# Detecção de ataques DDoS usando correlação espaço-temporal bayesiana

**Gabriel Mendonça[1], Gustavo H. A. Santos[1], Edmundo de Souza e Silva[1], Rosa M.M. Leão[1]**

[1]Universidade Federal do Rio de Janeiro (UFRJ), Rio de Janeiro, RJ.

{gabriel,gustavo,edmundo,rosam}@land.ufrj.br

***Resumo.*** *Ataques DDoS têm causado prejuízos consideráveis ao longo dos anos. Para mitigar seu impacto, a detecção deve ocorrer preferencialmente próximo à origem. Propomos neste trabalho um sistema leve de detecção de DDoS que usa apenas contadores de bytes e pacotes de roteadores domésticos. Para detectar ataques com informações limitadas, empregamos duas camadas: (1) um classificador treinado com dados reais de usuários domésticos; (2) um modelo hierárquico bayesiano que correlaciona alarmes de várias residências. Usamos código-fonte de malwares reais para gerar tráfego de ataque DDoS nas casas de um grupo de voluntários durante 31 dias. Os experimentos realizados em campo mostraram que nosso sistema possui excelente desempenho.*

***Abstract.*** *DDoS attacks have caused considerable damage over the years. To mitigate their impact, detection should preferably occur close to the attack origin. We propose in this work a lightweight DDoS detection system that solely employs byte and packet counts from off-the-shelf home routers. To detect attacks with limited information, our key insight consists in employing two detection layers: (1) a classifier trained with real home user data; (2) a Bayesian hierarchical model that correlates alarms from multiple homes. We use real IoT malware source code to collect DDoS attack data, generating attack traffic from the homes of a selected group of volunteers for 31 days. The field experiments have shown that our system has excellent performance.*

## 1. Introdução

Ataques DDoS estão entre os mais comuns na Internet [NETSCOUT 2021, Kaspersky 2021] e, apesar dos muitos esforços realizados nos últimos anos para mitigar seus danos, eles ainda representam uma grande fonte de preocupação. Durante a pandemia de COVID-19, vários provedores de serviços de mitigação de DDoS relataram um aumento acentuado desses ataques, provavelmente motivados pelo aumento da dependência de conectividade remota [DARKReading 2020].

Os dispositivos IoT, que desempenham o papel de alvos e fontes de diferentes tipos de ataques, são especialmente vulneráveis. Esses dispositivos *plug-and-play* são frequentemente instalados pelos usuários na rede doméstica, raramente recebendo atualizações de *firmware* e frequentemente mantendo as credenciais padrão ativas, abrindo oportunidades para *backdoors* e *exploits* para formar grandes *botnets* [Antonakakis et al. 2017, Kolias et al. 2017].

*Mirai*, um *malware* usado para construir uma grande *botnet* que atingiu uma população com 200.000 a 300.000 dispositivos, produziu em 2016 um ataque recorde que atingiu o pico de 623 Gbps [Akamai 2016]. Relatórios recentes mencionam que *botnets* baseadas em *Mirai* continuam a prosperar [Yoachimik 2021, Kaspersky 2021, NETSCOUT 2021]. Novas variantes do *Mirai* continuam a surgir usando mais dispositivos e explorando novas vulnerabilidades, mantendo praticamente inalterado o código-fonte dos vetores de ataque (por exemplo, *UDP flood*, *TCP SYN flood*).

O aumento dos ataques DDoS impõe novos desafios para sua detecção. A recente prevalência de tráfego *web* criptografado, por exemplo, tem demandado o redesenho das ferramentas atuais de monitoramento [Bihary 2017]. Para mitigar o impacto de ataques DDoS baseados em *botnet* lançados por dispositivos IoT, a detecção deve ocorrer preferencialmente perto da origem do ataque. Com soluções tradicionais que operam no núcleo da rede [Silveira et al. 2011, Liaskos et al. 2016, Marín et al. 2021, Nevat et al. 2018], a detecção geralmente ocorre longe da origem, dificultando a minimização do impacto dos ataques. Por outro lado, soluções centradas nos terminais (*hosts*) [Sedjelmaci et al. 2017, Summerville et al. 2015] podem ser eficazes para localizar e isolar aparelhos infectados, mas sua execução pode não ser viável em dispositivos IoT com recursos limitados.

A detecção de ataques DDoS em roteadores domésticos pode ser vista como uma alternativa melhor, uma vez que esses dispositivos estão próximos das fontes de ataque e, frequentemente, podem ser gerenciados remotamente por ISPs. No entanto, esses dispositivos são normalmente limitados em termos de memória e poder de processamento. Em particular, o uso de inspeção de pacotes ou informações coletadas de cabeçalhos de pacotes não é adequado para esses dispositivos. Além disso, a detecção não deve interferir no desempenho dos usuários domésticos nem violar sua privacidade. Embora a detecção de ataques DDoS em roteadores domésticos tenha sido considerada recentemente [Meidan et al. 2018, Doshi et al. 2018, Wan et al. 2020, Anthi et al. 2019, McDermott et al. 2018, Salman et al. 2019], as soluções propostas dependem de informações extraídas de *traces* de pacotes para detecção de ataques (por exemplo, endereços IP de origem e destino, protocolos, portas). Além disso, os resultados relatados vêm de *testbeds* de IoT com um número restrito de dispositivos. Nosso trabalho em [Mendonça et al. 2019b, Mendonça et al. 2019a] é uma exceção, já que não usa informações extraídas de pacotes.

É importante observar que a maioria dos ataques DDoS é sincronizada e reúne inúmeros dispositivos. Portanto, é de se esperar que haja alguma correlação entre o tráfego de residências infectadas distintas (que contenham um dispositivo IoT infectado por um *malware*). E mesmo dispositivos que pertençam a *botnets* diferentes podem lançar ataques DDoS para a mesma vítima simultaneamente [Wang et al. 2018]. Nosso trabalho em [Mendonça et al. 2019a] fez uso dessa observação. A ideia de executar detectores DDoS em servidores de borda, com ou sem a ajuda de detectores baseados em roteador doméstico, também foi explorada recentemente [Jia et al. 2020, Sudheera et al. 2021, Streit et al. 2021b]. Ainda assim, com exceção de [Streit et al. 2021b], as soluções propostas sofrem dos mesmos problemas de dependência de informações sensíveis dos usuários domésticos (extraídas de cabeçalhos de pacotes) e falta de avaliação com dados realistas.

Neste artigo, estendemos nossos trabalhos anteriores [Mendonça et al. 2019a,

Mendonça et al. 2019b] aprimorando a ideia de aproveitar nos modelos o sincronismo dos ataques DDoS para desenvolver mecanismos de detecção mais robustos. Assim como em [Mendonça et al. 2019a], utilizamos um sistema de duas camadas para a detecção: a primeira camada (*detector local*) roda no nível do usuário doméstico; a segunda (*correlação espaço-temporal*) é executada no nível do provedor de serviço (ISP). Nossas principais contribuições são resumidas a seguir.

(1) **Correlação espaço-temporal.** Assim como em [Mendonça et al. 2019a], consideramos correlações espaciais de casas distintas durante o mesmo intervalo de tempo, explorando o fato de que a maioria dos ataques DDoS são sincronizados. Entretanto, diferentemente de [Mendonça et al. 2019a], a correlação espaço-temporal entre os domicílios é capturada por meio de um modelo hierárquico bayesiano. O modelo é usado para decidir se há um ataque ou não com base no fator de Bayes. Os resultados mostram que a frequência de alarmes falsos pode ser reduzida substancialmente em comparação a [Mendonça et al. 2019a], mantendo a mesma eficiência na detecção de ataques.

(2) **Detecção de ataques DDoS reais.** Realizamos um experimento de campo com um grupo de 10 voluntários para testar o desempenho de nosso sistema. Um dispositivo Raspberry executando ataques DDoS reais, usando código-fonte extraído de *malwares Mirai* e *BASHLITE*, foi colocado na casa de cada voluntário emulando um dispositivo IoT infectado. Durante um período de 31 dias, ataques simultâneos foram lançados pelos dispositivos Raspberry Pi. *O sistema de duas camadas proposto foi capaz de detectar todos os ataques no experimento de campo*, exibindo baixas taxas de falsos positivos. Não temos conhecimento de trabalhos anteriores avaliando o desempenho de um detector DDoS em um cenário real como este.

Este artigo é organizado como segue. A Seção 2 descreve sucintamente o estado da arte para detecção de ataques DDoS. Na Seção 3, apresentamos o sistema de duas camadas proposto. A Seção 4 mostra nossos resultados e a Seção 5 conclui.

## 2. Trabalhos Relacionados

As soluções tradicionais para detecção e mitigação de DDoS se baseiam na análise de fluxos de pacotes, cabeçalhos e/ou *payloads* no núcleo da rede [Silveira et al. 2011, Liaskos et al. 2016, Marín et al. 2021, Nevat et al. 2018]. Embora soluções no núcleo da rede sejam adequadas para detectar diferentes tipos de anomalias, a detecção de um ataque DDoS, se bem-sucedida, geralmente ocorre longe da fonte do ataque, dificultando a mitigação do seu impacto pelo bloqueio das origens do tráfego. Além disso, o processamento de grandes volumes de tráfego leva a problemas de escalabilidade, que podem se traduzir em baixas taxas de detecção [Mehdi et al. 2011].

Os ataques DDoS detectados em roteadores domésticos podem ser facilmente bloqueados na origem, sem que o tráfego de ataque gerado consuma recursos após ingressar na rede do provedor. Adicionalmente, o usuário doméstico ou o ISP podem ser notificados sobre o ataque detectado. Várias soluções para detecção de ataques DDoS nos roteadores domésticos foram propostas anteriormente [Meidan et al. 2018, Doshi et al. 2018, Wan et al. 2020, Anthi et al. 2019, McDermott et al. 2018, Salman et al. 2019]. Todavia, com exceção da nossa abordagem em [Mendonça et al. 2019a, Mendonça et al. 2019b], as soluções propostas dependem de informações sensíveis extraídas do cabeçalho de pacotes de rede. Além disso, os resultados relatados geralmente têm ori-

gem em *sandboxes* restritas com um número limitado de dispositivos IoT (por exemplo, [Meidan et al. 2018, Doshi et al. 2018, Wan et al. 2020, Anthi et al. 2019, McDermott et al. 2018, Salman et al. 2019]).

Outros trabalhos propõem o uso de servidores de borda para detecção de ataques DDoS, como [Sudheera et al. 2021, Streit et al. 2021b, Jia et al. 2020]. Os dois primeiros utilizaram-se da cooperação de detectores rodando em roteadores domésticos, mas não o terceiro. Além de possuírem maior poder computacional do que os roteadores domésticos, servidores localizados na borda da rede são capazes de correlacionar anomalias oriundas de diferentes clientes. Entretanto, não permitem o bloqueio do tráfego na origem, i.e., no roteador ao qual o dispositivo IoT infectado (*bot*) está conectado.

Em [Streit et al. 2021b], outro trabalho do nosso grupo de pesquisa, os autores propõem um algoritmo baseado em decomposição tensorial para detecção de anomalias que não necessita da inspeção de cabeçalhos de pacotes. Os autores também adotam a metodologia de [Mendonça et al. 2019a] para obter seu *dataset*. Entretanto, o modelo utilizado para a correlação dos alarmes de diferentes residências (máximo *a posteriori*, baseado em [Mendonça et al. 2019a]) é bastante limitado quando comparado ao modelo hierárquico bayesiano apresentado neste trabalho. Além disso, os autores não apresentam resultados de experimentos reais de ataques DDoS.

Neste trabalho, aprimoramos o trabalho de [Mendonça et al. 2019a], onde usamos um sistema de duas camadas para detecção de DDoS com uma quantidade mínima de informações de usuários domésticos. Nosso modelo hierárquico bayesiano explora a correlação espaço-temporal entre alarmes de diferentes roteadores domésticos para reduzir consideravelmente a probabilidade de se obter um alarme falso em relação a [Mendonça et al. 2019a].

Estendemos nossos trabalhos anteriores de diversas maneiras. Atualizamos nossos dados com medições recentes, aumentando de 1.823 para 4.870 o número de residências e de 19 milhões para 116 milhões o número de amostras (janelas deslizantes). Com base em nosso modelo MAP anterior (máximo *a posteriori*), desenvolvemos um modelo hierárquico bayesiano muito mais robusto e flexível. Adicionalmente, apresentamos resultados com base em um experimento de campo com um grupo de voluntários realizando ataques DDoS, mostrando que nosso método funciona não apenas em laboratório, mas também quando usado em ambientes reais.

## 3. Sistema de detecção

Nosso objetivo principal é lidar com o seguinte desafio: *como detectar ataques poucos minutos após seu início usando apenas estatísticas de contadores de* bytes *e pacotes?* Nossa metodologia utiliza dados de medições reais de usuários domésticos juntamente com dados de experimentos controlados usando *malware* real. Combinando dados de atividade normal de rede com tráfego malicioso, obtemos os *datasets* necessários para treinar modelos de detecção com eficiência elevada.

Nosso sistema de detecção é composto por duas camadas de segurança: uma camada local no nível do usuário doméstico e uma camada de correlação espaço-temporal no nível do ISP (ver Figura 1). A configuração de duas camadas oferece mais flexibilidade, pois os ataques podem ser mitigados no roteador doméstico ou no ISP dependendo

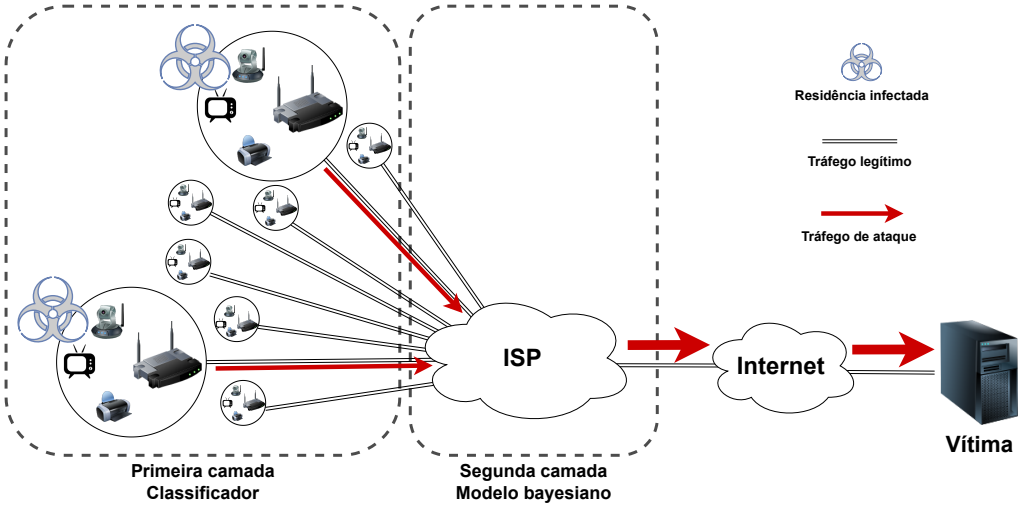



**Figura 1. Arquitetura do sistema de detecção de DDoS proposto.**

dos requisitos de desempenho (taxa de falso-positivo, taxa de verdadeiro-positivo, tempo de detecção). Mostramos que a grande maioria dos ataques pode ser detectada, reduzindo consideravelmente a chance de sucesso de ataques DDoS.

Dentre as $N$ residências monitoradas pelo sistema, assumimos que $b$ residências contêm algum dispositivo infectado. Essas são denominadas *residências infectadas*. Os dispositivos infectados (*bots*) fazem parte de uma *botnet* e respondem de forma sincronizada a um servidor de Comando e Controle (CnC), realizando ataques de modo intermitente. Destacamos que não fazemos quaisquer premissas sobre o tipo de dispositivos IoT conectados à rede doméstica. Alguns trabalhos recentes como [Meidan et al. 2018, Doshi et al. 2018, Wan et al. 2020, Anthi et al. 2019, McDermott et al. 2018, Salman et al. 2019] usam assinaturas de tráfego de apenas 9 dispositivos no máximo. Por outro lado, nosso *dataset* com medições de usuários domésticos contém mais de 10.000 dispositivos distintos.

Seguimos a metodologia proposta em [Mendonça et al. 2019a] para produzir um *dataset* rotulado combinando tráfego real de 4.870 residências com tráfego de ataques DDoS gerado com código-fonte de *malwares Mirai* e *BASHLITE* em laboratório.

### 3.1. Camada de detecção local

A primeira camada de nosso sistema é um classificador de aprendizado de máquina leve capaz de ser executado em roteadores domésticos, conforme apresentado em [Mendonça et al. 2019a]. A partir de amostras coletadas de contadores de *bytes* e pacotes a cada minuto, o classificador usa estatísticas simples (como desvio padrão) calculadas em uma janela de tempo. Consideramos uma *janela deslizante* de 5 minutos que desliza em intervalos de um minuto. As estatísticas são calculadas usando apenas as amostras da janela atual. Consequentemente, obtemos um novo conjunto de estatísticas a cada minuto. Mostramos na Seção 4 que uma janela deslizante de 5 minutos contém informações suficientes para detectar ataques com alta precisão e em menos de 2 minutos na maioria dos casos.

A cada minuto, o classificador usa as taxas de *bytes* (bps) e pacotes (pps) do tráfego de *upload* e *download* da janela atual. Para a janela, quatro métricas são conside-

radas: taxa de bps *upstream*, taxa de pps *upstream*, relação entre taxas de bps *upstream* e *downstream* e razão entre taxas de pps *upstream* e *downstream*. Para cada métrica, são calculadas três estatísticas: desvio padrão, máximo e diferença entre o máximo e o mínimo, totalizando 12 *features* para cada janela. Avaliamos outras estatísticas (média, mediana e mínimo), mas não obtivemos uma diferença significativa no resultado.

Usamos nosso *dataset* rotulado para treinar o classificador. Avaliamos cinco modelos diferentes: Regressão Logística, Árvore de Decisão, *Random Forest*, *Gaussian Naive Bayes* e *Multilayer Perceptron*. O melhor modelo – *Random Forest* – foi selecionado por meio de validação cruzada com 5 *folds* usando um conjunto de dados de treinamento.

### 3.2. Camada de correlação espaço-temporal

A segunda camada do sistema avalia a correlação entre os resultados de casas distintas em uma mesma janela de tempo. A intuição é simples: se apenas um roteador doméstico reporta positivo (*ataque*), é provável que seja um alarme falso. Por outro lado, se observamos um número alto de resultados positivos simultâneos, nossa evidência favorece a hipótese de um ataque DDoS. Nosso modelo hierárquico bayesiano captura o comportamento esperado de um ataque e, usando fatores de Bayes, permite avaliar o quanto nossa observação favorece as hipóteses de *ataque* / *não ataque*. Assim, conseguimos aumentar a probabilidade de detecção e reduzir alarmes falsos, conforme mostrado na Seção 4.

Seja $x_i$ o número total de roteadores domésticos que reportam positivo no tempo $t_i$ dentre os $N$ roteadores que executam o classificador. Nosso problema consiste, então, em decidir entre duas hipóteses com base no valor observado $x_i$:

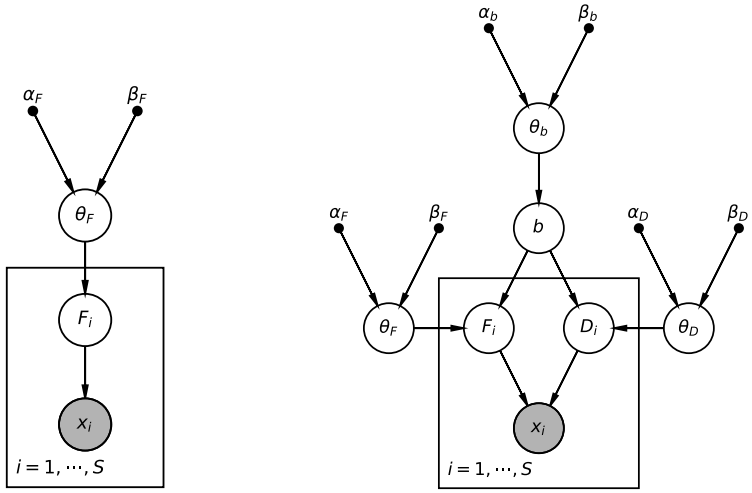
- $H_0$ (*hipótese nula*) — não houve ataque DDoS no período de tempo $t_i$; todos os $x_i$ resultados positivos reportados são *falso-positivos*.
- $H_1$ (*hipótese alternativa*) — houve um ataque DDoS no período de tempo $t_i$; os $x_i$ resultados positivos reportados são uma combinação de *verdadeiro-positivos* (de casas que participaram do ataque) e *falso-positivos* (de casas que não participaram do ataque).

O modelo proposto está representado na Figura 2. As variáveis latentes (desconhecidas) são representadas por nós brancos e a variável observada $x_i$ é representada por um nó cinza. Os pontos indicam os hiperparâmetros do modelo.

Para medir a credibilidade relativa das hipóteses $H_0$ e $H_1$, usamos a razão de verossimilhança conhecida como fator de Bayes (*Bayes factor*). Definimos o fator de Bayes como $BF_{10}(x_i) \triangleq p(x_i|H_1)/p(x_i|H_0)$. A verossimilhança do modelo (probabilidade marginal) pode ser interpretada como a probabilidade de observar a amostra $x_i$ no tempo $t_i$ dado que a hipótese $H_j$, $j = 0, 1$, é verdadeira.

**Verossimilhança sob a hipótese nula** Sob a hipótese nula $H_0$ (*não ataque*), assumimos que os $N$ roteadores domésticos geram um falso-positivo de modo independente com probabilidade $\theta_F$, onde $\theta_F$ é a taxa de falso-positivos (*False Positive Rate* - FPR) do classificador. O número de positivos $x_i$ dado $H_0$ e $\theta_F$ segue, então, uma distribuição binomial: $x_i|\theta_F, H_0 \sim \text{Binomial}(N, \theta_F)$.

Embora possa ser estimada por meio de experimentos controlados, a FPR real do classificador é desconhecida e pode até variar com o tempo (por exemplo, de-




**(a) Hipótese nula $H_0$ (sem ataque).** **(b) Hipótese alternativa $H_1$ (ataque).**

**Figura 2. Modelo Probabilístico Gráfico.**

vido a mudanças nos padrões de tráfego do usuário doméstico, como relatado em [Streit et al. 2021a, Feldmann et al. 2021]). Assim, em vez de usar uma estimativa pontual (*maximum likelihood estimation*), assumimos que $\theta_F$ é uma variável aleatória com distribuição *a priori* $p(\theta_F|H_0)$. Como $p(x_i|\theta_F, H_0)$ é binomial, adotamos uma distribuição *a priori* conjugada $\theta_F \sim$ Beta$(\alpha_F, \beta_F)$. Os *hiperparâmetros* $\alpha_F$ e $\beta_F$ expressam nosso conhecimento prévio acerca de $\theta_F$. Podemos interpretar esses hiperparâmetros como $\alpha_F$ falso-positivos observados anteriormente dentre $(\alpha_F + \beta_F)$ roteadores domésticos. Temos imediatamente que a distribuição marginal será beta-binomial com parâmetros $N$, $\alpha_F$ e $\beta_F$:

$$p(x_i \mid H_0) = \binom{N}{x_i} \frac{B(x_i + \alpha_F, N - x_i + \beta_F)}{B(\alpha_F, \beta_F)}, \tag{1}$$

onde $B(a, b)$ é a função beta definida como $B(a, b) = \int_0^1 z^{a-1}(1-z)^{b-1}\, dz$.

**Verossimilhança sob a hipótese alternativa** Imaginemos que esteja ocorrendo um ataque DDoS no instante de tempo $t_i$. Dentre os $x_i$ roteadores que reportam positivo, poderá haver tanto verdadeiro-positivos de residências que participam do ataque quanto falso-positivos de residências que não participam. Sendo $b$ o número de residências infectadas, a distribuição do número de falso-positivos e verdadeiro-positivos pode ser modelada separadamente. Seja $D_i$ o número de verdadeiro-positivos (detecções) no tempo $t_i$. Se $b$ casas participam do ataque (de um total de $N$ casas) e cada roteador detecta um ataque independentemente com probabilidade $\theta_D$, teremos $D_i \mid b, \theta_D, H_1 \sim$ Binomial$(b, \theta_D)$, onde $\theta_D$ é a taxa de verdadeiro-positivos do classificador (TPR ou *recall*). Analogamente, dada a FPR $\theta_F$ do classificador, o número de falso-positivos $F_i$ seguirá uma distribuição binomial: $F_i \mid b, \theta_F, H_1 \sim$ Binomial$(N - b, \theta_F)$. Como os parâmetros $\theta_D$ e $\theta_F$ são desconhecidos, assumimos uma distribuição *a priori* beta com parâmetros (respectivamente) $\alpha_D, \beta_D$ e $\alpha_F, \beta_F$. Teremos, portanto, $D_i \mid b, H_1 \sim$ Beta-Binomial$(b, \alpha_D, \beta_D)$ e $F_i \mid b, H_1 \sim$ Beta-Binomial$(N - b, \alpha_F, \beta_F)$.

O número total de positivos $x_i$ será a soma dos verdadeiro-positivos $D_i$ e falso-

positivos $F_i$ reportados pelos $N$ roteadores no momento $t_i$. O Modelo Probabilístico Gráfico representado pela Figura 2b indica que $F_i$ e $D_i$ são condicionalmente independentes dado $b$. Portanto, a PMF da soma de $F_i$ e $D_i$ será igual à convolução discreta de $p(F_i \mid b, H_1)$ e $p(D_i \mid b, H_1)$. Podemos então calcular a verossimilhança do modelo sob a hipótese alternativa $p(x_i \mid H_1)$ a partir de $p(x_i \mid b, H_1)$ e $p(b \mid H_1)$. Se cada uma das $N$ residências participa do ataque com probabilidade (desconhecida) $\theta_b$, assumindo uma distribuição *a priori* $\theta_b \sim \text{Beta}(\alpha_b, \beta_b)$ temos que o número total de atacantes $b$ é Beta-Binomial$(N, \alpha_b, \beta_b)$. Concluímos que a verossimilhança sob a hipótese alternativa $H_1$ é

$$p(x_i|H_1)=\sum_{b=0}^{N}\left[\binom{N}{b}\frac{B(b+\alpha_b,N-b+\beta_b)}{B(\alpha_b,\beta_b)}\sum_{k=\gamma_i}^{\delta_i}\left[\binom{N-b}{x_i-k}\frac{B(x_i-k+\alpha_F,N-b-(x_i-k)+\beta_F)}{B(\alpha_F,\beta_F)}\binom{b}{k}\frac{B(k+\alpha_D,b-k+\beta_D)}{B(\alpha_D,\beta_D)}\right]\right], \tag{2}$$

onde $\gamma_i \triangleq \max\{0, x_i - (N-b)\}$ e $\delta_i \triangleq \min\{b, x_i\}$[1].

**Fator de Bayes** Depois de definir a verossimilhança do modelo sob ambas as hipóteses em (1) e (2), podemos calcular o fator de Bayes correspondente $BF_{10}$:

$$\begin{aligned} BF_{10}(x_i, \alpha_b, \beta_b, \alpha_F, \beta_F, \alpha_D, \beta_D) &\triangleq \frac{p(x_i \mid H_1)}{p(x_i \mid H_0)} \\ &= \frac{\sum_{b=0}^{N}\left[\binom{N}{b}B(b+\alpha_b,N-b+\beta_b)\cdot\sum_{k=\gamma_i}^{\delta_i}\binom{N-b}{x_i-k}B(x_i-k+\alpha_F,N-b-(x_i-k)+\beta_F)\binom{b}{k}B(k+\alpha_D,b-k+\beta_D)\right]}{\binom{N}{x_i}B(x_i+\alpha_F, N-m+\beta_F)B(\alpha_b,\beta_b)B(\alpha_D,\beta_D)} \end{aligned} \tag{3}$$

Após observar $x_i$ resultados positivos em $N$ casas em um determinado período de tempo $t_i$, podemos empregar diretamente a Equação 3 para avaliar qual hipótese a evidência favorece. Caso haja evidências a favor de um ataque, o operador de rede pode tomar as medidas apropriadas dependendo do peso da evidência (por exemplo, anedótica, substancial, decisiva). Destacamos que uma implementação direta da Equação 3 permite calcular o valor exato do fator de Bayes com complexidade de pior caso $O(N^2)$, onde $N$ é o número de residências. Em geral, modelos bayesianos não triviais só admitem soluções aproximadas obtidas através de simulações (MCMC) ou cálculo numérico, não sendo esse o nosso caso. Lembramos ainda que esse cálculo, apesar de simples, não é feito no roteador doméstico, mas por um servidor do ISP que agrega os resultados binários (ataque / não ataque) de cada roteador.

## 4. Resultados

Apresentamos nesta seção uma avaliação de nossa abordagem utilizando dois conjuntos de dados distintos. Na Seção 4.1, mostramos os resultados para o *dataset* rotulado com 4.870 residências. Em seguida, apresentamos na Seção 4.2 os resultados para o experimento de campo com 10 voluntários realizando ataques DDoS reais.

Além das métricas de FPR (taxa de falso-positivos) e TPR (taxa de verdadeiro-positivos) definidas na literatura, empregamos também as métricas de Probabilidade de Detecção de Ataques (PDA) e Tempo de Detecção (TDD). Como o tráfego de ataque pode se estender por diversas janelas deslizantes consecutivas, ainda que o sistema falhe

[1]Para $k < \gamma_i$ e $k > \delta_i$, o produto $p(F_i = x_i - k \mid b, H_1) \cdot p(D_i = k \mid b, H_1)$ é zero.

em detectar um ataque numa determinada janela, ele pode ter sucesso em uma janela posterior. Assim, calculamos a PDA como a razão entre o número de ataques para os quais o modelo gerou um resultado verdadeiro-positivo em pelo menos uma das janelas que contém tráfego de ataque e o número total de ataques. Definimos o TDD de um ataque detectado pelo sistema como o número de janelas deslizantes necessárias até que seja gerado um resultado verdadeiro-positivo. Dada nossa taxa de amostragem, ataques detectados já na primeira janela possuem um TDD de um minuto.

### 4.1. Detecção de ataques usando *dataset* rotulado

Visando avaliar de maneira distinta a primeira e a segunda camadas do sistema proposto, separamos nosso *dataset* rotulado em duas partes. Usamos os primeiros 12 dias do *dataset* para avaliar a camada de detecção local e os 8 dias restantes para avaliar a camada de correlação espaço-temporal. No primeiro conjunto, $5\%$ das residências estão infectadas (com base em [Auchard 2016]), enquanto que no segundo conjunto reduzimos essa fração para $1\%$ com o intuito de fazer uma avaliação mais rigorosa de nosso modelo, já que uma proporção menor de residências infectadas torna o processo de correlação mais difícil.

Nosso classificador *Random Forest* apresenta uma TPR de $0,9524$ no conjunto de treinamento, com resultados semelhantes quando avaliamos cada vetor de ataque separadamente. Quando consideramos os resultados de múltiplas janelas contendo tráfego de um mesmo ataque, temos uma PDA igual a $0,9951$. Dentre os ataques detectados com sucesso, $86,90\%$ tiveram TDD de um minuto, enquanto $99,06\%$ apresentaram um TDD de 2 minutos ou menos. Além de detectar com sucesso mais de $99\%$ de todos os ataques, nosso classificador produziu um número reduzido de alarmes falsos. A FPR no conjunto de teste foi igual a $7,5 \cdot 10^{-5}$, equivalente a uma média de $0,11$ alarmes falsos por residência por dia.

Nossos resultados se tornam ainda melhores quando acrescentamos a camada de correlação espaço-temporal. Primeiro, treinamos o classificador usando os primeiros 12 dias do *dataset* e o aplicamos aos 8 dias restantes. Então, calculamos a variável de interesse $x_i$, definida como o número de residências que reportaram positivo para a janela deslizante relativa ao tempo $t_i$. A partir de $x_i$, o fator de Bayes (Equação 3) nos permite avaliar a evidência a favor da hipótese nula (sem ataque) ou a favor da hipótese alternativa (com ataque). Deve-se notar que não há como conhecer o número exato de residências infectadas $b$ e que é difícil estimar com precisão as taxas FPR ($\theta_F$) e TPR ($\theta_D$) do classificador, que dependem do tráfego residencial. Diferentemente de [Mendonça et al. 2019a], nosso modelo não usa um valor fixo para esses parâmetros, permitindo que essa incerteza seja capturada através de distribuições *a priori*. Seguimos, então, as recomendações de [Kruschke 2015, Chap.10] para a escolha dos hiperparâmetros. Como usamos distribuições *a priori* beta, nosso grau de confiança é representado pela soma dos hiperparâmetros $\alpha$ e $\beta$ [2].

Tendo como referência o ataque reportado em [Auchard 2016], definimos $\alpha_b = 1.5$ e $\beta_b = 10.5$, de modo que o valor mais provável *a priori* (moda) é $0,05$. Lembramos que a proporção real de infectados nesse conjunto de dados ($1\%$) é cinco vezes menor do que a moda. Assim, podemos fazer uma avaliação conservadora do nosso modelo, já

[2] Para uma discussão mais detalhada sobre a interpretação dos parâmetros da distribuição beta, ver [Kruschke 2015, Chap.6].

que a detecção se torna mais difícil com menos residências infectadas. Para o ajuste dos parâmetros das distribuições *a priori* de $\theta_F$ e $\theta_D$, nos baseamos no resultado do classificador nos primeiros 12 dias, definindo $\alpha_F = 1.75$, $\beta_F = 10000.25$ (moda igual a $7.5 \cdot 10^{-5}$) e $\alpha_D = 10.5$, $\beta_D = 1.5$ (moda igual a $0.95$). Essa escolha de hiperparâmetros, com $(\alpha_F + \beta_F) \gg (\alpha_D + \beta_D)$, reflete nosso conhecimento prévio de que a FPR real do classificador é mais fácil de se estimar do que sua TPR real, já que, no mundo real, amostras de tráfego legítimo são muito mais abundantes do que amostras contendo tráfego de ataques DDoS.

Calculando o fator de Bayes para as janelas deslizantes contendo tráfego de ataque, nosso modelo detectou *evidência decisiva* a favor de $H_1$ ($BF_{10} \geq 100$ [Wetzels et al. 2011]) em todos os casos. Portanto, temos uma PDA igual a 1 com Tempo de Detecção (TDD) de até um minuto para todos os ataques. Esse resultado sugere que nosso modelo bayesiano é robusto, detectando ataques mesmo quando o número real de atacantes no *dataset* é consideravelmente menor do que o valor esperado da variável latente $b$ dada sua distribuição *a priori*. Além disso, o fator de Bayes das janelas sem tráfego de ataque indicou pelo menos *evidência substancial* a favor de $H_0$ ($BF_{10} \leq 1/3$ [Wetzels et al. 2011]) em todos os casos.

Deste modo, ficam evidentes os benefícios de se considerar uma segunda camada de detecção de ataques na borda da rede do ISP: todos os ataques no *dataset* foram detectados em até um minuto sem que nenhum alarme falso fosse gerado. Para isso, nosso modele exige uma quantidade mínima de informação prévia representada de modo intuitivo na escolha dos hiperparâmetros, mesmo quando a moda das distribuições *a priori* não está tão próxima do valor real. É importante observar que, à medida que obtemos mais informações sobre o desempenho do classificador, podemos facilmente ajustar os hiperparâmetros do modelo para traduzir a redução de nossa incerteza acerca do valor real das variáveis latentes. Isso é possível porque adotamos distribuições *a priori* beta, cujos parâmetros possuem interpretação trivial nesse contexto.

### 4.2. Detecção de ataques em campo usando voluntários

Analisamos agora o desempenho de nosso sistema com dados de ataques DDoS em campo. Recrutamos um grupo de dez voluntários que receberam um dispositivo *Raspberry Pi* "infectado" rodando código-fonte *Mirai* e *BASHLITE* modificado. Ao remover algumas funções do software, garantimos que o malware modificado não pudesse infectar outros dispositivos na rede. Os *Raspberri Pi* foram programados para iniciar ataques DDoS em tempos pseudo-aleatórios predeterminados tendo como alvo um servidor dentro do ISP. Os intervalos entre os ataques seguiram uma distribuição exponencial, com amostras inferiores a 5 minutos descartadas, evitando impactar o uso das redes domésticas participantes. Os voluntários não sabiam quando ocorreria cada ataque, permitindo que continuassem seguindo sua rotina diária sem serem influenciados. Durante o experimento, alternamos entre os vetores de ataque (*UDP flood*, *TCP SYN flood* e *TCP ACK flood*) e as implementações (*Mirai* e *BASHLITE*).

Os voluntários foram monitorados durante um período de 31 dias, de 9 de agosto a 9 de setembro de 2021. No total, realizamos 111 ataques DDoS (média de $3,6$ ataques por dia). As amostras dos contadores de *bytes* e pacotes obtidas durante o experimento indicam que: (1) alguns ataques geraram taxas de *upload* em bps e/ou pps semelhantes

ao tráfego residencial legítimo; e (2) a distribuição das taxas de *download* é idêntica em períodos com e sem ataques, o que é esperado.

Para obter um *dataset* contendo tráfego de residências com e sem dispositivos infectados, juntamos o tráfego dos voluntários com o tráfego de 190 residências selecionadas aleatoriamente dentre as 4.870 residências do *dataset* rotulado. Assim, podemos emular um cenário em que temos medições de centenas ou milhares de residências, mas apenas uma pequena fração ($5\%$) contém um dispositivo infectado e participa de ataques DDoS esporádicos. Até onde sabemos, não há na literatura um *dataset* com características tão realísticas quanto as que elaboramos neste trabalho, contendo tráfego real de usuários domésticos combinado com ataques DDoS baseados em *malwares* reais.

Começamos a avaliação de nosso sistema treinando o classificador *Random Forest* usando todo o *dataset* rotulado. Em seguida, avaliamos o classificador usando o *dataset* contendo 31 dias de tráfego de $N = 200$ residências, dentre as quais $b = 10$ participaram dos ataques DDoS. A probabilidade de detecção de ataque (PDA) considerando apenas a primeira camada de detecção é de $\approx 0,97$ (contra $\approx 0,995$ no *dataset* rotulado). Dentre os ataques detectados com sucesso, $73\%$ foram detectados em até 2 minutos. A FPR foi aproximadamente dez vezes maior do que a observada no *dataset* rotulado: $6,8 \cdot 10^{-4}$, o que equivale a uma média de $0,98$ alarmes falsos por residência por dia.

Para o cálculo do fator de Bayes da camada de correlação espaço-temporal, ajustamos os hiperparâmetros do modelo assim como na Seção 4.1, usando como base o desempenho do classificador no conjunto de teste do *dataset* rotulado (TPR $= 0.95$ e FPR $= 7.5 \cdot 10^{-5}$). Como o número $N$ de residências é menor, reduzimos as somas $(\alpha + \beta)$ de acordo, definindo $\alpha_b = 1.025$, $\beta_b = 1.475$, $\alpha_F = 1.0375$, $\beta_F = 500.9625$, $\alpha_D = 1.475$ e $\beta_D = 1.025$.

Notamos aqui que os *datasets* possuem características bem distintas que levaram a uma piora na FPR e na TPR do classificador. Todavia, mesmo ajustando nossos hiperparâmetros de acordo com o desempenho no *dataset* rotulado, todos os ataques em campo foram detectados com sucesso (PDA igual a 1), com *evidência forte* a favor de $H_1$ ($BF_{10} \geq 10$ [Wetzels et al. 2011]) em pelo menos uma janela de cada ataque. Cerca de $85\%$ dos ataques foram detectados em até 2 minutos, com um TDD máximo de 4 minutos. Não houve alarmes falsos com *evidência forte* a favor de $H_1$ durante o período de 31 dias, e apenas uma janela deslizante levou a um alarme falso com *evidência substancial* de ataque (($BF_{10} \geq 3$ [Wetzels et al. 2011]). No geral, $99,56\%$ das amostras "normais" forneceram evidências a favor da hipótese $H_0$ (sem ataque) segundo nosso modelo. Para comparação, o modelo de máximo *a posteriori* proposto em [Mendonça et al. 2019a] gerou 288 alarmes falsos com os parâmetros ajustados do mesmo modo.

Quando avaliado com dados de ataques DDoS em campo, nosso sistema de detecção de duas camadas foi capaz de: (1) detectar 97% de todos os ataques nos roteadores domésticos (sem fazer uso da correlação espaço-temporal); (2) detectar todos os ataques quando a camada de correlação espaço-temporal é considerada; (3) fornecer um Tempo de Detecção razoavelmente pequeno (até 2 minutos na maioria dos casos); (4) gerar apenas um alarme falso com *evidência substancial* num período de 31 dias.

## 5. Conclusão

A evolução e prevalência dos ataques DDoS baseados em *botnets* apresentam novos desafios para sua detecção e mitigação. A maioria dos métodos propostos anteriormente depende de dados sensíveis extraídos de cabeçalhos de pacotes (endereços IP de origem/destino, protocolos, portas) para a detecção de ataques. Além disso, os resultados frequentemente se limitam a experimentos usando um número bem limitado de dispositivos IoT atuando como *bots*. Neste artigo, estendemos [Mendonça et al. 2019a], apresentando um sistema de duas camadas para detecção de ataques DDoS que depende apenas de dados de contadores de *bytes* e pacotes coletados em roteadores residenciais. A primeira camada de detecção usa um classificador simples, instalado em roteadores domésticos, enquanto a segunda camada, instalada em um servidor do ISP, usa um modelo hierárquico bayesiano para correlacionar as indicações de ataque emitidas por roteadores residenciais do provedor.

Em parceria com um ISP, obtivemos dados de 20 dias de tráfego real de *upload* e *download* de 4.870 residências em 14 cidades diferentes. Para treinar e avaliar nossos modelos, combinamos esses dados com *traces* de ataques gerados com código-fonte *Mirai* e *BASHLITE*. Além disso, conduzimos um experimento de campo com um grupo de 10 voluntários durante um período de 31 dias usando um dispositivo Raspberry Pi conectado por WiFi ao roteador doméstico que executa ataques DDoS com *malwares* reais. Não temos conhecimento de trabalhos anteriores avaliando o desempenho de um detector DDoS em um cenário como esse.

Nossos resultados mostram que o tráfego malicioso pode ser detectado em roteadores domésticos em até dois minutos com alta probabilidade, com baixas taxas de falsos alarmes. Usando apenas estatísticas de contadores de *bytes* e pacotes coletadas a cada minuto, nossa primeira camada de detecção foi capaz de detectar $97\%$ dos ataques reais em nosso experimento, com uma taxa de falso-positivos (FPR) igual a $6,8 \cdot 10^{-4}$. Considerando também a camada de correlação espaço-temporal, nosso método foi capaz de detectar todos os ataques gerando apenas um alarme falso num período de 31 dias. Desta forma, mostramos que é viável detectar ataques DDoS com alta eficiência, e sem depender de nenhuma informação de cabeçalhos ou conteúdos de pacotes, impraticáveis de se coletar quando o tráfego é criptografado e/ou resultante de uma VPN. Não sabemos de nenhuma outra experimentação que tenha sido realizada em um ambiente tão realista como o nosso, com tráfego de milhares de residências e ataques com código real para testar um modelo de detecção.

## Agradecimentos

Este trabalho é parcialmente suportado por projeto de cooperação MCTIC-RNP/NSF e projetos do CNPq e FAPERJ, além de bolsas CAPES.

## Referências

[Akamai 2016] Akamai (2016). Q3 2016 state of the Internet - security report. Technical report, Akamai.

[Anthi et al. 2019] Anthi, E., Williams, L., Słowińska, M., Theodorakopoulos, G., and Burnap, P. (2019). A supervised intrusion detection system for smart home iot devices. *IEEE Internet of Things Journal*, 6(5):9042–9053.

[Antonakakis et al. 2017] Antonakakis, M., April, T., Bailey, M., Bernhard, M., Bursztein, E., Cochran, J., Durumeric, Z., Halderman, J. A., Invernizzi, L., Kallitsis, M., et al. (2017). Understanding the Mirai botnet. In *USENIX Security Symposium*, pages 1092–1110.

[Auchard 2016] Auchard, E. (2016). German Internet outage was failed botnet attempt: report. https://www.reuters.com/article/us-deutsche-telekom-outages-idUSKBN13N12K. Accessed: 2021-07-02.

[Bihary 2017] Bihary, C. (2017). How to Monitor Encrypted Traffic and Keep Your Network Secure. https://www.garlandtechnology.com/blog/how-to-monitor-encrypted-traffic-and-keep-your-network-secure. Accessed: 2022-02-24.

[DARKReading 2020] DARKReading (2020). DDoS Attacks Spiked, Became More Complex in 2020. https://www.darkreading.com/attacks-breaches/ddos-attacks-spiked-became-more-complex-in-2020/d/d-id/1339814. Accessed: 2021-07-02.

[Doshi et al. 2018] Doshi, R., Apthorpe, N., and Feamster, N. (2018). Machine Learning DDoS Detection for Consumer Internet of Things Devices. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 29–35.

[Feldmann et al. 2021] Feldmann, A., Gasser, O., Lichtblau, F., Pujol, E., Poese, I., Dietzel, C., Wagner, D., Wichtlhuber, M., Tapiador, J., Vallina-Rodriguez, N., Hohlfeld, O., and Smaragdakis, G. (2021). A year in lockdown: How the waves of covid-19 impact internet traffic. *Commun. ACM*, 64(7):101–108.

[Jia et al. 2020] Jia, Y., Zhong, F., Alrawais, A., Gong, B., and Cheng, X. (2020). FlowGuard: An intelligent edge defense mechanism against IoT DDoS attacks. *IEEE Internet of Things Journal*, 7(10):9552–9562.

[Kaspersky 2021] Kaspersky (2021). IT threat evolution Q1 2021. https://securelist.com/it-threat-evolution-q1-2021-non-mobile-statistics/102425/. Accessed: 2021-07-02.

[Kolias et al. 2017] Kolias, C., Kambourakis, G., Stavrou, A., and Voas, J. (2017). DDoS in the IoT: Mirai and other botnets. *Computer*, 50(7):80–84.

[Kruschke 2015] Kruschke, J. K. (2015). *Doing Bayesian data analysis: A tutorial with R, JAGS, and Stan (2nd)*. San Diego, CA: Academic Press.

[Liaskos et al. 2016] Liaskos, C., Kotronis, V., and Dimitropoulos, X. (2016). A novel framework for modeling and mitigating distributed link flooding attacks. In *INFOCOM 2016*, pages 1–9. IEEE.

[Marín et al. 2021] Marín, G., Casas, P., and Capdehourat, G. (2021). Deepmal-deep learning models for malware traffic detection and classification. In *Data Science – Analytics and Applications*, pages 105–112. Springer Fachmedien Wiesbaden.

[McDermott et al. 2018] McDermott, C. D., Majdani, F., and Petrovski, A. (2018). Botnet detection in the Internet of things using deep learning approaches. In *2018 international joint conference on neural networks (IJCNN)*, pages 1–8.

[Mehdi et al. 2011] Mehdi, S. A., Khalid, J., and Khayam, S. A. (2011). Revisiting traffic anomaly detection using software defined networking. In *International workshop on recent advances in intrusion detection*, pages 161–180. Springer.

[Meidan et al. 2018] Meidan, Y., Bohadana, M., Mathov, Y., Mirsky, Y., Shabtai, A., Breitenbacher, D., and Elovici, Y. (2018). N-BaIoT–Network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders. *IEEE Pervasive Computing*, 17(3):12–22.

[Mendonça et al. 2019a] Mendonça, G., Santos, G. H., de Souza e Silva, E., Leão, R. M., Menasché, D. S., and Towsley, D. (2019a). An extremely lightweight approach for

ddos detection at home gateways. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 5012–5021. IEEE.

[Mendonça et al. 2019b] Mendonça, G., Santos, G. H., de Souza e Silva, E., Leão, R. M. M., Menasche, D. S., et al. (2019b). Uma abordagem para detecção de ddos a partir de roteadores domésticos. In *Anais do XXXVII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 834–847. SBC.

[NETSCOUT 2021] NETSCOUT (2021). 2H 2020 Threat Intelligence Report – DDoS in a Time of Pandemic. https://www.netscout.com/threatreport/. Accessed: 2021-07-02.

[Nevat et al. 2018] Nevat, I., Divakaran, D. M., Nagarajan, S. G., Zhang, P., Su, L., Ko, L. L., and Thing, V. L. (2018). Anomaly detection and attribution in networks with temporally correlated traffic. *IEEE/ACM Transactions on Networking*, 26(1):131–144.

[Salman et al. 2019] Salman, O., Elhajj, I. H., Chehab, A., and Kayssi, A. (2019). A machine learning based framework for iot device identification and abnormal traffic detection. *Transactions on Emerging Telecommunications Technologies*, page e3743.

[Sedjelmaci et al. 2017] Sedjelmaci, H., Senouci, S. M., and Taleb, T. (2017). An accurate security game for low-resource iot devices. *IEEE Transactions on Vehicular Technology*, 66(10):9381–9393.

[Silveira et al. 2011] Silveira, F., Diot, C., Taft, N., and Govindan, R. (2011). Astute: Detecting a different class of traffic anomalies. *ACM SIGCOMM CCR*, 41(4):267–278.

[Streit et al. 2021a] Streit, A., Ribeiro, M. C., Leão, R. M., de Souza e Silva, E., et al. (2021a). Efeito do confinamento causado pela pandemia covid-19 nos perfis de tráfego residencial. In *Anais do XXXIX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 238–251. SBC.

[Streit et al. 2021b] Streit, A., Santos, G. H., Leão, R. M., de Souza e Silva, E., Menasché, D., and Towsley, D. (2021b). Network anomaly detection based on tensor decomposition. *Computer Networks*, 200:108503.

[Sudheera et al. 2021] Sudheera, K. L. K., Divakaran, D. M., Singh, R. P., and Gurusamy, M. (2021). Adept: Detection and identification of correlated attack stages in iot networks. *IEEE Internet of Things Journal*, 8(8):6591–6607.

[Summerville et al. 2015] Summerville, D. H., Zach, K. M., and Chen, Y. (2015). Ultra-lightweight deep packet anomaly detection for Internet of things devices. In *Performance Computing and Communications Conference*, pages 1–8. IEEE.

[Wan et al. 2020] Wan, Y., Xu, K., Xue, G., and Wang, F. (2020). Iotargos: A multi-layer security monitoring system for internet-of-things in smart homes. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*, pages 874–883. IEEE.

[Wang et al. 2018] Wang, A., Chang, W., Chen, S., and Mohaisen, A. (2018). Delving into internet ddos attacks by botnets: characterization and analysis. *IEEE/ACM Transactions on Networking*, 26(6):2843–2855.

[Wetzels et al. 2011] Wetzels, R., Matzke, D., Lee, M. D., Rouder, J. N., Iverson, G. J., and Wagenmakers, E.-J. (2011). Statistical evidence in experimental psychology: An empirical comparison using 855 t tests. *Perspectives on Psychological Science*, 6(3):291–298.

[Yoachimik 2021] Yoachimik, O. (2021). Cloudflare thwarts 17.2M rps DDoS attack — the largest ever reported. https://blog.cloudflare.com/cloudflare-thwarts-17-2m-rps-ddos-attack-the-largest-ever-reported/. Accessed: 2021-08-20.