IoT-Flows: Lightweight Policy Enforcement of Information Flows in IoT Infrastructures

EQUIPE

Coordenador no Brasil:
José Augusto Suruagy - Universidade Federal de Pernambuco (UFPE)

Coordenador nos EUA
Darko Marinov (University of Illinois)

PARCEIROS
Universidade Federal de Pernambuco (UFPE)
University of Illinois Urbana-Champaign
University of Michigan

SITE
https://iot-flows.cin.ufpe.br/

CONTATO
iot-flows@cin.ufpe.br

DESCRIÇÃO

Introdução

O número de dispositivos no mercado da Internet das Coisas (IoT) é de aproximadamente 7 bilhões atualmente e chegará a 25 bilhões em 2021 [Gartnet]. Prover soluções de segurança eficientes nesses dispositivos é um desafio devido ao fato de seus recursos computacionais disponíveis serem limitados. A grande quantidade de dispositivos IoT disponíveis e suas fraquezas em termos de segurança, os tornam alvos atrativos para uso em ataques de DDoS (*Distributed Deny of Service*) na Internet. Como exemplo, em 2016, a Internet sofreu o maior ataque de DDoS já registrado, chegando a um volume de tráfego da ordem de 1,3 Tbps. Esse ataque utilizou uma Botnet, a Mirai, formada por dispositivos IoT conectados como câmeras de segurança, DVRs e *gateways* residenciais, afetando grande parte da costa leste dos Estados Unidos e gerando prejuízos financeiros da ordem de bilhões de dólares. Desde então, a garantia da segurança de dispositivos IoT se tornou crítica.

Embora diversas soluções para garantir a segurança de dispositivos IoT tenham sido apresentadas nos últimos anos, a maioria delas sofre de duas limitações principais: (i) atuam apenas em uma camada de rede e (ii) não fornecem suporte nativo que permita extensão para o tratamento de novos ataques. Portanto, existe um desafio envolvendo a flexibilidade e integração das soluções propostas para mitigar as ameaças no contexto IoT.

Proposta e Objetivos

Este projeto visa melhorar a segurança de redes IoT e contribuir para a mitigação de ataques DDoS realizados por dispositivos pertencentes a essas redes. Para isso, o objetivo central é a proposição e o desenvolvimento de uma plataforma capaz de monitorar e controlar a comunicação dos dispositivos na rede IoT a fim de 1) detectar ataques e comportamentos anômalos; 2) atualizar dinamicamente e em tempo real, no hub IoT da rede, políticas de comunicação; 3) bloquear dispositivos com comportamento anômalo (e.g. impedido a comunicação destes com a Internet e demais dispositivos da rede IoT); 4) bloquear ataques detectados e; 5) alertar o administrador da rede ou usuário autorizado sobre ocorrências na rede IoT.

Detecção de ataques de origem externa: por estarem conectados à Internet, dispositivos IoT também são alvos de ataques. Exemplos incluem os de negação de serviço (DoS) para a exaustão de recursos (e.g., memória, processamento, banda passante) e invasão tanto para uso do dispositivo em botnets quanto para violação da privacidade dos usuários da rede. Portanto, é de vital importância que o sistema proposto seja capaz de detectar ataques de origem interna e externa à rede para maximizar a segurança.
Arquitetura de *Software* Distribuída: a plataforma utiliza uma arquitetura de *software* distribuída com múltiplos componentes, cada um possuindo uma responsabilidade específica e única. Dessa forma, pode-se ver cada componente como um módulo que trata de um problema diferente, amenizando a complexidade de um sistema que busca garantir a segurança na comunicação na rede IoT.
Abordagem multicamadas: diferentes ataques podem se valer de vulnerabilidades e fraquezas em diferentes camadas da pilha de protocolos. Por exemplo, ataques de *jamming* ocorrem na camada física e consistem em impedir o uso do canal de comunicação pelos dispositivos da rede. Exemplos relacionados à camada de enlace incluem DoS (e.g. ataque ao CSMA/CA em interfaces IEEE 802.11), MAC spoofing, quebra de criptografia com consequente perda de privacidade (e.g. ataque KRACK em redes IEEE 802.11), entre outros. Na camada de rede, datagramas do protocolo IP podem, por exemplo, sofrer ataques de *spoofing* e *replay*. Na camada de transporte, por exemplo, a simplicidade do UDP é explorada em ataques de reflexão/amplificação para geração de DDoS de grande porte. Na camada de aplicação, por exemplo, o CoAP é inerentemente inseguro e precisa contar com outros protocolos (e.g., DTLS, IPSec) para segurança. A plataforma proposta analisa dados de múltiplas camadas para maior abrangência na detecção de ataques.
Arquitetura de Rede Distribuída: a plataforma utiliza monitores distribuídos na rede, capturando tráfego para extração de informações nas diferentes camadas da pilha de protocolos para detecção mais ampla de ataques.
Extensibilidade: o crescente número de dispositivos IoT e com potencial de uso em ataques de grandes proporções, despertam interesse de atacantes no desenvolvimento contínuo de novos tipos de ataques. Por isso, a plataforma proposta também possui como foco permitir sua extensão para a detecção de novos ataques. Isso é feito fornecendo ao administrador ou usuário autorizado a opção de se criar novas regras para detecção de ataques com base na análise do tráfego de rede e por meio de um aplicativo móvel, promovendo ainda a usabilidade e longevidade do sistema.

A Plataforma IoT-Flows

A plataforma IoT-Flows visa garantir a segurança do canal de comunicação entre dispositivos IoT utilizando-se de análise de eventos complexos (CEP) para detectar comportamentos suspeitos na rede. O uso de CEP é feito aplicando regras ao tráfego de rede, detectando comportamentos que mapeiam ataques que tenham como alvo os dispositivos IoT.

O sistema proposto faz uso de conceitos utilizados em sistemas autônomos para atuar de maneira distribuída nas diferentes camadas de rede. A divisão do sistema em componentes, descritos a seguir, garante a atuação de cada componente de forma modular.

Arquitetura

A arquitetura do sistema é baseada no (M)onitor A(nalyze) P(lan) E(xecute)-K(nowledge), arquitetura introduzida pela IBM para sistemas autônomos. A Figura 1 ilustra os componentes do sistema proposto em comparação com os componentes tradicionais descritos no MAPE-K. Descrevemos abaixo de forma breve os diferentes componentes da arquitetura:

Monitores

A responsabilidade central dos monitores é de capturar os pacotes trafegados nas redes sem fio (WiFi) e cabeada, extraindo metadados relevantes para análise do fluxo da rede. Com o objetivo de ter acesso ao máximo de informações sobre o tráfico, o sistema trabalha com dois tipos de monitores: (i) Um monitor baseado na captura de pacotes utilizando interfaces de rede em modo promíscuo, que captura informações da camada física e de enlace e tem arquitetura distribuída e (ii) Um monitor que recebe o tráfego espelhado do roteador da rede IoT e captura informações da camada de enlace, rede, transporte e aplicação. O segundo monitor apresenta relação de 1 para 1 com o roteador da rede IoT. A Figura 2 ilustra a arquitetura dos monitores em relação aos demais componentes.

Pattern API (Knowledge)

Este é um componente vital do sistema, abrigando as diferentes regras relacionadas a detecção dos ataques bem como as ações a serem tomadas uma vez detectado o ataque. Um *pattern* mapeia uma regra para ação. Como exemplo, por meio da API pode-se cadastrar uma regra que bloqueia o tráfego partindo de um dispositivo IoT para endereços externos que não os do fabricante do mesmo. O sistema, por meio da regra detectaria tal comportamento aplicando a regra aos pacotes da rede no analisador CEP (abaixo). Uma vez detectado o comportamento suspeito, o sistema poderia, por exemplo, gerar um alerta ao usuário ou mesmo bloquear o dispositivo comprometido na rede.

CEP Analyzer (Analyzer)

Esse componente tem como responsabilidade receber o tráfego da rede enviado pelos monitores e aplicar regras preconfiguradas utilizando CEP [1] para detecção de violações de políticas de fluxo e de comportamentos suspeitos, i.e., ataques.

Policy Manager (Plan)

Esse componente atua como ponte entre o analisador e os componentes de execução descrito abaixo. Esse componente mapeia um pattern para uma determinada ação. Por exemplo, uma vez detectado um ataque DDoS com o uso do pattern, todo o fluxo de rede envolvendo o dispositivo comprometido pode ser bloqueado por meio de uma ação pré-cadastrada.

Execute components (Execute)

Esses componentes tem como responsabilidade executar a ação pré-determinada para cada *pattern*. Como exemplo, estão roteadores de rede e geradores de alerta, atuando ao sistema detectar um comportamento suspeito.

Resultados Preliminares

A plataforma é atualmente capaz de detectar ataques de rede tradicionais e de diferentes camadas, como SYN Flood, ARP Spoofing e Slowloris, tipos de ataque de negação de serviço (DoS) [2]. Durante o evento, serão apresentadas demonstrações de cenários envolvendo tais ataques utilizando placas de prototipação (Raspberry PIs) simulando dispositivos IoT, roteadores open-source e dispositivos atacantes.

Conclusões

Garantir a segurança de dispositivos IoT contra novas e tradicionais ameaças é crucial. Este projeto visa entregar uma plataforma capaz de melhorar a segurança em redes IoT atuando nas diferentes camadas de rede TCP/IP. A plataforma contrapõe as limitações encontradas nas soluções atuais, i.e., extensibilidade e usabilidade, usando para tal conceitos de sistemas autônomos, processamento de eventos complexos e monitoramento distribuído da rede.

Referências

[1] Cugola G, Margara A. "Processing flows of information: From data streams to complex event processing." ACM Computing Surveys (CSUR), 2012, 44(3): 15.

[2] Bellardo, J, Savage, S. "802.11 Denial-of-Service Attacks: Real Vulnerabilities and Practical Solutions." USENIX Security Symposium (2003).

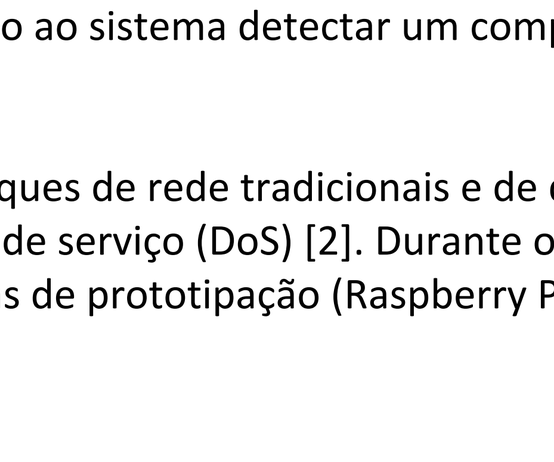

Figura 1 Arquitetura da plataforma IoT-Flows

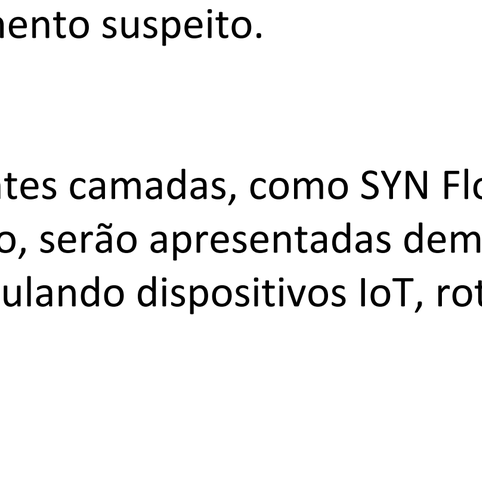

Figura 2 Arquitetura dos monitores