Roubo de Modelo para Ataque Adversarial em Sistemas de Detecção de Intrusão

Rafael Gomes Moreira[1], **Rafael Oliveira da Rocha**[1], **Leonardo Gonçalves Chahud**[1,2]**e Lourenço Alves Pereira Junior**[1]

[1]Instituto Tecnológico de Aeronáutica – São Jose dos Campos, SP – Brasil

[2]Universidade de São Paulo – São Carlos, SP – Brasil

{moreirargm,rafaelror,ljr}@ita.br, lchahud02@usp.br

Abstract. *Machine learning-based network intrusion detection systems can be vulnerable to adversarial attacks. However, executing these attacks requires knowledge of the internal information of the model in use, which may not be available to the attacker. This paper introduces a model stealing method focused on the equivalence of feature contribution between the target and substitute models and a black-box approach of the EBFA technique, named EBFA_BB. Compared to the attacks used as a baseline, the proposed attack was able to create substitute models with at least 10% more equivalence in the most significant features of the target model.*

Resumo. *Sistemas de detecção de intrusão em rede baseados em aprendizado de máquina podem ser vulneráveis a ataques adversariais. Porém, a realização desses ataques demanda conhecimento de informações internas do modelo utilizado, que podem estar indisponíveis para o atacante. Este trabalho apresenta um método de roubo de modelo com o foco na equivalência da contribuição das características entre modelo alvo e substituto, bem como uma abordagem caixa-preta da técnica EBFA, chamada EBFA_BB. Em comparação com os ataques utilizados como baseline, o ataque proposto conseguiu criar modelos substitutos com, no mínimo, 10% a mais de equivalência das características mais importantes do modelo alvo.*

1. Introdução

Observamos um número cada vez maior de dispositivos conectados em redes, principalmente pela disseminação da IoT, e, conforme estudo apresentado pelo Equipe TGT Consultant [Consult 2022], até 2025 haverá cerca de 27,1 bilhões de dispositivos IoT conectados. Assim, como efeito colateral adverso desse grande aumento de conectividade, a segurança passa a ser um fator de preocupação, tanto pelo fato do maior volume de informações transitando pelas redes tornarem mais difícil a detecção de ações maliciosas, como pelo fato de dispositivos IoT tornarem-se elementos de acesso para infraestruturas empresariais [Alshaikhli et al. 2022] protegidas ou serem utilizados como elementos amplificadores de ataques.

Com o volume de informação cada vez maior circulando na Internet, pesquisadores de segurança buscaram meios de caracterizar diversos ataques e, beneficiando-se das técnicas de aprendizado de máquina, treinar modelos que

fossem capazes de detectar a ocorrência desses ataques [Sharafaldin et al. 2018] e [Neto et al. 2023]. Por outro lado, os algoritmos de aprendizado de máquina podem ser vulneráveis a um tipo específico de ataque, chamado de Ataque Adversarial ou *Adversarial Machine Learning* [Huang et al. 2011], no qual algumas características do artefato avaliado pelo algoritmo de aprendizado de máquina são modificadas com objetivo de alterar o valor de inferência do modelo, ou ainda, alterar o seu comportamento para uma classe de objetos [Biggio et al. 2013]. Portanto, este tipo de ataque pode ser utilizado para modificar pontualmente algum aspecto do fluxo de rede malicioso, com o objetivo de evitar a detecção dos ataques pelo sistema de detecção de intrusão.

A realização de ataques adversariais contra sistemas de detecção de intrusão em rede baseados em aprendizado de máquina pode ocorrer em dois contextos: *Black-box*, no qual desconhecemos o funcionamento interno do modelo de classificação; e *White-box*, no qual a estrutura interna utilizada na criação do modelo é conhecida, tal como a técnica de aprendizado de máquina, a representação das características extraídas do fluxo de rede e como ela foi utilizada no treinamento do modelo etc. A despeito disso, grande parte das ferramentas baseadas em aprendizado de máquina para classificação de vulnerabilidades de rede utilizam-se do contexto *black-box* como uma forma de garantir certo nível de segurança dos dados ou da técnica utilizada. Porém, com o objetivo de serem adquiridas tais informações indisponíveis, são utilizadas técnicas de roubo de modelo [Oliynyk et al. 2023]. Tais técnicas buscam criar modelos substitutos, ou *surrogate models*, que são treinados com o objetivo de replicarem as características do alvo e permitirem a elaboração e o teste dos ataques adversarias ainda em ambiente controlado [Papernot et al. 2017].

Embora trabalhos anteriores tenham apresentado bons resultados em domínios como detecção de imagens, até onde é de nosso conhecimento, para o contexto de classificadores de rede, não foram encontrados trabalhos sobre roubo de modelo com foco na equivalência das características entre alvo e substituto, objetivando a realização de um ataque adversarial. Ademais, classificadores empregados em sistemas de detecção de intrusão (NIDS) apresentam desafios maiores, observadas sua variância temporal. Assim, destacam-se as variações na contribuição de características (*features*) e seus distintos níveis de contribuição no resultado final da função de classificação em função do contexto de rede da aplicação (*workload*) [Domingues et al. 2022]; e também uma quantidade menor de pré-processamento quando comparado ao domínio das imagens, considerando que os principais indicadores de comprometimento vêm de métricas de cabeçalhos e fluxos.

Assim, o problema endereçado neste artigo consiste em **Avaliar se técnicas de roubo de modelo focadas na criação de modelos substitutos com equivalência da contribuição das características viabilizam ataques adversariais contra sistemas de detecção de intrusão em rede**. Para resolver o problema da pesquisa, as seguintes Questões de Pesquisa foram definidas:

- **QP1:** A comparação entre a acurácia do modelo alvo e do modelo substituto podem ser utilizadas como métrica para avaliar se houve cópia do modelo?
- **QP2:** O peso dado às características de um modelo alvo pode ser comparado com o peso dado às características de um modelo substituto de maneira *Black-box*, ou seja, sem ter acesso ao modelo alvo treinado, apenas às suas respostas?

- **QP3:** Dada a previsão de limitações e a complexidade inerente à obtenção de soluções definitivas para problemas complexos, pode-se identificar algum método considerado 'bala de prata' para geração de amostras sintéticas e minimização do uso de dados reais?

As contribuições deste trabalho são:

- Método para realização de um ataque de roubo de modelo contra classificadores de vulnerabilidades de rede com o foco em identificar as contribuições de cada característica no modelo alvo, e possibilitar a realização de ataques adversariais;
- Abordagem *black-box* da técnica *Explainability-Based Feature Agreement*, chamada Concordância de Características Caixa-Preta Baseada em Explicabilidade, ou EBFA_BB;
- Comparar os resultados de ataques de roubo de modelo contra classificadores de vulnerabilidade de rede tradicional e IoT, bem como avaliar se o valor de acurácia, métrica comumente utilizada pelos trabalhos que tratam sobre roubo de modelo, pode ser válida para a criação de um modelo substituto com as mesmas características internas que o modelo alvo.

Organizamos o artigo de forma que a seção 2, apresenta o referencial teórico. Nas Seções 3 e 4 apresentamos nossa proposta de ataque, experimentos e resultados alcançados. A seção 5 apresenta a conclusão e os trabalhos futuros.

2. Trabalhos Relacionados

Grande parte dos trabalhos relacionados com as técnicas sobre roubo de modelo e ataques adversariais são focadas em classificadores que utilizam dados de imagens [Oliynyk et al. 2023], cujas representações já são bastante conhecidas e que, pela própria característica do artefato representado, são mais facilmente modificadas e geram poucos problemas quando essas são realizadas.

Considerando trabalhos que atuaram no contexto em que as informações internas sobre os modelos alvos são conhecidas, Tràmer [Tramèr et al. 2016] discute como modelos confidenciais, desenvolvidos a partir de dados sensíveis, podem ser subvertidos por meio de interfaces de consultas públicas. Jagielski, Calini, Bethelot, Kurakin e Papernot [Jagielski et al. 2020], definem uma taxonomia, diferenciando os ataques pelos seus objetivos (Acurácia ou fidelidade) e propões um ataque com foco em redes neurais. Porém, para o nosso escopo, ambos os trabalhos atuam no contexto *white-box*, fugindo da realidade dos classificadores de rede *in the wild*. Além disso, o primeiro usa a acurácia como métrica de validação do modelo substituto e essa métrica não foca na explicabilidade local do modelo, apenas na sua explicabilidade global.

No contexto de ataques *black-box*, Orekondy, Schiele e Fritz [Orekondy et al. 2019] propõem um ataque que imita uma competição entre "vítima" e "atacante" em uma estratégia de incremento de amostragens, incluindo dados reais e dados sintéticos. Já Papernot, McDaniel e Goodfellow [Papernot et al. 2017], evidenciam a importância do processo de roubo de modelo na execução de ataques adversariais. O modelo alvo é utilizado como um oráculo, no qual geram-se rótulos para esses dados sintéticos. Com esses dados rotulados, obtém-se um modelo treinado que pode ser utilizado para criação de novos dados sintéticos utilizando do método chamado de *Jacobian-based Augmentation*. Ambos os trabalhos contribuíram significativamente

para a área, entretanto ambos utilizaram a métrica da acurácia para medir a transferência das características do alvo para o substituto, bem como concentraram seus esforços no domínio das imagens, o qual é muito mais simplificado que o domínio de aplicações de redes de computadores; consequentemente, o primeiro trabalho requer um esforço de adaptação maior principalmente na garantia de que o modelo substituo realmente é uma cópia do modelo alvo, ao passo que o segundo utiliza-se de amostras totalmente sintéticas rotuladas com o potencial de gerar distribuições que fogem da realidade do domínio alvo.

Ainda em um contexto *black-box*, Truong, Walls, Maini e Papernot [Truong et al. 2021] apresentam uma abordagem *Data-free*, na qual geram dados baseados em uma *Generative Adversarial Network* sem fazer uso de dados de domíno utilizados no treinamento do modelo alvo. Entretanto, além do trabalho focar suas ações no domínio das imagens, o uso de dados gerados por uma GAN pode criar amostras que fogem da distribuição do domínio avaliado, no nosso caso, o fluxo de rede.

No melhor de nossos esforços, este é o primeiro trabalho que direciona a análise de ataque de roubo de modelos contra classificadores de vulnerabilidade de rede tradicional e também com foco em IoT, bem como apresenta técnica *black-box* para avaliação do nível de concordância da contribuição das características das amostras no resultado do modelo alvo e modelo substituto, viabilizando ataques adversariais com foco na alteração dessas características.

Tabela 1. Ataques de Roubo de Modelo e técnica utilizada na avaliação do modelo substituto. NN = Redes Neurais, CNN = Redes Neurais Convolucionais; Bb = Black-box, Wh = White-box

Ataque	Modelo Alvo	Domínio	Validação	Contexto
[Orekondy et al. 2019]	NN	Imagens	Acurácia	Bb
[Jagielski et al. 2020]	NN	Imagens	Sinal ReLU	Wb
[Papernot et al. 2017]	NN	Imagens	Acurácia	Bb
[Tramèr et al. 2016]	Diversos	Diversos	Equação	Wb
[Truong et al. 2021]	CNN	Imagens	Erro	Bb
Nosso trabalho	Diversos	Fluxo de rede	EBFA_BB	Bb

3. Concordância de Características Caixa-Preta baseada em Explicabilidade

Para muitas técnicas de aprendizado de máquina, é difícil termos uma ideia clara de como o modelo, após treinamento com os respectivos dados de domínio, atribuem importância para cada característica de maneira direta, sem utilização de outros métodos. Para resolver esse problema, diversos métodos foram apresentados e um deles chama-se *SHapley Additive exPlanations* ou SHAP [Lundberg and Lee 2017], no qual a técnica utiliza-se dos Shapley Values em que a importância das variáveis é diferente conforme a ordem na qual são distribuídas. Testando todas as ordens possíveis e calculando a média, encontram-se os *Shapley Values*. Nesse contexto, a importância das variáveis pode ser considerada como a contribuição dessa variável para o resultado do modelo. Todavia, a abordagem utilizada no método SHAP é bastante custosa computacionalmente e envolve o conhecimento interno do modelo, dificultando uma abordagem *black-box*.

Além disso, podemos citar o processo conhecido como *Explainability-Based Feature Agreement* (EBFA) [Severi et al. 2021], o qual utiliza-se do SHAP para identificar a importância das características para o modelo alvo e o modelo substituto e, após isso, verificar o valor percentual da intersecção das k características mais importantes [Rigaki and Garcia 2023].

Por conta do EBFA ser realizado no contexto *white-box*, principalmente pela utilização do SHAP, buscou-se na literatura métodos e aproximações que permitiriam chegar nos *Shapley Values* sem o custo do SHAP. Neste trabalho, a técnica escolhida para viabilizar a utilização do EBFA sem informações do alvo é o *Sampling SHAP*, de Erik trumbelj e Igor Kononenko [Štrumbelj and Kononenko 2014], na qual são realizados sorteios de amostras contidas em um conjunto de dados de fundo. Após isso, dados de interesse e os dados sorteados têm os valores de suas características permutadas entre si e a média das diferenças entre os resultados antes e após da permutação são próximos aos valores Shapley.

Para adequação ao contexto deste artigo, nós propomos uma versão diferenciada da técnica EBFA, a qual chamamos de Concordância de Características Caixa-Preta baseada em Explicabilidade ou EBFA_BB. Nela, realizamos a substituição da técnica SHAP pela técnica Sampling SHAP, o que evita a necessidade de retreinamento do modelo alvo. Além disso, fixamos o uso do método de *Kernel SHAP Explainer* tendo em vista que no contexto *black-box*, as informações sobre qual função de aprendizado de máquina foi utilizada no sistema alvo estarão indisponíveis e aquele método permite ser usado para avaliar a saída de qualquer tipo de função. Por fim, utilizamos como dados de fundo para realização do *Sampling SHAP*, centroides normalizados calculados pela técnica *Kmeans* dos dados, coletados no mesmo domínio, porém separados por classes e diferentes daqueles utilizados no treinamento do modelo. Os fatores de aleatoriedade empregados tanto na execução do *Sampling SHAP* como na escolha de amostras representativas para criarem os centroides utilizados como dados de fundo ocasionam variações nos *Shapley Values*, porém esse risco foi aceitável considerando a realização do método no ambiente *black-box*, sem o uso das amostras utilizadas no treinamento do modelo alvo.

3.1. Modelo de Ataque Black-Box

Neste trabalho, propomos a utilização de técnicas de roubo de modelo com o objetivo específico de realizar-se um ataque adversarial contra modelos de classificação de vulnerabilidade de rede e também em ambiente IoT. Os ataques foram baseados exclusivamente em acesso à API de classificação e recebimento de respostas.

No contexto estabelecido, ou seja, adquirir informações sobre a importância que o alvo atribui a características utilizadas durante seu treinamento, o modelo substituto possui, como principal função, apresentar um percentual das principais características cujo objetivo é direcionar ataques adversariais, ou seja, definir quais as melhores características um atacante pode direcionar seus esforços para que o ataque adversarial seja realizado. Dessa forma, fixou-se, como modelos substitutos, algoritmo baseados em árvore, visto que, por conta de atributos intrínsecos à sua construção, podem indicar o grau de importância que são atribuídos às características utilizadas em seu treinamento.

A utilização dos dados de domínio ocorreu da seguinte forma: um valor inicial

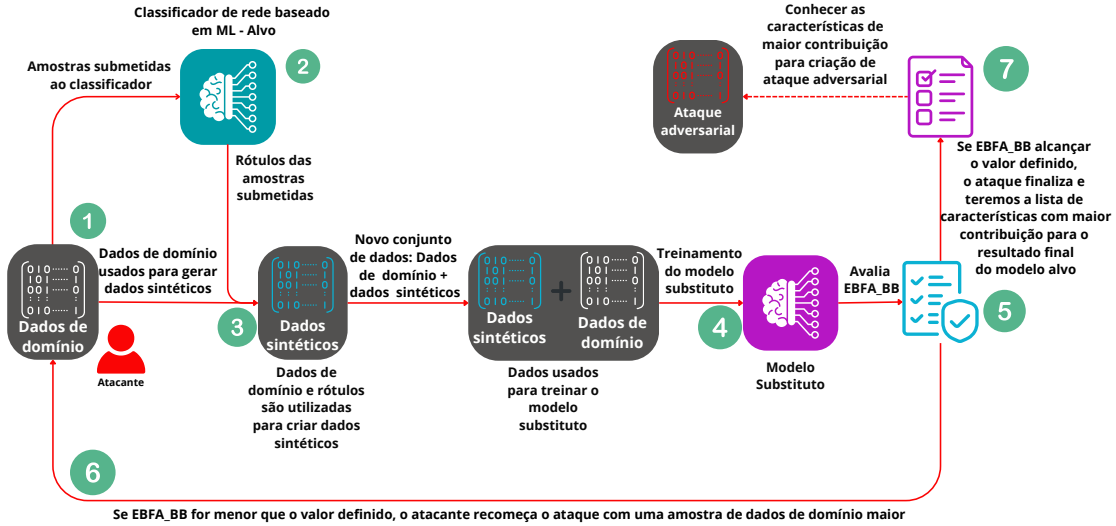

Figura 1. Modelo de ataque proposto neste trabalho

de dados reais é definido e esses dados são classificados pelo sistema alvo (Números 1 e 2 da Figura 1). O conjunto composto por dados e classes é utilizado para a criação de novos dados sintéticos (Número 3 da Figura 1). Para as técnicas definidas, considera-se que os dados sintéticos criados manterão proximidade com os dados originais e, por conta disso, atribui-se a eles as mesmas classes, evitando acesso ao oráculo (alvo). O conjunto de dados reais e sintéticos são utilizados para treinar o modelo substituto baseado em árvore(Número 4 da Figura 1). Para avaliar se as características do modelo alvo foram transferidas para o modelo substituto, utiliza-se a técnica EBFA_BB (Número 5 da Figura 1), utilizando-se, como dados de fundo, os centroides calculados com base na amostra utilizada no ciclo atual. Caso o valor do EBFA_BB esteja abaixo do estabelecido pelo atacante, o ciclo repete-se com o incremento de dados reais (Número 6 da Figura 1). Caso o valor do EBFA_BB atinja o valor definido (Número 7 da Figura 1), então teremos, dentro do percentual escolhido, as características com maior contribuição iguais em ambos os modelos (alvo e substituto). As fases do modelo proposto podem ser observadas na Figura 1. Os mesmos passos podem ser observados no Algoritmo 1.

4. Experimentos e Resultados

A arquitetura utilizada para realização dos experimentos segue conforme Figura 1, no qual podemos observar a existência de um classificador de vulnerabilidade de rede para IDS e em ambiente IoT; a utilização de dados de domínio e dados sintéticos para treinamento do modelo substituto; a utilização do método proposto neste artigo, EBFA_BB para verificação da importância das características no modelo alvo e a possibilidade de uso desta lista de características importantes na elaboração de ataques adversariais.

4.1. Conjunto de dados utilizados

4.1.1. CIC-IDS2017:

O conjunto inicial de dados utilizado no experimento [Sharafaldin et al. 2018] trata sobre um *dataset* com o objetivo de detecção de vulnerabilidades e intrusão de rede para

Algorithm 1: EBFA_BB para obtenção do modelo substituto

```
Modelo_Alvo, EBFA_BB_Esperado, Incremento_Dados, Tecnica_Sintetico,
Numero_Dados_Inicial Modelo_Substituto, Caracteristicas_Importantes,
Dados_Reais
Inicialize Modelo_Substituto
Dados_Reais ← Numero_Dados_Inicial
while EBFA_BB_Esperado > EBFA_BB do
    Dados_Reais, classes ← Classifique_Dados(Dados_Reais, Modelo_Alvo)
    Dados_Sinteticos ← Gera_Dados_Sinteticos(Dados_Reais,
      Tecnica_Sintetico, Classe_Dados_Reais)
    Dados_Treinamento ← Combine(Dados_Reais, Dados_Sinteticos)
    Treina_Modelo(Modelo_Substituto, Dados_Treinamento)
    EBFA_BB ← Calcula_EBFA_BB(Modelo_Alvo, Modelo_Substituto)
    Dados_Reais ← Dados_Reais + Incremento_Dados
Caracteristicas_Importantes ←
  Identifique_Caracteristicas(modeloSubstituto)
return Modelo_Substituto, Caracteristicas_Importantes
```

IDS. No trabalho referenciado, foram criados perfis de atacantes executando ataques do tipo *Brute Force attack, Heartbleed Attack, Botnet, DoS Attack, DDos Attack, Web Attack* e *Infiltration Attack*. As características foram extraídas conforme ferramenta CICFlowMeter [Lashkari et al. 2017], disponível publicamente na página do *Canadian Institute for Cybersecurity*. Para este trabalho, as classes foram condensadas, possuindo apenas as classes Benigno, para fluxos de rede tradicionais; e Malicioso, para fluxos de rede contendo ataques.

4.1.2. ClClo2023:

Outro conjunto de dados utilizado neste trabalho foi o *dataset* de ataques em larga escala em ambiente IoT [Neto et al. 2023], no qual foram realizados 33 tipos de ataques divididos em 7 classes específicas, a saber: *DDoS Attack, Brute Force Attack, Spoofing Attack, DoS Attack, Recon Attack, Web-based Attack* e *Mirai*. Para extração das características foram utilizadas as ferramentas CICFlowMeter [Lashkari et al. 2017] e Nfstream [Aouini and Pekar 2022].

4.1.3. Dados Sintéticos:

Para criação dos dados sintéticos, foram considerados 2 métodos: A interpolação linear regular [Getreuer 2011], a qual selecionam-se dois pontos reais aleatórios para cria um ponto sintético que esteja no espaço intermediário entre os anteriores; A inclusão de ruído gaussiano no conjunto de dados reais [Arslan et al. 2019], gerando novos dados sintéticos. A criação dos dados sintéticos ocorre durante a execução do ataque. Para o conjunto de amostras de dados reais selecionados para cada tentativa de ataque, a mesma quantidade

de dados sintéticos é gerada, conforme técnicas anteriores, e são adicionadas ao conjunto de dados para treinamento do modelo substituto(Números 1,2 e 3 da Figura 1).

4.2. Modelos avaliados

Para gerar um ambiente próximo da realidade, definiu-se que as respostas apresentadas pelas API de classificação conteriam apenas os valores zero (fluxo de rede benigno) e um (fluxo de rede malicioso) e para definição das técnicas de aprendizado de máquina, foram selecionadas algumas daquelas utilizadas nos artigos que criaram os conjuntos de dados. Do trabalho de Sharafaldin [Sharafaldin et al. 2018], para o conjunto de dados para classificação de vulnerabilidades para IDS (CIC-IDS2017): *Random Forest Classifier, Decision Tree Classifier, KNN, QDA Classifier, ADABoost Classifier, Logistic Regression* e *MLP Classifier*. Do trabalho de Neto [Neto et al. 2023], para o conjunto de dados para classificação de ataques em ambiente IoT (ClClo2023): *Logistic Regression* e *Random Forest Classifier*. Em nenhum dos trabalhos citados foram fixados valores para os hiperparâmetros dos respectivos modelos. Dessa forma, neste trabalho utilizaram-se os valores padrão para hiperparâmetros, conforme definido na documentação da biblioteca scikit-learn. Importante mencionar que como premissa para nossa avaliação, o escopo deste artigo considera que os ataques são bem-sucedidos aqueles capazes de criar um modelo substituto que possibilite elencar as características de maior contribuição no resultado do modelo alvo, independente da técnica utilizada no modelo alvo. Neste caso, usamos o *sampling* SHAP para obter dos datasets (CIC-IDS2017, ClClo2023, e dados sintéticos) as quinze características com maior contribuição para o resultado final da função de classificação.

4.3. Experimentos

Os experimentos foram divididos na adaptação de trabalhos anteriores para estabelecimento de uma *baseline* e na avalição da nossa proposta de ataque.

Baseline: abordagem estado da arte que utilizam acurácia como métrica. Nesta, o objetivo era avaliar se ataques de roubo de modelo utilizando unicamente dados de domínio, como adaptado do trabalho de Tràmer [Tramèr et al. 2016] e ataques utilizando-se de técnicas de aumento de dados, como adaptado do trabalho de Orekondy [Orekondy et al. 2019] conseguiriam gerar modelos substitutos. Exatamente como feito nos trabalhos referenciados, utilizou-se da acurácia para validar a criação do modelo substituto e verificar se o fato da acurácia de ambos serem iguais significaria um alto valor na intersecção entre as quinze características com maior contribuição no resultado final de cada um, ou seja, as características mais importantes em ambos os modelos seriam equivalentes. Neste experimento, as atividades foram realizadas em um contexto *white-box* e foram realizados testes em que o alvo e o modelo substituto utilizavam a mesma técnica de aprendizado de máquina. Como complemento, realizou-se testes fixando o modelo substituto em uma *Random Forest Classifier*. Os ataques foram realizados contra os modelos de classificadores de rede para IDS (CIC-IDS2017).

Para viabilizar esse experimento, foram estabelecidos os seguintes critérios: Valores de acurácia do modelo alvo e substituto exatamente iguais. Nesse caso, o teste avançaria para o cálculo do EBFA de ambos os modelos. Caso os valores de acurácia

não fossem iguais (métrica fundamental para os experimentos *baseline* e estabelecidos nos artigos originais), até o limite de 2420 amostras de dados reais, o teste seria parado, indicando que o ataque não foi bem-sucedido e o valor de EBFA não seria calculado. O valor de EBFA calculado neste experimento utilizou-se da técnica SHAP e o cálculo de acurácia. Ambos as métricas necessitaram de dados utilizados durante o treinamento do modelo, configurando um contexto *white-box*. Os resultados podem ser vistos nas tabelas 2 e 3.

Abordagem proposta usando EBFA_BB Neste experimento, o objetivo foi replicar os ataques do experimento *baseline*, porém utilizando o nosso ataque. Após isso, utilizamos nossa técnica contra os classificadores de vulnerabilidade em ambiente IoT. Em ambos os ataques, consideramos o contexto *black-box*, desconsiderando informações sobre o modelo alvo, porém considerando o conhecimento da forma de representação dos dados utilizados no treinamento dos modelos. Além disso, as respostas das API de classificação permaneceram configuradas para os valores de 0 e 1 (classificador binário).

Para viabilizar a aplicação do nosso ataque, foram estabelecidos os seguintes critérios: Valores de EBFA_BB acima de 0,6 já indicariam valor superior aos maiores valores alcançados na *baseline* e finalizariam o ataque. O ataque seria finalizado caso o valor de EBFA_BB não atingisse o mínimo, até o limite de 4000 amostras reais, com incremento de 400 dados reais a cada ciclo de ataque e os resultados podem ser vistos nas tabelas 4 e 6.

4.4. Análise dos resultados

Para os experimentos realizados conforme definido nos experimentos para obtenção do resultado *baseline*, fora realizado ataques baseados simplesmente em enviar dados incrementalmente para serem classificados pelo modelo alvo (*Query-Only*) e juntamente com a resposta alcançada, treinar o modelo substituto até que o valor da acurácia do modelo alvo fosse igual ao valor da acurácia do modelo substituto. Essa atividade foi realizada usando-se a mesma técnica de aprendizado de máquina para o modelo alvo e modelo substituto, alcançando os resultados da tabela 2, lado esquerdo. Após isso, fixou-se uma técnica baseada em árvore como modelo substituto, nesse experimento, uma *Random Forest Classifier*, cujos valores constam da tabela 2, lado direito. O valor de EBFA -1 indica que este teste não foi realizado pelo atingimento de critério de parada do ataque.

Tabela 2. Ataque Query-only e ataque Query-only usando Random Forest como modelo substituto

Modelo	Amostras	Acurácia	EBFA	Modelo	Amostras	Acurácia	EBFA
ADABoostClassifier	732	0,980	0,463448	ADABoostClassifier	1660	0,983	0,459310
KNNClassifier	2420	0,957	-1	KNNClassifier	2420	0,986	-1
LogisticRegression	2420	0,802	-1	LogisticRegression	188	0,862	0,466207
MLPClassifier	2420	0,482	-1	MLPClassifier	260	0,738	0,478621
QDAClassifier	984	0,800	0,554483	QDAClassifier	796	0,770	0,304828
RandomForestClassifier	2420	0,988	-1	RandomForestClassifier	2420	0,990	-1

Em ambos os resultados, ocorreram casos em que o critério limite de amostras reais para término do ataque foi atingido sem alcançar igualdade entre o valor da acurácia

do modelo alvo e do modelo substituto, impedindo assim o cálculo do EBFA e verificação da equivalência na contribuição das características dos dados.

A partir deste momento, realizamos os mesmos ataques anteriores, porém utilizando dados reais aumentados com dados sintéticos na criação do modelo substituto. Para esses testes, alcançamos os resultados presentes na tabela 3, no lado esquerdo para os testes utilizando técnica do modelo alvo e substituto iguais e no lado direito fixando-se a técnica *random forest* como modelo substituto.

Tabela 3. Ataque Query-only com ruído gaussiano e ataque Query-only com ruído gaussiano e Random Forest como modelo substituto

Modelo	Amostras	Acurácia	EBFA	Modelo	Amostras	Acurácia	EBFA
ADABoostClassifier	1084	0,979	0,500559	ADABoostClassifier	1752	0,983	0,340112
KNNClassifier	2420	0,949	-1	KNNClassifier	2420	0,983	-1
LogisticRegression	2420	0,573	0,448268	LogisticRegression	548	0,864	0,348603
MLPClassifier	2420	0,500	0,413184	MLPClassifier	148	0,722	0,397542
QDAClassifier	2420	0,779	0,371173	QDAClassifier	108	0,798	0,372737
RandomForestClassifier	2420	0,983	-1	RandomForestClassifier	2420	0,983	-1

A utilização de dados sintéticos trouxe alguns ganhos para os resultados alcançados, porém considerando os critérios previamente estabelecidos, ainda existem modelos alvo em que a acurácia, usada como critério, impede que valores de EBFA sejam calculados, tendo em vista que alcançam o limite máximo estabelecido. Ou seja, para o critério estabelecido, o ataque foi mal-sucedido.

Dessa forma, observamos que ao utilizarmos a acurácia para criação do modelo substituto, estaríamos focando apenas na explicabilidade global das características e com isso, poderíamos perder informações sobre a lógica com a qual o modelo funciona internamente. No nosso contexto, perderíamos informações sobre o quanto cada característica utilizada no treinamento contribui para o resultado final. A partir dos resultados alcançados até este momento, podemos estabelecer resultados mínimos esperados, considerando que as abordagens anteriores estavam no contexto *white-box* e o ataque proposto neste artigo avança em direção a um ataque *black-box*.

O experimento para avaliação do EBFA_BB, inicialmente, atacamos os mesmos alvos utilizados no experimento *baseline* fazendo uso do ataque proposto neste artigo, utilizando, como modelo substituto, a técnica *Decision Tree Classifier*. Assim, alcançamos os seguintes resultados da tabela 4.

Tabela 4. Nosso ataque, Ruído Gaussiano e Decision Tree Classifier como modelo substituto

Modelo	Amostras	EBFA_BB
ADABoostClassifier	400	0,8407
KNNClassifier	400	0,8099
LogisticRegression	400	0,9267
MLPClassifier	400	0,8587
QDAClassifier	400	0,8627
RandomForestClassifier	3600	0,6673

Dessa forma, podemos observar que os resultados do nosso ataque foram muito superiores que os resultados alcançados anteriormente. Independente de algum erro existente, relacionado à falta de acesso aos dados originais utilizados no treinamento do modelo em contexto *black-box*, mas sim com dados de domínio que geram aproximações nos valores *Shapley*.

Para validarmos o resultado alcançado, vamos utilizar as características inerentes aos modelos baseados em árvore para listar as quinze características mais importantes do modelo alvo e do substituto e avaliamos o erro existente, relacionado na tabela 5.

Tabela 5. Quinze principais características dos modelos substituto e alvo. Em negrito, as características comuns a ambos.

ID	Característica - Modelo substituto	ID	Característica - Modelo alvo
51	**Average Packet Size**	12	Bwd Packet Length Std
53	**Avg Bwd Segment Size**	**11**	**Bwd Packet Length Mean**
17	Flow IAT Max	**40**	**Packet Length Std**
11	**Bwd Packet Length Mean**	**39**	**Packet Length Mean**
62	Subflow Fwd Bytes	**51**	**Average Packet Size**
52	**Avg Fwd Segment Size**	**65**	**Init_Win_bytes_forward**
5	**Fwd Packet Length Max**	**53**	**Avg Bwd Segment Size**
40	**Packet Length Std**	9	Bwd Packet Length Max
3	Total Length of Fwd Packets	38	Max Packet Length
39	**Packet Length Mean**	41	Packet Length Variance
75	Idle Max	66	Init_Win_bytes_backward
65	**Init_Win_bytes_forward**	**5**	**Fwd Packet Length Max**
76	Idle Min	**52**	**Avg Fwd Segment Size**
34	Bwd Header Length	7	Fwd Packet Length Mean
22	Fwd IAT Max	1	Total Fwd Packets

Após identificarmos as características mais importantes e avaliarmos a intersecção entre elas, observamos que, nesse caso, oito das quinze principais características estão nos dois modelos, conforme Tabela (5), equivalente a 53,33% das características. Calculando a diferença entre o valor de EBFA_BB e o valor real, isso pode nos apresentar um possível valor de erro do nosso método. Nesse caso, aproximadamente 0,13% de erro encontrado.

Vale ressaltar que diversos fatores podem interferir nos valores de erro do EBFA_BB, tais como a seleção de amostras para servir de dados de fundo, os quais não pertencem ao conjunto de dados de treinamento do modelo alvo; o efeito de sorteio realizado pela técnica *Sampling SHAP* e *Kernel Explainer* etc. Independentemente disso, os valores alcançados pelo método foram bastante altos e isso nos indica que, mesmo com o erro presente, o modelo substituto conseguiu capturar informações sobre importância das características do modelo alvo e, no contexto desse trabalho, indica que pode ser um elemento facilitador na realização de ataques adversariais.

Mantendo o contexto da classificação de ataques em rede, porém em ambientes IoT, realizamos nosso ataque contra classificadores de ataques em rede IoT. Os resultados podem ser observados conforme tabela 6. Da mesma forma como ocorreu no experimento

anterior, após definição do valor de EBFA_BB, os resultados foram bastante promissores. Além disso, podemos ter uma expectativa de quantas características nós poderemos focar no modelo substituto para realizar o ataque adversarial.

Tabela 6. Resultados do nosso ataque nos modelos de classificar de ataques em ambiente IoT

Modelo	Modelo substituto	EBFA_BB	Amostras
LogisticRegression	Random Forest Classifier	1,0	400
LogisticRegression	Decision Tree Classifier	0,6020	3200
RandomForestClassifier	Random Forest Classifier	1,0	400
RandomForestClassifier	Decision Tree Classifie	0,6840	400

4.5. Respostas às Questões de Pesquisa

Como resposta para ***QP1***, podemos afirmar que se o objetivo de realizar o roubo do modelo for apenas replicar o seu desempenho, ou seja, proporcionar explicabilidade global do alvo, acreditamos que a acurácia pode ser válida como métrica. Porém, se o objetivo for identificar informações funcionais, ou seja, explicabilidade local das características do alvo, acreditamos que os valores de acurácia podem falhar em refletir o grau de contribuição que determinada característica emprega no resultado final.

Como resposta para ***QP2***, podemos dizer que uma forma de avaliar o peso dados às características de uma modelo de maneira *black-box* foi apresentado neste trabalho, chamado EBFA_BB, no qual são utilizados dados do domínio em questão, e a contribuição é calculada por meio da técnica de *Sampling SHAP*.

Como resposta para ***QP3***, podemos dizer que é impossível afirmar, dentro do nosso conhecimento, que exista um método para gerar dados sintéticos que sempre replique as características dos dados reais. Durante nossos experimentos, foi possível observar que resultados distintos foram alcançados, conforme mudança na escolha do método para gerar dados sintéticos. Consideramos que diversos fatores podem contribuir para a escolha do método, tal como o fator de aleatoriedade que algumas técnicas utilizam, durante seu treinamento, e que todos devem ser testados durante a execução dos ataques.

5. Conclusão

Neste trabalho, apresentamos um método de roubo de modelo com foco na equivalência da contribuição das características do modelo alvo. O ataque proposto conseguiu criar um modelo substituto que replicava essas contribuições em, pelo menos, 10% a mais que os ataques usados como *baseline* e que utilizavam a acurácia como métrica. Com o conhecimento de informações internas indisponíveis do Sistema de Detecção de Instrusão alvo, um atacante pode direcionar a realização de ataques adversariais. Ainda neste trabalho, apresentamos uma abordagem caixa-preta da técnica EBFA, a qual chamamos de EBFA_BB que possibilitou a realização do ataque proposto sem que se tenha os dados de treinamento do modelo alvo.

Ainda em termos comparativos, realizou-se ataques que utilizam a comparação entre acurácia do modelo alvo e substituto como métrica e verificamos que, no contexto

de classificadores de redes e para adquirir informações para a realização de ataques adversariais, tal métrica foi incapaz de avaliar a captura de informações da contribuição das características do modelo.

Os resultados alcançados neste trabalho podem ser evoluídos em diversos aspectos e, como trabalhos futuros pretendemos: avaliar o uso de outras métricas durante o treinamento do modelo substituto, tal como F1-Score, Precisão e Recall, e a influência delas na capacidade de replicação do comportamento interno do modelo alvo; Realizar análises multiclasse dos datasets; Investigar outras formas de seleção de dados reais ou criação de dados sintéticos utilizando aprendizado por reforço; Por fim, considerando a falta de trabalhos nesta área de conhecimento, realizar a comparação com outros trabalhos que surgirem, com o objetivo de realizar novas comparações de resultados.

Todos os códigos utilizados neste trabalho estão disponibilizados em `https://github.com/c2dc/sbrc2024-ebfa_bb`.

Agradecimentos

Este trabalho tem apoio financeiro do Programa de Pós-graduação em Aplicações Operacionais—PPGAO/ITA, da Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) processo #2020/09850-0, e da CAPES.

Referências

Alshaikhli, M., Elfouly, T., Elharrouss, O., Mohamed, A., and Ottakath, N. (2022). Evolution of internet of things from blockchain to iota: A survey. *IEEE Access*, 10:844–866.

Aouini, Z. and Pekar, A. (2022). Nfstream: A flexible network data analysis framework. *Computer Networks*, 204:108719.

Arslan, M., Guzel, M., Demirci, M., and Ozdemir, S. (2019). Smote and gaussian noise based sensor data augmentation. In *2019 4th International Conference on Computer Science and Engineering (UBMK)*, pages 1–5.

Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G., and Roli, F. (2013). Evasion attacks against machine learning at test time. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2013, Prague, Czech Republic, September 23-27, 2013, Proceedings, Part III 13*, pages 387–402. Springer.

Consult, T. (2022). Isg provider lens internet of things services and solutions brazil 2022. `https://www.tgt.com.br/blog/2022/08/10/isg-provider-lens-internet-of-things-services-and-solutions-brazil-2022/`. Accessed: jan. 15, 2024.

Domingues, M., Bertoli, G., de Melo, L., Saotome, O., Santos, A., and Pereira, L. (2022). Avaliação da capacidade de generalização de ids stateful utilizando aprendizado de máquina. In *Anais do XXII Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais*, pages 236–249, Porto Alegre, RS, Brasil. SBC.

Getreuer, P. (2011). Linear methods for image interpolation. *Image Processing On Line*, 1:238–259.

Huang, L., Joseph, A. D., Nelson, B., Rubinstein, B. I., and Tygar, J. D. (2011). Adversarial machine learning. In *Proceedings of the 4th ACM workshop on Security and artificial intelligence*, pages 43–58.

Jagielski, M., Carlini, N., Berthelot, D., Kurakin, A., and Papernot, N. (2020). High accuracy and high fidelity extraction of neural networks. In *29th USENIX security symposium (USENIX Security 20)*, pages 1345–1362.

Lashkari, A. H., Gil, G. D., Mamun, M. S. I., and Ghorbani, A. A. (2017). Characterization of tor traffic using time based features. In *International Conference on Information Systems Security and Privacy*, volume 2, pages 253–262. SciTePress.

Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Neto, E. C. P., Dadkhah, S., Ferreira, R., Zohourian, A., Lu, R., and Ghorbani, A. A. (2023). Ciciot2023: A real-time dataset and benchmark for large-scale attacks in iot environment.

Oliynyk, D., Mayer, R., and Rauber, A. (2023). I know what you trained last summer: A survey on stealing machine learning models and defences. *ACM Computing Surveys*.

Orekondy, T., Schiele, B., and Fritz, M. (2019). Knockoff nets: Stealing functionality of black-box models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4954–4963.

Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., and Swami, A. (2017). Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519.

Rigaki, M. and Garcia, S. (2023). The power of meme: Adversarial malware creation with model-based reinforcement learning. *arXiv preprint arXiv:2308.16562*.

Severi, G., Meyer, J., Coull, S., and Oprea, A. (2021). Explanation-Guided backdoor poisoning attacks against malware classifiers. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 1487–1504. USENIX Association.

Sharafaldin, I., Lashkari, A. H., and Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp*, 1:108–116.

Štrumbelj, E. and Kononenko, I. (2014). Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41:647–665.

Tramèr, F., Zhang, F., Juels, A., Reiter, M. K., and Ristenpart, T. (2016). Stealing machine learning models via prediction {APIs}. In *25th USENIX security symposium (USENIX Security 16)*, pages 601–618.

Truong, J.-B., Maini, P., Walls, R. J., and Papernot, N. (2021). Data-free model extraction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4771–4780.