Arranque Seguro de Redes 6LoWPAN para prevenir Ataques Vampiro na Internet das Coisas

Tiago Diogo e Miguel Pardal

Instituto Superior Técnico, Av. Rovisco Pais, 1, 1049-001 Lisboa, Portugal,
{tiago.diogo,miguel.pardal}@tecnico.ulisboa.pt

1 Introdução

A Internet das Coisas pode ser vista como uma teia de dispositivos interligados entre si que vão desde vestuário inteligente (*wearables*) até redes de sensores de gama empresarial. Apesar da enorme variedade e diferenças entre estes dispositivos, algo que todos têm em comum é a sua limitação de recursos. Dada esta variedade de ambientes, uma brecha na segurança destas redes pode implicar uma fuga de informação confidencial ou prover informações sobre as escolhas e paradeiro de um largo número de indivíduos constituindo uma violação de privacidade [1]. Nessa medida um estudo extensivo de ataques dirigidos a dispositivos restritos de recursos foi conduzido na secção 3 e uma estratégia comum de mitigação apresentada na secção 4. Esta estratégia apresenta garantias de segurança em troca de um aumento da complexidade da infraestrutura de apoio, o que por sua vez levou à proposta de uma estação de gestão da rede. A secção **??** dá uma visão mais detalhada destes dois sistemas. Para definir com precisão o tipo e quantidade de recursos necessários ao funcionamento de uma rede com estas garantias de segurança, conduzimos experiências de ocupação de espaço e consumo energético em dispositivos físicos, bem como testes de usabilidade ao nosso sistema. Os resultados estão expostos na secção 5. Por fim a secção 6 apresenta as nossas conclusões e oportunidades para trabalho futuro.

2 Visão Geral da Rede

Neste tipo de arquiteturas, os nós sensores ou atuadores pertencem a uma rede com recursos muito restritos que usa protocolos desenhados para essas necessidades e mecanismos de compressão de cabeçalhos para reduzir o tamanho dos pacotes, necessitando assim de um encaminhador de fronteira (*border router*) de forma a comunicar com redes exteriores. Este tipo de arquiteturas pode ser usado, por exemplo, em sistemas de intrusão domésticos ou em sistemas de monitorização fabril. Se um atacante conseguisse desativar estes sistemas, poderia causar um encerramento da fábrica devido à falta de controlo sobre as condições de trabalho. Estas são preocupações reais apoiadas por uma gama de ataques que se foca em desativar redes Internet das Coisas (IdC) por colocar os nós *offline*. Tais ataques são de seguida analisados e documentados.

3 Análise de Ataques

No entanto, dadas as características deste dispositivos existe uma grupo de ataques ao nível da camada de rede com especial interesse e importância: os ataques de esgotamento de energia, também conhecidos como ataques "vampiro". Estes, focam-se em drenar as baterias – a "vida" – dos dispositivos, trabalhando ao longo do tempo para desativar completamente a rede. Alguns destes ataques focam-se em implementações específicas enquanto outros são agnósticos ao protocolo utilizado [2][3].

Embora seja verdade que existem estratégias de mitigação para estes ataques [2], elas implicam verificações e validações adicionais em cada nó e para cada pacote. Por exemplo, os nós poderiam verificar os cabeçalhos dos pacotes para detetar ciclos no trajeto da mensagem e invalidar o pacote. Ou caso vários pacotes repetidos cheguem ao destino, poder-se ia analisar o caminho feito por cada pacote e o ponto de convergência no trajeto indicaria o nó que enviou múltiplos pacotes revelando assim o atacante. Infelizmente estas estratégias de mitigação colocam um fardo pesado sobre estes nós com recursos muito limitados. Para cada ataque adicional que quiséssemos mitigar, mais validações teriam de ser empregues, chegando-se a um ponto em que os recursos despendidos em validações poder-se-iam equiparar a um ataque aos recursos desse elemento. Para resolver este problema, propomos que um arranque (*bootstrapping*) seguro seja efetuado para cada novo dispositivo da rede. A secção seguinte detalha o que é um processo de *bootstrapping*, como é que ele é efetuado e algumas soluções existentes que seguem esta abordagem.

4 *Bootstrapping* Seguro

No nosso trabalho propomos um sistema de *bootstrapping* seguro onde:

- Não é necessário *hardware* adicional durante a instalação no terreno;
- Não é necessário obter credenciais adicionais após instalação no terreno;
- Não é necessário recorrer a terceiros para gerar ou instalar credenciais;
- Todas as mensagens enviadas na rede são cifradas e autenticadas desde o primeiro momento

Chamámos à nossa solução AutoStrap porque se destina a permitir um processo de *bootstrapping* seguro e eficiente, que tenha o mínimo de interação necessária com os operadores do sistema e não necessite de conhecimento interno do sistema para ser usado.

4.1 AutoStrap

O principio de funcionamento do AutoStrap é a adição de um novo componente na arquitetura da infraestrutura – o *bootstrapper* – que é responsável pela escrita para o dispositivo de todos os identificadores e credenciais de segurança

necessários para uma operação que responda aos objetivos e requisitos do sistema, sem necessidade de obter credenciais após o inicio da fase de operação ou de confiar em credenciais criadas por terceiros. As credenciais utilizadas são um identificador único e uma chave de 128 bits usada pelo protocolo Advanced Encryption Standard (AES) [4] no modo Counter with CBC-MAC (CCM) [5]. O uso desta chave e protocolo permite que os pacotes tenham o seu conteúdo cifrado e o seu cabeçalho autenticado com um Message Integrity Code (MIC) também gerado a partir dessa mesma chave. Isto assegura que todos os pacotes que se propagam na rede são confidenciais, íntegros e autênticos. Desta forma, os "vampiros" não serão capazes de se introduzir na topologia, frustrando assim as suas tentativas de conduzir ataques de esgotamento de bateria. Este processo é iniciado por um operador do sistema, com a ligação do dispositivo ao *bootstrapper*, e após selecionar o tipo de *hardware* numa interface gráfica embutida, todo o restante processo é realizado automaticamente em segundo plano tal como apresentado no diagrama de sequência presente na Figura 1.

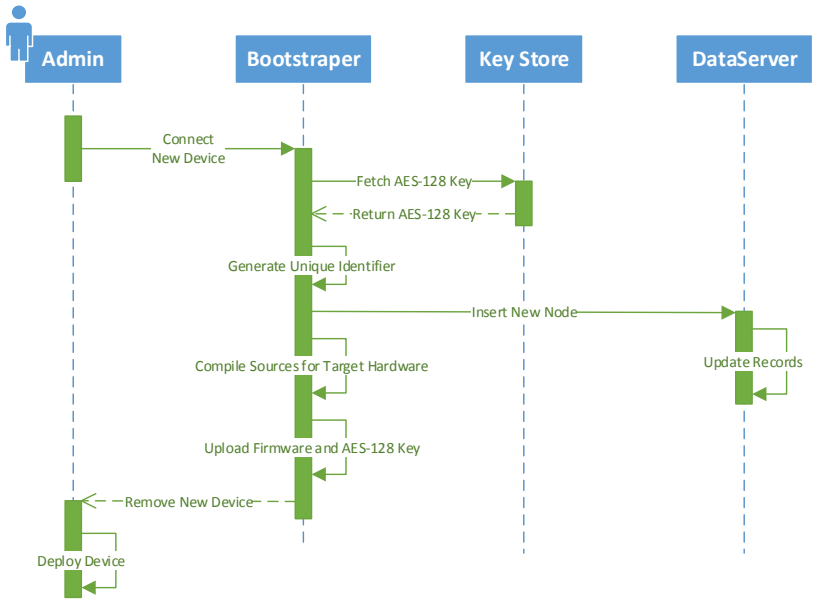

Figura 1: Processo de *Bootstrapping*

4.2 Arquitetura do Sistema

A inovação contida neste proposta advém dos outros dois componentes do sistema, o *CoAP Client Observer*, e o *Bootstrapper* que faz por sua vez uso de um *Key Store*. O *Key Store* é responsável por guardar a chave de rede utilizada no processo de *bootstrapping*. O *Bootstrapper* é responsável por conduzir o processo AutoStrap mapeando o novo dispositivo no sistema e fornecendo-lhe as credenciais de segurança necessárias. Por fim o *CoAP Client Observer* atua como o único cliente da rede. Em vez de termos utilizadores a requisitar novas leituras diretamente aos sensores da rede com recursos limitados, este cliente regista-se junto dos dispositivos existentes e é notificado de cada novo valor ou evento despoletado pelos dispositivos.

5 Avaliação

Primeiro, medimos e documentamos os recursos físicos necessários para suportar os protocolos e estratégias de mitigação escolhidas de forma a entender se são adequados ao *hardware* usado na IdC. Segundo, conduzimos testes de usabilidade ao nosso sistema de gestão de redes para analisar a sua eficiência e facilidade de uso.

5.1 Aplicabilidade ao *Hardware*

Analisando os resultados podemos verificar que a introdução destes mecanismos comporta um aumento de 3.02% na utilização de memória *flash* e 1.02% na utilização de memória Random-Access Memory (RAM), permitindo-nos concluir que apenas uma pequena fração de memória adicional é usada, o que nos parece um bom resultado.

Para entender se o tamanho de *firmware* usado pelo nosso sistema é adequado aos dispositivos da IdC foram analisadas três placas de desenvolvimento (Zolertia RE-Mote[1], Arago Systems Wismote[2] e Texas Instruments CC2538DK[3])

5.2 Processo de Bootstrapping

Nesse caso, após a primeira sequência todo o processo de abertura da interface, seleção de *hardware* e compilação já está realizado, representando um ganho de 22 segundos resultando num tempo real de 12 segundos. Desta forma podemos concluir que a nossa solução é prática e adequada aos cenários IdC.

[1] http://zolertia.io/product/hardware/re-mote
[2] http://www.wismote.com
[3] http://http://www.ti.com/tool/cc2538dk

Tabela 1: Consumos Energéticos

Modo	V	mA	W
Radio ON	9.0	17	0.15
Radio OFF	9.0	5.2	0.05

Tabela 2: Tempos de Bootstrapping

Operação	Tempo(s)
Inserir Novo Dispositivo	2
Abrir Interface	5
Selecionar Hardware	4
Compilar Código Fonte	13
Eliminar Firmware Existente	5
Upload Novo Firmware	3
Remover Novo Dispositivo	2

6 Conclusões

Obter comunicações seguras não é uma tarefa fácil, dadas as limitações dos dispositivos utilizados na redes IdC. Para atingir este objetivo realizámos uma análise extensiva aos protocolos e estratégias de mitigação de ataques "vampiros" existentes estabelecendo uma pilha de protocolos que usámos nos dispositivos com recursos limitados. Experiências em *hardware* real confirmam que a solução tem tanto um tamanho como um consumo energético compatível com os dispositivos IdC. Face às exigências adicionais em termos de infraestrutura que o nosso modelo de comunicação necessita, propusemos uma arquitetura central que vai de encontro às necessidades correntes dos ambientes IdC abordados e emprega mecanismos ao nível aplicacional para reduzir o número de pacotes que fluem na rede. Esta infraestrutura possui também os componentes adicionais de baixo impacto em termos de recursos necessários que permitem a execução do processo de *bootstrapping* seguro proposto, automático, que não necessita de conhecimento interno por parte do operador e com métricas de usabilidade que o tornam adequado aos ambientes IdC.

Como trabalho futuro, salientamos a necessidade de proteção da memória dos dispositivos para impedir o roubo de credenciais de segurança e consequente clonagem dos dispositivos. Deixamos como sugestão o uso de circuitos integrados com mecanismos de impedimento de leitura ou o bloqueio via software de certas regiões de memória dos dispositivos onde residem as chaves de rede.

Referências

1. Ukil, A., Bandyopadhyay, S., Pal, A.: Privacy for IoT: Involuntary privacy enablement for smart energy systems. 2015 IEEE International Conference on Communications (ICC) (2015) 536–541
2. Vasserman, E.Y., Hopper, N.: Vampire attacks: Draining life from wireless ad Hoc sensor networks. IEEE Transactions on Mobile Computing **12**(2) (2013) 318–332
3. Pongle, P., Chavan, G.: A survey: Attacks on RPL and 6LoWPAN in IoT. 2015 International Conference on Pervasive Computing (ICPC) **00**(c) (2015) 1–6
4. Fips, N.: 197: Announcing the advanced encryption standard (AES). ... Technology Laboratory, National Institute of Standards ... **2009** (2001) 8–12

5. Corp, C.: Formal Specification of the CCM * Mode of Operation René Struik Voice : Fax : Re : Abstract as well as some (informational) design rationale . This document is an edited Purpose Notice. (2005) 1–19