Portanto, para classificar os seis cenários deste estudo, foi calculada $F(c)$ para cada um deles. O resultado da classificação deles pode ser observado na Tabela 5.6.

Um aspecto observado durante a classificação, foi que o cenário "Clonar um usuário" não foi classificado como "muito difícil", sendo que durante a criação dos cenários, havia uma certa expectativa de que esse seria classificado como o mais difícil de todos.

TAB. 5.5: Tabela representativa dos valores da função $F(c)$ associados aos níveis de dificuldade

Níveis de Dificuldade	Intervalo para $F(c)$
Muito difícil	Entre 0,81 e 1,0
Difícil	Entre 0,61 e 0,80
Médio	Entre 0,41 e 0,60
Fácil	Entre 0,21 e 0,40
Muito fácil	Entre 0,0 e 0,2

TAB. 5.6: Classificação dos cenários após os experimentos.

Cenário	Nível de dificuldade
Curtir	Muito fácil
Compartilhar	Muito fácil
Interação inicial com amigos	Fácil
Interação diversificada no Facebook	Difícil
Atuação como humano no Facebook	Muito difícil
Clonar um usuário	Médio

Acredita-se que isso tenha ocorrido em função da quantidade limitada de arquivos de entrada a serem utilizados. Houve esta limitação porque tais artefatos eram gerados através do experimento de Ferreira (2018), que consistia numa determinada quantidade de usuários do Facebook que deveria utilizar um aplicativo capaz de automatizar um ataque de engenharia social.

Após os testes realizados em Ferreira (2018), tais arquivos foram disponibilizados pelo autor mediante à autorização das pessoas que participaram do experimento. Entretanto, a quantidade deste tipo de arquivo foi limitada em apenas dois porque somente este número de usuários que participaram dos testes autorizaram o uso do seu arquivo neste estudo. Portanto, não houve diversidade de comportamento.

Por conta dos experimentos e categorização dos mesmos através dos diversos cenários apresentados servirem de instrumento de criação de um *dataset* descrito e classificado, considerou-se necessário descrever a estrutura do *dataset*. Ela será apresentada na subseção 5.2.1, de modo a facilitar o entendimento dos terceiros que queiram utilizar a base em relação ao seus atributos. Algumas informações, *e.g.* data de nascimento, que não estão presentes na base de dados, poderão ser enviadas. Esta informação é de fácil disponibilização porque foi desenvolvida a documentação dos processos realizados. Isso foi feito principalmente porque foram criados perfis para representarem menores de idade.

Contudo, como o Facebook não permite menores de 13 anos, foram atribuídas datas

de nascimentos diferentes daquelas estipuladas inicialmente durante a definição dos perfis dos *socialbots*. Portanto, a data do Facebook não é necessariamente condizente com o comportamento apresentado, *i.e.*, embora o perfil apresente maioridade, o seu comportamento pode ser de uma pessoa com idade inferior à 18 anos.

5.2.1 ESTRUTURA DO *DATASET* GERADO

O conteúdo gerado através das ações realizadas pelos *socialbots* durante o experimento foram armazenados na estrutura descrita nesta subseção. Conforme visto no capítulo 4, o Backendless foi escolhido para atender ao requisito de persistência de dados, armazenando dados referentes a mais de 700 ações de *socialbots*. Com a preocupação de disponibilizar uma base de dados consistente, um versionamento de base foi realizado, a fim de desconsiderar as atividades que eram realizadas apenas para validar as funcionalidades a serem realizadas pelo *socialbots* no Facebook e o correto armazenamento das mesmas.

Desta forma, foram criadas a *database* de teste (utilizada na etapa da validação da *social botnet* do experimento), e a de produção, sendo somente a última a ser disponibilizada. Para a escolha de qual era a base a ser usada, era enviado um parâmetro na url do Socialtree Web, sinalizando qual delas deve ser utilizada para armazenamento. Embora apresentem níveis de im portâncias diferentes, as bases dispunham de uma mesma estrutura de tabelas, ou seja, eram formadas da seguinte maneira:

- **Tabela Action:** é responsável por armazenar as informações referentes às ações efetuadas pelo controlador da *social botnet* utilizando os perfis dos *socialbots*. Entende-se por ação toda e qualquer interação que os *socialbots* podem realizar no Facebook. Tais atividades foram explicitadas na Seção 4.1 do capítulo 4. Desta forma, as informações armazenadas na tabela são:
  - *id:* URL identificadora da ação, normalmente é a própria URL fornecida pelo Facebook que remeterá ao conteúdo gerado pela ação. Tais URLs seguem o padrão "https://pagina_do_perfil/tipo_de_acao/codigo_numerico_da_acao_definido_pelo_facebook", *e.g.*, "https://www.facebook.com/fulana.silva.744/posts/1715835585153924";
  - *id_action_user:* uma URL que identifica unicamente na base do Facebook o usuário que efetuou a ação;
  - *action:* uma string que identifica a ação efetuada, podendo ter os valores,

"*post*", "*like*", "*comment*", "*reply*", "*addFriend*", "*removeFriend*", "*acceptFriendRequest*", "*rejectFriendRequest*";

- *action_scenario:* chave estrangeira para o cenário de teste associado, caso haja algum;
- *action_accepted_date:* armazena a hora e data em que um pedido de amizade foi aceito;
- *body_message:* Conteúdo da ação. Ex.: Texto de um post ou comentário feito.
- *body_attached_file:* URL de algum arquivo que tenha sido anexado na ação. Utilizado comumente para armazenar uma imagem postada.
- *reference_body_message:* Conteúdo interagido. Ex.: Texto de um post onde o usuário comentou ou curtiu.
- *reference_id:* URL identificadora do conteúdo interagido. Ex.: URL de um post onde o usuário gerenciado comentou ou curtiu.

- **Tabela Test_Scenario:** é a responsável por armazenar as informações referentes aos cenários de teste executados no sistema utilizando os *socilabots* gerenciados através do Socialtree Web. Como o Backendless já define um atributo *objectId* para todas as tabelas que recebe um valor único para cada entrada persistida, não houve a necessidade de criar um campo para esse fim. Desta forma, as informações armazenadas nesta tabela são:
  - *name:* nome do cenário de teste.
  - *description:* descrição do cenário de teste. Através desta informação, é possível detalhar melhor o cenário, caso o seu nome não seja suficientemente autoexplicativo para o usuário do *dataset*

- **Tabela Notify_Interaction**: armazena a interação de terceiros com perfis dos *socialbots* gerenciados na interface do Socialtree Web. As interações de terceiros são obtidas através da verificação a cada um minuto realizada pela aplicação *Socialtree Automation*, descrita no capítulo 4. As notificações podem ser resultantes de convite de páginas ou para eventos, respostas a comentários em posts textuais ou de fotos do feed do próprio perfil ou em comentários feitos em grupos, curtidas ou reações, menções e respostas a comentários onde foi feita uma menção. As informações armazenadas são:

- *objectId:* id gerado automaticamente pelo Backendless.
- *notification_time:* momento da ocorrência da interação.
- *url:* url do conteúdo onde ocorreu a interação.
- *notification_type:* Mapeia os tipos da notificação recebida, que podem ser:
  a) plan_user_invited: convite de evento;
  b) fbpage_fan_invite: convite para curtir uma página;
  c) group_comment_reply: resposta a comentário em algum grupo;
  d) group_comment: comentário em um grupo;
  e) photo_reply: resposta à post de foto;
  f) like: curtida;
  g) comment_mention: citação ao usuário;
  h) mentions_reply: resposta de comentários onde o usuário foi citado;
  i) feed_comment_reply: resposta de comentários do feed;
  j) feedback_reaction_generic: reação;
  k) friend_request_accepted: resposta positiva de solicitação de amizade feita através da ferramenta.
- author: Nome do autor da notificação;
- authors_ids: São os IDs (identificadores) dos autores diferentes do nome, cujo formato no banco é chave-valor, onde ambos os campos são iguais. Através desta informação, a digitar http://facebook.com/profile.php?id=valorouchave, é possível encontrar a página de um perfil.

Além de ser aplicado

5.3 COMPARAÇÃO ENTRE *DATASETS*

Para construir um *dataset* gerado a partir de ações da *social botnet* no Facebook, foram realizados os experimentos, descritos na seção 5.2. As ações dos experimentos foram armazenadas na estrutura explicitada na subseção 5.2.1.

A fim de analisar os aspectos positivos e negativos deste *dataset*, foi realizada uma comparação entre ele e os demais conjunto de dados com atuação na mesma OSN. De um certo modo, isso também auxilia na definição dos pesquisadores de qual base de dados será mais vantajosa em um determinado estudo, porque a comparação é embasada nos atributos que eles apresentam.

Para a realização da comparação, foram escolhidos alguns dos *datasets*, cujo conteúdo está associado ao Facebook e apresenta informações além do relacionamento representado exclusivamente através de grafos. Esta restrição foi estabelecida a fim de garantir que os *datasets* a serem comparados apresentariam atuação na mesma OSN deste estudo e alguma informação sobre ações ou preferências dos usuários. Assim, contrastou-se a base de dados gerada com Jin et al. (2013) e McAuley e Leskovec (2012). O primeiro utiliza um conjunto de dados com diversos atributos sintéticos de usuários do Facebook embasados em diversos *clusters* de usuários, em função do *dataset* original, cuja fonte não é indicada, conter apenas as redes de amigos. Já o segundo é constituído apenas de "círculos"(ou "listas de amigos") do Facebook (MCAULEY; LESKOVEC, 2012).

Além disso, também foram selecionadas as informações dos *socialbots* que podiam ser obtidas de forma direta, apenas através da observação dos dados, ou indireta, onde era preciso aplicar técnicas, como algum algoritmo ou mineração de dados, para manipular e extrair informações que até o momento não eram tão explícitas. Desta forma, é possível observar o resultado da comparação através da Tabela 5.7.

TAB. 5.7: Comparação entre os *datasets* a partir dos atributos dos mesmos em relação aos perfis do Facebook. No contexto do dataset gerado e classificado neste estudo, os perfis estão associados aos *socialbots*.

Atributos	*Dataset deste estudo*	(MCAULEY; LESKOVEC, 2012)	(JIN et al., 2013)
Conteúdo comentado	Sim	Não	Não
Conteúdo curtido	Sim	Não	Não
Conteúdo postado	Sim	Não	Não
Resposta de comentário	Sim	Não	Não
Lista de amigos	Sim	Sim	Não
Amizade solicitada por um perfil	Sim	Não	Não
Solicitação de amizade aceita ou recusada por um perfil	Sim	Não	Não
Identificação dos perfis (nome ou URL)	Sim	Sim	Não
Preferências dos perfis	Sim	Sim	Não
Conteúdo em chats	Não	Não	Não
Mapeamento das ações realizadas pelos perfis	Sim	Não	Não
Indicação da data e horário de uma ação realizada	Sim	Não	Não
Mapeamento das interações com os perfis realizadas por terceiros	Sim	Não	Não
Grau médio de relevância de um usuário em função dos amigos	Sim	Não	Sim

Com base na Tabela 5.7, é possível observar o resultado da comparação dos *datasets* e limitação na quantidade de *dataset* relacionado ao Facebook, em função do número de objetos de comparação ser pequeno. Através da base gerada neste estudo pode-se obter informações que não são extraídas dos demais objetos utilizados nesta comparação. Por exemplo, McAuley e Leskovec (2012) mostra os círculos de amizade, mas não é indicado o modo como que eles foram estabelecidos, ou seja, quem adicionou quem, mas isso é mapeado no *dataset* deste estudo.

Alguns dados, como as preferências dos perfis, embora não estejam inclusas dentro do *dataset* gerado neste estudo, podem ser fornecidos caso haja interesse do pesquisador a utilizar este *dataset*, pois a lógica de criação de cada *socialbot* foi documentada. Isso foi feito como um complemento da metodologia da descrição do *dataset*.

Além disso, o desenvolvimento de uma metolodogia para classificação do *dataset* e a implementação da mesma, não foi algo observado nas bases de dados analisadas, já que não havia necessidade de detalhamento em função da diversificação limitada dos atributos contidos nas mesmas. Portanto, observa-se que o *dataset* gerado resolve o problema de escassez de dados descritos discutido neste estudo.

5.4 CARACTERÍSTICAS DO FACEBOOK E DE OUTROS FORNECEDORES DE SERVIÇOS ANALISADAS DURANTE OS EXPERIMENTOS

Como o objetivo secundário deste estudo é a criação de uma *social botnet*, durante os experimentos, foi possível observar algumas políticas de segurança que visavam limitar ações suspeitas, sejam elas relacionadas aos provedores de email ou ao Facebook.

A criação do perfil ou validação no Facebook a partir número de celular, é algo que sofre uma forte verificação, pois a identificação de números celulares temporários na maioria dos casos ocorria imediatamente. Para um mesmo número considerado válido, era permitido o seu uso em até 3 vezes, lembrando que ele estava associado a perfis distintos. Normalmente diante de uma suspeita, o Facebook solicitava a inserção de um número. Neste caso, as opções existentes seriam colocar um número de celular válido ou criar um novo perfil seguindo a mesma metodologia. Após ultrapassar o limite de uso de um mesmo número, aparecia a mensagem

> *"There was an error verifying your contact information: The phone number you're trying to verify was recently used to verify a different account. Please try a different number."*

Outra validação também observada durante o período prévio à realização dos experimentos, o qual era também analisado como o Facebook se comportava, foi que nos casos onde os perfis são criados sem a inserção de qualquer informação pessoal e sem execução de ações, a conta é excluída em menos de uma semana ou, para utilizá-la, é necessário inserir um celular válido.

Entretanto, uma questão observada em relação à validação por celular, foi que nos cenários em que um mesmo número era utilizado em diferentes perfis (até 3), o Facebook sugeria os amigos associados àquelas contas de usuários, facilitando a inserção dos *socialbots* nesta mídia social. Mesmo com isso, este tipo de validação é algo que limita bastante a criação de perfis falsos, sendo esse um ponto positivo do Facebook.

Outro aspecto observado em função da abordagem inicial para criação da *social botnet* ter sido fundamentada com base na API GRAPH Facebook, foi a restrição executada por esta mídia social, que não existia inicialmente e foi inserida durante o período dos experimentos, para que um usuário pudesse utilizar um aplicativo em desenvolvimento. Como a GRAPH API é ainda utilizada somente para fazer o *login* dos perfis do *socialbots*, foi necessário encontrar um meio de contornar esta validação. Essa é feita através de número de celular, cujas limitações já foram indicadas, ou pelo cadastro de um método de pagamento que consiste no registro de um cartão de débito ou crédito. Durante este cadastro, o Facebook analisa a veracidade do cartão através da cobrança de US$1,00 e o extorno desta quantia é instantâneo.

Na validação através de um método de pagamento, o Facebook não exige a inserção do nome do titular do cartão, já que é solicitado apenas o número do cartão, código de segurança e data de validade. Desta forma, é possível inserir um cartão diversas vezes em perfis distintos para pagamento, sem aparecer qualquer erro no Facebook. É válido ressaltar que esta mídia social permitia o uso de cartões virtuais, providos por alguns fornecedores de cartão de crédito.

Mesmo sem erro no método de pagamento ao inserir um cartão diversas vezes, o seu uso não é efetivo para a liberação do acesso a um aplicativo em desenvolvimento pelo perfil falso. Um ponto de atenção é que ao inserir o cartão nos 3 perfis e depois removê-lo, não faz com que a quantidade de vezes que o mesmo pode ser utilizado ultrapasse a quantidade de 3 usos no total. Isso de certa forma indica que mesmo o usuário removendo determinadas informações, a mesma não é excluída realmente do Facebook. O mesmo foi observado com a inserção e exclusão de um mesmo número de celular.

Em relação ao uso de uma mesma máquina com um IP fixo para o processo de automatização na criação das contas do Facebook, não foi observada nenhuma restrição

neste contexto pelo Facebook, já que foi permitido criar mais de vinte contas em um único dia. Normalmente, a limitação existia por parte dos provedores de emails clássicos, como o Hotmail, Gmail ou Yahoo. O primeiro restringia a criação de cinco emails por dia por números de celulares. O segundo permitia por dia a geração de duas contas com um mesmo número de celular.

Já o terceiro, no primeiro dia de uso permitiu a criação de 10 contas com o mesmo celular. Posteriormente, surgiu um bloqueio que durou dois dias e, em seguida, foi permitido criar cinco contas de email por dia sem o bloqueio aparecer novamente. Mesmo com o apontamento destas questões relacionadas aos provedores de emails, não foi possível indicar se o não bloqueio por IP se manteria se tivesse sido usada uma quantidade maior de números.

Uma questão importante observada foi que após os perfis do Facebook serem criados, confirmados e terem um período curto de uso, dificilmente existia alguma validação posterior em relaçâo às ações que executavam. Isso foi observado sobretudo na etapa do experimento referente à validação da *social botnet*. Já que houve casos onde foram adicionadas 101 pessoas em 1 perfil em apenas dia, sendo que 83 pessoas foram aceitas por um *socialbot* em 4 minutos, e não existiu qualquer tipo de bloqueio pelo Facebook. Outro exemplo ocorreu durante a inserção de imagens de repositórios públicos, já que o Facebook não os identificou como falsos.

Desta forma, observa-se que a parte crítica para a geração da *social botnet* é constituída das etapas que consistem na criação das contas no Facebook e a respectiva validação das mesmas por meio de informações onde é restrita a quantidade de vezes em que são utilizadas. Entende-se que email, número de celular e dados de cartões de crédito representam estas informações.

Isso foi observado principalmente porque diferentes soluções para a criação e validação das contas tiveram que ser abordadas, como a geração de cartões virtuais e utilização de ferramentas de emails e celulares temporários. Entretanto, o mesmo não aconteceu com os dados que também poderiam ser mapeados pelo Facebook. Por exemplo, em uma amostra de 50 perfis, em somente 6 destes perfis o Facebook solicitou foto logo após ou durante a criação da conta. Após a inserção da foto, o Facebook emite a mensagem

> "Você não pode entrar no momento. Entraremos em contato com você em breve depois de analisarmos sua foto. Você será desconectado do Facebook como uma medida de segurança."

Após esta validação, os perfis podem ou não ser liberados para uso. No contexto deste

estudo, os perfis não foram liberados.

6 CONSIDERAÇÕES FINAIS

O Facebook é a mídia social mais utilizada no mundo (STATISTA, 2018) e isso associado à possibilibidade de utilizar os *socialbots* presentes na mesma para fins maliciosos, evidencia a necessidade de detectar comportamentos anômalos. Através da detecção é possível propor contramedidas que inibam a atuação de uma *social botnet* maliciosa.

Para identificar tais comportamentos, muitas das abordagem vistas na área de Segurança da Informação exigem dados completos e seus relacionamentos. Entretanto, através dos trabalhos relacionados, observou-se a existência do problema de escassez de dados disponíveis com estas características.

Em função disso, um dos objetivos propostos deste estudo foi criação de uma arquitetura conceitual, cujas as fases resultam na geração de uma base de dados: com identificação de quais eram os *socialbots*; com indicação da semântica dos cenários compostos por ações; formadas pelas atividades executadas e conteúdos publicados por estes robôs sociais.

Ao cumprir com este objetivo, é possível disponibilizar um conjunto de dados mediante apresentação de motivo de uso e solicitação formal para o IME e para a autora do estudo, para treinamento de algoritmos de detecção de *social botnet*. Através deste *dataset*, é possível definir um denominador comum para comparações entre os algoritmos, ou seja, as pesquisas que atuam no mesmo contexto deterão um meio de comparar os resultados e de garantir a imparcialidade dos estudos apresentados.

Além disso, este *dataset* pode ser usado em diferentes viés para a inteligência artificial como, por exemplo, a exploração das interações humanas com os *socialbots*. Desta maneira, seria possível propiciar respostas inteligentes e humanizadas com base na análise das interações dos usuários legítimos.

Como foi necessária a criação de uma arquitetura para criar esta base de dados, o fato dela ter sido também apresentada de modo conceitual contribuiu principalmente para manutenção da base ao longo dos anos por outros pesquisadores, pois a arquitetura proposta não está limitada às tecnologias.

Outra contribuição alcançada foi obtida através da criação de uma metodologia para criação de *socialbots*, pois através da sua manipulação e análise da efetividade da mesma foi possível avaliar o *Facebook Immune System* e indicar aspectos que as pesquisas relacionadas à detecção de *social botnet* devem observar. De um certo modo, ao serem aplicadas

diversas abordagens para as etapas de criação e validação dos perfis, foi possível observar: as deficiências de segurança no Facebook; as situações em que a validação realizada por esta mídia é efetiva; e a necessidade de alguns dos serviços que são utilizados para criar os perfis serem mais restritivos. Por exemplo, o Yahoo realiza o bloqueio inicialmente após a criação de 10 perfis, mas após dois dias, não foi observado o bloqueio.

Em função dos aspectos supracitados, mesmo sendo observada várias tentativas de validar ou de realizar bloqueio, esta dissertação evidenciou que a dificuldade do Facebook em mapear os *socialbots* ainda existe, mesmo isso já sendo indicado em Kumar e Reddy (2012). Além disso, este estudo indica que o foco do problema está após as fases de criação e validação dos perfis utilizados pelos robôs sociais. Isso ratifica o fato de que a área de detecção de *social botnet* tem aspectos a serem aprimorados, principalmente em relação ao Facebook, onde os estudos são escassos devido ao alto custo de se trabalhar com ele em relação às outras mídias sociais, como Twitter (DOUGLAS, 2017).

Desta forma, sugere-se então o uso de soluções, compostas de avaliação social e técnica, para proteger com maior eficácia as OSNs, já que há possibilidade delas sofrerem explorações técnicas (por exemplo, exploradores maliciosos das vulnerabilidades de uma OSN) ou humanas (engenharia social, entre outras).

Do lado dos usuários do OSN, observou-se que eles não são suficientemente cuidadosos ao aceitarem pedidos de amizade, especialmente quando compartilham amigos comuns com o remetente. Desta forma, de um modo, esta análise contribuiu para gerar também um alerta para os usuários desta OSN.

6.1 DIFICULDADES ENCONTRADAS

Durante a pesquisa, teve-se dificuldade em atuar sobre o Facebook em função das restrições não previstas das atividades em sua GRAPH API. Isso porque, durante o levantamento de qual mídia social seria utilizada, foram verificadas quais ações através delas eram permitidas e as datas de atualização, a fim de minimizar o impacto nesta pesquisa.

Inicialmente, o Facebook menciona que cada versão permanecerá por pelo menos 2 anos a partir do lançamento, para que seja fornecida uma linha de tempo sólida de funcionamento dos aplicativos (FACEBOOK, 2016), mas não é explicitado quais atividades se tornarão obsoletas.

Durante a etapa de implementação não foi passível de execução o comentário em páginas diferentes das do dono do perfil através da GRAPH API e, também foi averiguado o encerramento dos prazos da *realise* 2.3, no qual não era mapeável as atividades que não

seriam mais fornecidas.

Entretanto, ao perceber que as atualizações da GRAPH API limitavam cada vez mais as atividades, a atuação da mesma precisou ser reduzida e a maior parte dos requisitos passou a ser atentido com uma solução que envolvesse a página web do Facebook.

Como a página pode sofrer ajustes sem avisos prévios e a atualização da mesma para os usuários não era simultânea, ou seja, alguns utilizavam uma versão antiga, outros visualizam as novas, foi necessário executar durante todo o perído da realização deste estudo a manutenção da ferramenta utilizada para gerenciamento dos *socialbots*.

Outra questão que foi restritiva deste trabalho foi a limitação de *hardware* disponível, o que impacta principalmente na exploração simultânea dos *socialbots* e das ações que os mesmos podem executar. Normalmente, *botnets* maliciosas não sofrem com esta limitação em função de utilizar o hardware e rede dos usuários que infectam para executar as atividades dos *bots* (FENG et al., 2011).

Um aspecto que tende a dificultar ainda mais as pesquisas nesta OSN é a criação ou evolução dos mecanismos de defesa que está sendo feito pelo Facebook (NEWS, 2018), devido à colheta indevida de informações de até 87 milhões de usuários realizada pela Cambridge Analytica (TIMES, 2018).

6.2 TRABALHOS FUTUROS

Como trabalhos futuros, propõe-se principalmente a manutenção evolutiva do *dataset*. Isso poderá ser feito de diferentes formas, seja através da construção de novos experimentos por meio da definição de novos cenários, ou pela definição de uma arquitetura conceitual capaz de aplicada na geração de uma base de dados independente da mídia social, ou através de experimentos com maior durabilidade.

Em relação às interações do *socialbot*, outro ponto de exploração é a investigação da relação entre os diferentes perfis de usuários legítimos que interagiram com os *socialbots*, a fim de buscar padrões em seus comportamentos. Desta forma, seria possível recomendar boas práticas para estes usuários.

Ainda no tópico de interações, considera-se como um ponto de aprimoramento a inclusão de inteligência artificial (IA) nos casos onde as ações dos *socialbots* são feitas de modo automático, já que a automatização realizada neste estudo é feita através dos parâmetros passados através de um arquivo de entrada. Atráves do uso de IA, é possível que a ferramenta seja utilizada para diferentes fins como, por exemplo, para propagar conhecimento educional.

Ao integrar a ferramenta desenvolvida com o *chatbot*, que é um software para realizar uma comunicação informal entre um humano e um computador (SHAWAR; ATWELL, 2007), permitiria uma conversa entre os *socialbots* com os usuários legítimos em tempo real, aprimorando ainda mais o comportamento destes robôs sociais.

Outro aspecto a ser evoluído é a implementação de uma arquitetura conceitual, capaz de atuar na geração de *dataset*, independente da mídia social. Assim, será possível fazer uma análise de como um *social botnet* é capaz de se infiltrar em qualquer OSN.

Existe também a possibilidade de analisar o impacto de uma *social botnet* em diferentes aspectos, tais como: sua influência na economia de um setor, por exemplo, quando é utilizada para *marketing* de um produto ou empresa; avaliação do seu impacto nos relacionamentos sociais, seja por meio de grafos ou outras técnicas.

6.3 DISCUSSÃO ÉTICA

Dada a natureza de uma SbN, ao definir o escopo da pesquisa foi iniciada uma discussão ética, a fim de avaliar se é éticamente aceitável e justificável a criação de *social botnet* e *socialbots*. A metodologia da pesquisa e sua execução foram definidos de modo que os experimentos não incluíssem ações maliciosas para criação dos *socialbots*, como uso de *malware* ou de informações vendidas no mercado ilícito, e para a execução de suas respectivas ações. Os identificadores dos usuários foram definidos por URL justamente pela possibilidade de serem adquiridos através de *crawlers* que varrem os perfis no Facebook, como foi visto em Catanese et al. (2011).

Assim, procurou-se reduzir ao mínimo possível os riscos para um usuário legítimo e ainda assim, buscou-se manter as ações sendo realistas, visando mapear de modo confiável a viabilidade de um ataque num cenário real, de modo que isso tudo refletisse em um *dataset* para os demais estudos nesta área.

Portanto, esta pesquisa de um modo geral contribui para que a comunidade obtenha uma visão genuína em relação aos meios de exploração de uma OSN. Facilitando no mapeamento de como evoluir as técnicas de defesa, naturalmente após a etapa de detecção, em relação aos comportamentos anômalos realizados através do Facebook.

Como parte de um código de ética visto em Boshmaf et al. (2013), os detalhes relevantes do experimento que podem impactar diretamente ao Facebook, ao ser explorado por criminosos, será enviado diretamente para esta mídia social, através do acesso ao site *www.facebook.com/security*.

7 REFERÊNCIAS BIBLIOGRÁFICAS

ABT, S.; BAIER, H. Are we missing labels? a study of the availability of ground-truth in network security research. In: 2014 THIRD INTERNATIONAL WORKSHOP ON BUILDING ANALYSIS DATASETS AND GATHERING EXPERIENCE RETURNS FOR SECURITY (BADGERS), 3., 2014. **Anais**... [S.l.: s.n.], 2014, p. 40–55. Acesso em: 13 jul. de 2016.

AGRAWAL, M.; VELUSAMY, R. L. R-salsa: A spam filtering technique for social networking sites. In: 2016 IEEE STUDENTS' CONFERENCE ON ELECTRICAL, ELECTRONICS AND COMPUTER SCIENCE (SCEECS), s.n., 2016. **Anais**... [S.l.: s.n.], 2016, p. 1–7.

AL-DAYIL, R. A.; DAHSHAN, M. H. Detecting social media mobile botnets using user activity correlation and artificial immune system. In: 2016 7TH INTERNATIONAL CONFERENCE ON INFORMATION AND COMMUNICATION SYSTEMS (ICICS), 7., 2016. **Anais**... [S.l.: s.n.], 2016, p. 109–114.

BANKS ALEX. 2015 Brazil Digital Future in Focus. Disponível em: <https://www.comscore.com/por/Insights/Apresentacoes-e-documentos/2015/2015-Brazil-Digital-Future-in-Focus>. Acesso em: 18 mai. de 2015.

AMAZON. The top 500 sites on the web. Disponível em: <http://www.alexa.com/topsites>. Acesso em: 24 jan. de 2017.

FUNDAÇÃO SOFTWARE APACHE. Apache HTTP Server Project. Disponível em: <http://httpd.apache.org/ABOUT_APACHE.html>. Acesso em: 07 de jun. de 2018.

BAI, X.; TSAI, W. T.; PAUL, R.; FENG, K. ; YU, L. Scenario-based modeling and its applications. In: PROCEEDINGS OF THE SEVENTH IEEE INTERNATIONAL WORKSHOP ON OBJECT-ORIENTED REAL-TIME DEPENDABLE SYSTEMS. (WORDS 2002), 7., 2002. **Anais**... [S.l.: s.n.], 2002, p. 253–260.

BOLLEN, J.; MAO, H. ; ZENG, X. Twitter mood predicts the stock market. **Journal of Computational Science**, v. 2, n. 1, p. 1 – 8, 2011. Disponível em: <http://www.sciencedirect.com/science/article/pii/S187775031100007X>. Acesso em: mar. de 2011.

BOSHMAF, Y.; MUSLUKHOV, I.; BEZNOSOV, K. ; RIPEANU, M. Design and analysis of a social botnet. **Comput. Netw.**, v. 57, n. 2, p. 556–578, 2013.

BOSHMAH, Y.; MUSLUKVO, I.; BEZNOSOV, K. ; RIPEANU, M. The socialbot network: When bots socialize for fame and money. **ASAC 11 Proceedings of the 27th Annual Computer Security Applications Conference**, v. 8, n. 2, p. 93–102, 2011.

FACEBOOK BUSINESS. Informações do público. Disponível em: <https://www.facebook.com/ads/audience-insights/interests?act=149803375090494&age=18-&country=BR>. Acesso em: 30 nov. de 2017.

CAIDA. Data Collection, Curation and Sharing. Disponível em: <https://www.caida.org/data/>. Acesso em: 21 jul. de 2018.

CATANESE, S. A.; DE MEO, P.; FERRARA, E.; FIUMARA, G. ; PROVETTI, A. Crawling facebook for social network analysis purposes. In: PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON WEB INTELLIGENCE, MINING AND SEMANTICS, s.n., 2011. **Anais**... [S.l.: s.n.], 2011, p. 52.

RORY CELLAN-JONES. Facebook has more than 83 million illegitimate accounts. Disponível em: <http://www.bbc.com/news/technology-19093078>. Acesso em: 8 mai. de 2016.

CHEN, K.; CHEN, L.; ZHU, P. ; XIONG, Y. Unveil the spams in weibo. In: 2013 IEEE INTERNATIONAL CONFERENCE ON GREEN COMPUTING AND COMMUNICATIONS AND IEEE INTERNET OF THINGS AND IEEE CYBER, PHYSICAL AND SOCIAL COMPUTING, s.n., 2013. **Anais**... [S.l.: s.n.], 2013, p. 916–922.

COMPAGNO, A.; CONTI, M.; LAIN, D.; LOVISOTTO, G. ; MANCINI, L. V. Boten elisa: A novel approach for botnet c&c in online social networks. In: 2015 IEEE CONFERENCE ON COMMUNICATIONS AND NETWORK SECURITY (CNS), s.n., 2015. **Anais**... [S.l.: s.n.], 2015, p. 74–82.

FGV DAPP. Robôs, redes sociais e política: estudo da FGV/DAPP aponta interferências ilegítimas no debate público na Web. Disponível em: <http://dapp.fgv.br/robos-redes-sociais-e-politica-estudo-da-fgvdapp-aponta-interferencias-ilegitimas-no-debate-publico-na-web/>. Acesso em: 27 set. de 2017.

FGV DAPP. Robôs, Redes Sociais e Política no Brasil. Disponível em: <http://dapp.fgv.br/wp-content/uploads/2017/08/Robos-redes-sociais-politica-fgv-dapp.pdf>. Acesso em: 20 ago. de 2017.

DARPA. Darpa Intrusion Detection Scenario Specific Datasets. Disponível em: <http://www.ll.mit.edu/mission/communications/cyber/CSTcorpora/ideval/data/>. Acesso em: 21 jul. de 2018.

DATASUS. Nascidos Vivos - Brasil. Disponível em: <http://tabnet.datasus.gov.br/cgi/deftohtm.exe?sinasc%2Fcnv%2Fnvuf.def>. Acesso em: 30 nov. de 2017.

DAVIS, C. A.; VAROL, O.; FERRARA, E.; FLAMMINI, A. ; MENCZER, F. Botornot: A system to evaluate social bots. In: PROCEEDINGS OF THE 25TH INTERNATIONAL CONFERENCE COMPANION ON WORLD WIDE WEB, 25., 2016. **Anais**... [S.l.: s.n.], 2016, p. 273–274.

COSTA GEANDRESON; ALVARES DOUGLAS. Extração de dados em redes sociais usando Python. Disponível em:

<http://www.linc.ufpa.br/fabricasistemas/cursoextracao/materiais/3.%20Twitter%20e%20Facebook%20API.pdf>. Acesso em: 21 set. de 2017.

EXAME. As 200 cidades mais populosas do Brasil. Disponível em: <https://exame.abril.com.br/brasil/as-200-cidades-mais-populosas-do-brasil/>. Acesso em: 5 jul. de 2017.

GOVEIA FABIO. Conversas citando Aécio no twitter. Disponível em: <http://www.labic.net/blog/internet-2/bots-contra-a-sociedade/>. Acesso em: 15 mai. de 2014.

FACEBOOK. Platform Versioning. Disponível em: <https://developers.facebook.com/docs/apps/versions>. Acesso em: 1 mai. de 2016.

FENG, X. X.; PENG, Y. ; ZHAO, Y. L. The analysis of botnet based on http protocol. In: MATERIALS SCIENCE AND ENGINEERING, 1., 2011. **Anais**... [S.l.]: Trans Tech Publications, 2011, p. 575–579. Acesso em: 13 jul. de 2016.

FERRARA, E.; VAROL, O.; DAVIS, C. A.; MENCZER, F. ; FLAMMINI, A. The rise of social bots. **CoRR**, v. abs/1407.5225, p. 96–104, 2014.

FERREIRA, M. L. **Metodologia para execução de engenaria social automatizada**. 2018. 60 f. Dissertação (Mestrado em Sistemas e Computação) – Instituto Militar de Engenharia, Rio de Janeiro, 2018. Disponível em: <s.n>. Acesso em: 03 mai. de 2018.

FIELDING, R. T. **Architectural Styles and the Design of Network-based Software Architectures**. 2000. 162 f. Tese (Doutorado em Ciência da Computanção e Informação) – Universidade de Califórnia, Irvine, 2000.

GARCIA-TEODORO, P.; DIAZ-VERDEJO, J.; MACIÁ-FERNÁNDEZ, G. ; VÁZQUEZ, E. Anomaly-based network intrusion detection: Techniques, systems and challenges. **computers & security**, v. 28, n. 1, p. 18–28, 2009.

GHANADI, M.; ABADI, M. Socialclymene: A negative reputation system for covert botnet detection in social networks. In: TELECOMMUNICATIONS (IST), 2014 7TH INTERNATIONAL SYMPOSIUM ON, 7., 2014. **Anais**... [S.l.: s.n.], 2014, p. 954–960.

GLOBO. Sequestrador diz ter planejado crime com informações de rede social. Disponível em: <http://g1.globo.com/sc/santa-catarina/noticia/2014/06/sequestrador-diz-ter-planejado-crime-com-informacoes-de-rede-social.html>. Acesso em: 4 jun. de 2014.

HE, Y.; ZHANG, G.; WU, J. ; QIANGL. Understanding a prospective approach to designing malicious social bots. **Security and Communication Networks**, v. 2, p. 1–18, 2015.

HINGSTON, P. A new design for a turing test for bots. In: PROCEEDINGS OF THE 2010 IEEE CONFERENCE ON COMPUTATIONAL INTELLIGENCE AND GAMES, s.n., 2010. **Anais**... [S.l.: s.n.], 2010, p. 345–350.

IBGE. Censo Demográfico 2010. Disponível em: <https://censo2010.ibge.gov.br/nomes/#/ranking>. Acesso em: 2 jan. de 2018.

FACEBOOK IQ. Informações do público. Disponível em: <https://scontent-gig2-1.xx.fbcdn.net/v/t39.2365-6/10000000_118565898944995_257717601336033280_n.pdf?_nc_cat=0&oh=75b719869564f22a711c08c25a9d0f09&oe=5B42EFE1>. Acesso em: 10 fev. de 2018.

CHIEN ERIC; SHEARER JARRAD. W32.Koobface. Disponível em: <https://www.symantec.com/security_response/writeup.jsp?docid=2008-080315-0217-99&tabid=2>. Acesso em: 3 jan. de 2018.

JI, Y.; HE, Y.; JIANG, X. ; LI, Q. Towards social botnet behavior detecting in the end host. In: 2014 20TH IEEE INTERNATIONAL CONFERENCE ON PARALLEL AND DISTRIBUTED SYSTEMS (ICPADS), 20., 2014. **Anais**... [S.l.: s.n.], 2014, p. 320–327.

JI, Y.; HE, Y.; JIANG, X.; CAO, J. ; LI, Q. Combating the evasion mechanisms of social bots. **Computers & Security**, v. 58, p. 230–249, 2016.

JIN, L.; JOSHI, J. B. D. ; ANWAR, M. Mutual-friend based attacks in social network systems. **Comput. Secur.**, v. 37, p. 15–30, 2013. Disponível em: <http://dx.doi.org/10.1016/j.cose.2013.04.003>. Acesso em: 8 de ago. de 2016.

KIMURAI, H.; BASSO, L. F. C. ; MARTIN, D. M. L. Redes sociais e o marketing de inovações. **RAM. Revista de Administração Mackenzie**, v. 9, p. 157 – 181, 2008. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1678-69712008000100008>. Acesso em: 29 jul de 2016.

GOGA OANA; VENKATADRI GIRIDHARI;GUMMADI P. KRISHNA. The Doppelgänger Bot Attack: Exploring Identity Impersonation in Online Social Networks. Disponível em: <https://people.mpi-sws.org/ gummadi/papers/impersonators_IMC2015.pdf>. Acesso em: 23 jan. de 2018.

KUMAR, N.; REDDY, R. N. **Automatic detection of fake profiles in online social networks**. 2012. 31 f. Trabalho de Conclusão de Curso (Bachelor of Technology in Computer Science and Engineering) – National Institute of Technology Rourkela, Orissa, Índia, 2012.

BARRACUDA LABS. Attackers Use Fake Friends to Blend into Facebook. Disponível em: <https://blog.barracuda.com/2012/02/02/attackers-use-fake-friends-to-blend-into-facebook/>. Acesso em: 05 jan. de 2018.

LEE, K.; EOFF, B. D. ; CAVERLEE, J. Seven months with the devils: A long-term study of content polluters on twitter.. In: FIFTH INTERNATIONAL AAAI CONFERENCE ON WEBLOGS AND SOCIAL MEDIA, 15., 2011. **Anais**... [S.l.: s.n.], 2011, p. 1–8.

MCAULEY, J.; LESKOVEC, J. Learning to discover social circles in ego networks. In: PROCEEDINGS OF THE 25TH INTERNATIONAL CONFERENCE ON NEURAL INFORMATION PROCESSING SYSTEMS - VOLUME 1, 25., 2012. **Anais**... USA: Curran Associates Inc., 2012, p. 539–547.

MOME. MOME Database. Disponível em: <https://www.ist-mome.org/database/index.html>. Acesso em: 22 jul. de 2018.

FACEBOOK NEWS. *Cracking Down on Platform Abuse*. Disponível em: <https://newsroom.fb.com/news/2018/03/cracking-down-on-platform-abuse/>. Acesso em: 21 mar. de 2018.

MIGLIACCI PAULO. Sites vendem amigos, seguidores e "curtidas"nas redes sociais. Disponível em: <http://www1.folha.uol.com.br/tec/2014/05/1452055-sites-vendem-amigos-seguidores-e-curtidas-nas-redes-sociais.shtml>. Acesso em: 28 mai. de 2016.

THE WASHINGTON POSTER. Obama Raised Half a Billion Online. Disponível em: <http://voices.washingtonpost.com/44/2008/11/obama-raised-half-a-billion-on.html>. Acesso em: 3 mai. de 2016.

PREDICT. PREDICT. Disponível em: <https://www.predict.org/>. Acesso em: 22 jul. de 2018.

PROCOB. Censo Demográfico 2010. Disponível em: <https://www.procob.com/os-sobrenomes-mais-comuns-do-brasil/>. Acesso em: 23 dez. de 2017.

SELENIUM PROJECT. SeleniumHQ. Disponível em: <https://www.seleniumhq.org/docs/>. Acesso em: 4 jun. de 2016.

CARDOSO RAFAEL. Selenium – WebDriver. Disponível em: <https://tutorialselenium.wordpress.com/2015/02/09/selenium-webdriver-configuracao-eclipse/>. Acesso em: 4 mai. de 2016.

FACEBOOK REPORTS. Facebook Reports Fourth Quarter and Full Year 2017 Results. Disponível em: <https://investor.fb.com/investor-news/press-release-details/2018/Facebook-Reports-Fourth-Quarter-and-Full-Year-2017-Results/default.aspx>. Acesso em: 04 fev. de 2018.

SHAWAR, B. A.; ATWELL, E. Chatbots: are they really useful?. In: LDV FORUM, s.n., 2007. **Anais**... [S.l.: s.n.], 2007, p. 29–49.

FELIZARDO SILVIA. Internautas se mobilizam nas redes sociais para ajudar vítimas das enchentes. Disponível em: <http://g1.globo.com/mundo/noticia/2011/01/internautas-se-mobilizam-nas-redes-sociais-para-ajudar-vitimas-das-enchentes.html>. Acesso em: 3 mai. de 2016.

STATISTA. Facebook - Statistics & Facts. Disponível em: <https://www.statista.com/topics/751/facebook/>. Acesso em: 03 jan. de 2018.

TANNER, B. K.; WARNER, G.; STERN, H. ; OLECHOWSKI, S. Koobface: The evolution of the social botnet. In: 2010 ECRIME RESEARCHERS SUMMIT, s.n., 2010. **Anais**... [S.l.: s.n.], 2010, p. 1–10.

THOMAS, K.; NICOL, D. M. The koobface botnet and the rise of social malware. In: 2010 5TH INTERNATIONAL CONFERENCE ON MALICIOUS AND UNWANTED SOFTWARE, 5., 2010. **Anais**... [S.l.: s.n.], 2010, p. 63–70.

NEW YORK TIMES. *After Cambridge Analytica, Privacy Experts Get to Say 'I Told You So'*. Disponível em: <https://www.nytimes.com/2018/04/12/technology/privacy-researchers-facebook.html>. Acesso em: 12 abr. de 2018.

TIWARI, V. Analysis and detection of fake profile over social network. In: 2017 INTERNATIONAL CONFERENCE ON COMPUTING, COMMUNICATION AND AUTOMATION (ICCCA), s.n., 2017. **Anais**... [S.l.: s.n.], 2017, p. 175–179.

TURING, A. M. **Computing machinery and intelligence**. [S.l.]: Springer, 2009. 23–65 p.

TYAGI, A. K.; AGHILA, G. Detection of fast flux network based social bot using analysis based techniques. In: 2012 INTERNATIONAL CONFERENCE ON DATA SCIENCE ENGINEERING (ICDSE), s.n., 2012. **Anais**... [S.l.: s.n.], 2012, p. 23–26.

WATTERS, P. A.; HERPS, A.; LAYTON, R. ; MCCOMBIE, S. Icann or icant: Is whois an enabler of cybercrime?. In: 2013 FOURTH CYBERCRIME AND TRUSTWORTHY COMPUTING WORKSHOP, 4., 2013. **Anais**... [S.l.: s.n.], 2013, p. 44–49.

WHITMAN, M. E.; MATTORD, H. J. **Principles of information security**. [S.l.]: Cengage Learning, 2011.

WITS. WITS: Waikato Internet Traffic Storage. Disponível em: <https://wand.net.nz/wits/>. Acesso em: 23 jul. de 2018.

ZHANG, J.; ZHANG, R.; ZHANG, Y. ; YAN, G. The rise of social botnets: Attacks and countermeasures. **IEEE Transactions on Dependable and Secure Computing**, v. PP, n. 99, p. 1–1, 2017.