AdC: um Mecanismo de Controle de Acesso para o Ciclo de Vida das Coisas Inteligentes

Lucas Fernando Puhl
puhl@alunos.utfpr.edu.br

Departamento de Informática - DAINF
Universidade Tecnológica Federal do Paraná

Segurança Computacional, Junho, 2018

Sumário

Internet das Coisas (IdC)

- A Internet das Coisas (IdC) já é parte de nossas vidas. A ideia de um ambiente inteligente formado por elementos computacionais heterogêneos interconectados se tornou realidade.
- A IdC combina objetos físicos, sensores e atuadores, gerando sistemas ciber-físicos (cidades, redes e casas inteligentes).

Autenticação das Coisas (AdC)

- A autenticação é primordial para a segurança. Os mecanismos de autenticação permitem a distinção entre dados legítimos e forjados, assim como a determinação da origem de uma mensagem.
- Mesmo que a segurança em IdC tenha recebido atenção da comunidade científica, as abordagens existentes não atendem às necessidades de autenticação em IdC.

Autenticação das Coisas (AdC)

Necessidade do AdC

- Uma das principais características do AdC é que os dispositivos são implantados sem a necessidade de conexões cabeadas. Para tal, o AdC lança mão da proteção física das casas e de um dispositivo confiável (e.g., o smartphone do administrador do domínio Doméstico) para servir de ponte durante a primeira comunicação entre o novo dispositivo e o servidor Doméstico e, assim, engendrar a segurança.

Autenticação das Coisas (AdC)

Public Key Infrastructure (PKI)

- Os esquemas tradicionais baseados em infraestrutura de chaves públicas PKI e certificados causam sobrecarga significativa de CPU, memória, armazenamento, comunicação e gerenciamento, o que inviabiliza sua utilização em dispositivos de IdC.

Diferentes domínios:

- IdC geralmente requer que dispositivos de um certo domínio sejam capazes de interoperar, de forma segura, com dispositivos que pertençam a diferentes domínios. Acontece que grande parte dos esquemas de autenticação projetados para dispositivos com restrições de recursos assumem que os dispositivos pertencem a um domínio único e, portanto, não podem ser diretamente aplicados a IdC.

Problemas

Autenticação e controle de acesso ainda não estão solucionadas

- A criptografia de chaves públicas tradicional é computacionalmente mais cara, não adequada para a maior parte dos dispositivos de IdC. Além disso, dada a heterogeneidade dos dispositivos de IdC e seus múltiplos domínios, esquemas de segurança desenvolvidos para outros cenários de redes sem fio e dispositivos com restrições de recursos não podem ser diretamente aplicados a IdC.

Objetivo

- Projetar, desenvolver e avaliar um esquema de autenticação e controle de acesso para todo o ciclo de vida dos dispositivos de IdC.
- Sendo composta por uma família de protocolos criptográficos que provê autenticação e controle de acesso a cada um dos estágios do ciclo de vida de um dispositivo, a saber:
  - **Manufatura;**
  - **Aquisição;**
  - **Implantação;**
  - **Operação;**
  - **Descarte.**

Objetivo

- Nosso principal objetivo é projetar uma solução de autenticação e controle de acesso adequada para IdC.
- Neste contexto, consideramos:
  - eletrodomésticos inteligentes que interagem entre si;
  - interagem com os dispositivos pessoais dos usuários;
  - e interagem com um servidor Doméstico e um servidor Cloud, que age com o fabricante dos dispositivos.
- Nossa solução deve atender às restrições de eficiência e segurança exigidas pelas aplicações de IdC.

Abordagem

- A distribuição das chaves para a inicialização da segurança.
- O controle de acesso para gerenciar as permissões de execução das operações nos dispositivos de IdC.

Abordagem

- **Criptografia Baseada em Identidade(CBI)** para a distribuição das chaves;
- Enquanto **Criptografia Baseada em Atributos (CBA)** é aplicada para implementar criptograficamente o esquema de **Controle de Acesso Baseado em Atributos (CABA)**, que, por sua vez, gerencia o controle de acesso às operações nos dispositivos.

Protocolos

Criptografia Baseada em Identidade(CBI)

- O principal desafio para adoção de CBI é o conhecido problema da custódia de chaves, onde o Gerador de Chaves Privadas (GCP) pode passar-se por qualquer usuário no sistema. Para resolver este problema, concebemos uma **arquitetura de dois domínios CBI isolados:** o domínio do fabricante, chamado Cloud, e o domínio local, a que chamamos Doméstico.
- Estes dominiosos definem as relações de confiança fabricante-dispositivo e (usuário-)(dispositivo-)dispositivo, respectivamente. Não há sobreposição dessas relações e, portanto, material criptográfico gerado no domínio Cloud não é válido no domínio Doméstico e vice-versa.

Protocolos

Controle de Acesso Baseado em Atributos (CABA)

- No controle de acesso das operações nos dispositivos, o CABA simplifica as relações, substituindo permissões discricionárias por políticas baseadas nos atributos dos usuários, o que permite considerar características do recurso e informação de contexto.

Protocolos

Criptografia Baseada em Atributos (CBA)

- Fornece mecanismos criptográficos para adoção do CABA. Mais precisamente, um esquema de assinatura CBA;

Assinatura Baseada em Atributos (ABA):

- Pode ser usado para assegurar um subconjunto mínimo de atributos necessários para a execução de uma operação.

Protocolos Auxiliares

- **ChaveDeSessão:** derivar chaves de sessão;
- **AcordoDeChaves:** acordar chaves par a par (pairwise keys);
- **Distribuição**: distribuir chaves privadas;
- **Vinculação e Desvinculação:** vincular e desvincular um usuário de um dispositivo no domínio Cloud.

Protocolos Auxiliares

ChaveDeSessão

- O procedimento ChaveDeSessão(k, i)recebe uma chave previamente compartilhada k e um contador i mantido atualizado entre os interlocutores para derivar chaves de sessão. Esta ideia é baseada no trabalho de [Perrig et al. 2001], onde uma função pseudoaleatória é usada para gerar novas chaves.

Protocolos Auxiliares

AcordoDeChaves

- O protocolo AcordoDeChaves(A, B) é baseado no trabalho de [Sakai et al. 2000]. Ele permite que dois dispositivos (A e B) do mesmo domínio Doméstico troquem chaves par a par empregando emparelhamentos bilineares.

Protocolos Auxiliares

Distribuição

- Já no protocolo DISTRIBUIÇÃO(D, S, Z , Y) , o servidor S do domínio Z emite a Y chave privada S(D,Z), relativa ao criptossistema Y, para o dispositivo D. A segurança desta operação é garantida pelo uso de uma chave de sessão derivada de uma chave par a par k(D,S) previamente compartilhada entre o dispositivo D e o servidor S.

Protocolos Auxiliares

Vinculação e Desvinculação

- Por fim, os protocolos VINCULAÇÃO(D,U) e DESVINCULAÇÃO(D,U) vincula e desvincula, respectivamente, o dispositivo D e o usuário U no domínio Cloud.

Autenticação das Coisas

Manufatura

Executado durante a fabricação dos dispositivos:

- Neste estágio, o material criptográfico do domínio Cloud é carregado nos dispositivos. O servidor Cloud C gera a identidade id(D,C) e a chave privada S(D,C) do dispositivo D, no domínio Cloud (subscrito C) e criptossistema CBI.
- Posteriormente, Cloud C gera uma chave par a par entre C e D, cria um contador c(D), e os envia juntamente com o restante do material criptográfico, para o dispositivo D.
- Este envio é feito de forma segura via um canal físico, garantindo comunicação confidencial e autenticada.

Autenticação das Coisas

Aquisição

- O distribuidor comunica a Cloud e gera um código PIN para liberar o dispositivo e entrega ao usuário.

Autenticação das Coisas

Implantação

Inicializa a segurança dos dispositivos em seus domínios Domésticos

- Para inserir o novo dispositivo, o usuário digita o código de acesso pin.
- Em seguida, obtém a identidade no domínio Doméstico e a chave pública, criptossistema CBI do domínio, junto com o contador do domínio que vai prover freshness para broadcasts originados no domínio.
- O dispositivo gera, de forma aleatória, uma chave par a par, temporária entre dispositivo e domínio, a cifra usando a chave pública recém recebida e envia a cifra resultante, juntamente com sua identidade, no domínio e as operações suportadas para o dispositivo. Então, o usuário usa o dispositivo para configurar os atributos e predicados das operações.

Autenticação das Coisas

Operação

Governa a operação cotidiana entre os dispositivos:

- O dispositivo A faz a requisição.
- B responde com o predicado da operação requisitada.
- O dispositivo A prova que pode executar a operação com a assinatura da mensagem usando sua chave privada CBA do mesmo domínio.

Autenticação das Coisas

Descarte

- Um usuário usa o dispositivo A para requisitar a operação de descarte do dispositivo B.
- Então, B se desvincula do seu dono e apaga todas suas chaves, de ambos os domínios Cloud e Doméstico, além de todas as chaves simétricas compartilhadas com outros dispositivos.

Implementação

- Os servidores Cloud e Doméstico são implementados em LAMP (Linux, Apache, MySQL e PHP).
- Enquanto que a implementação dos dispositivos se dá em smartphones Android.
- Os servidores comunicam-se com os dispositivos usando o protocolo HTTP e invocam do PHP os esquemas criptográficos previamente compilados.
- Nos dispositivos Android, as interfaces de usuário são implementadas utilizando a interface nativa e a chamada das funções criptográficas é feita via Interface Nativa Java (Java Native Interface – JNI).

Criptografia

- A camada de software criptográfico é construída em linguagem C.
- Criptográfica RELIC: processadores de 8 e 16 bits e 4KB de RAM.
- Implementar de forma eficiente diversos algoritmos criptográficos baseados em curvas elípticas em diferentes níveis de segurança.
- Otimização por meio de código assembly elaborado especificamente para a arquitetura ARM.

Criptografia Baseada em Atributos (CBA)

- A implementação é dependente de Criptografia Baseada em Emparelhamentos (CBE) e de Monotone Spam Programs – (MSPs), que são matrizes usadas para representar os predicados do AdC como expressões booleanas.

Computação de Emparelhamentos

- A implementação é dependente de Criptografia Baseada em Emparelhamentos (CBE) e de Monotone Spam Programs – (MSPs), que são matrizes usadas para representar os predicados do AdC como expressões booleanas.
- Pode ser dividida em duas fases: o laço de Miller, que consiste de um algoritmo de quadrado e multiplicação e a exponenciação final.

Avaliação

- A segurança do AdC foi avaliada utilizando uma ferramenta de verificação automática de protocolos de segurança e mostramos que a solução é adequada para dispositivos embarcados com restrição de recursos.

Segurança

- O AdC garante as seguintes propriedades de segurança:
  - autenticação, usando MACs e assinaturas digitais;
  - confidencialidade, usando cifras;
  - freshness, usando nonces (number used once) e contadores;
  - integridade, usando MACs, assinaturas digitais e funções de hash;
  - não-repúdio, usando assinaturas digitais.

Scyther

Uma ferramenta de verificação automática de protocolos de segurança

- A ferramenta faz sua análise verificando se um dado protocolo é vulnerável a ataques que poderiam violar certas propriedades de segurança.
- O Scyther assume que as primitivas criptográficas são ideais do ponto de vista de segurança e encontra falhas decorrentes de trocas de mensagens nos protocolos.

Experimentos

- Foram avaliamos os custos computacionais, quantificados em função do número de primitivas criptográficas de maior custo – emparelhamentos e multiplicações de pontos de curvas elípticas por escalar e a sobrecarga de comunicação do AdC, quantidade extra de bytes devido à criptografia
- O esquema criptográfico mais custoso é o ABA, cerne do mecanismo de controle de acesso e o mais frequentemente utilizado no AdC.
- A sobrecarga de comunicação é computada considerando nonces de 16 bytes e requisições e confirmações de 1 byte.

Experimentos

- AdC foram avaliados em duas plataformas:
  - Arduino Due (processador ARM M3 de 32 bits e 84 MHz; 96 KB de RAM; 512 KB de memória flash)
  - Intel Edison (processador Atom de 32 bits e 500 MHz; 1 GB de RAM; 4 GB de memória flash)

Experimentos

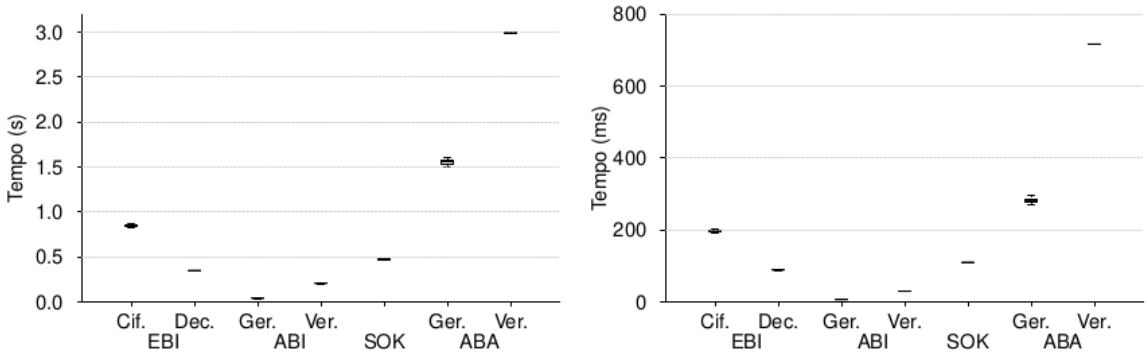
- Tempos de execução:
  - Cifração, Decifração, Geração de assinatura, e Verificação de assinatura.
  - Cada esquema foi executado 100 vezes.

Experimentos

- As figuras 7(a) e 7(b) mostram os tempos de execução para:
  - Cifração, Decifração, Geração de assinatura, e Verificação de assinatura.
  - Cada esquema foi executado 100 vezes.

Experimentos

- Como esperado, o ABA é o esquema mais custoso. No Due são necessários 1,6s para gerar assinaturas e menos de 3,0s para verificá-las, usando predicados da forma $A \wedge B$. Já no Edison estes números chegam a cerca de 300ms e 750ms, respectivamente.



(a) Due (b) Edison

Figura 7. Tempos de execução.

Experimentos

- A figura 8 mostra os tempos de execução para o ABA no Due, com predicados da forma $A \wedge B$, diferentes níveis de segurança (80 e 128 bits) e as verificações determinística e probabilística de assinatura.
- A figura 9 mostra os tempos de verificação de assinatura para diferentes predicados. No eixo x representamos a variação no número de atributos.

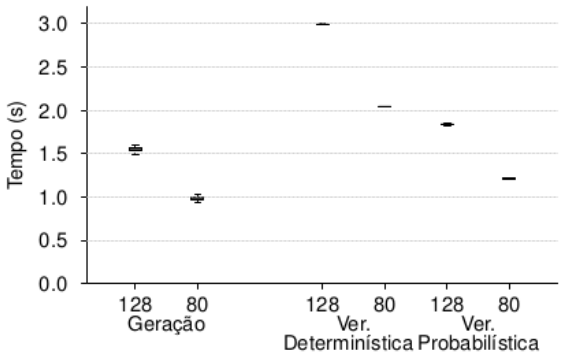


Figura 8. ABA no Due.

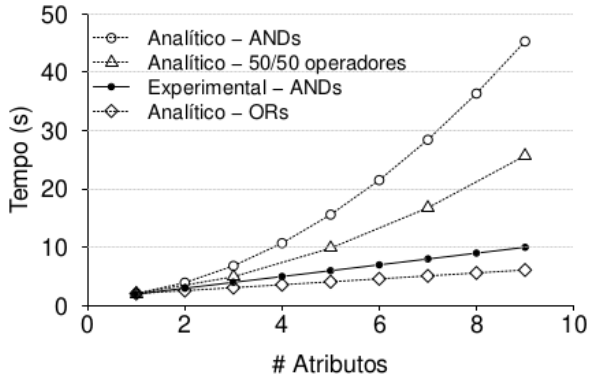


Figura 9. Ver. ABA no Due.

Conclusão

- O AdC foi avaliado analítica e experimentalmente.
- Os resultados analíticos mostram que a sobrecarga imposta pelo AdC tanto para CPU quanto para comunicação, no pior caso, são gerenciáveis até mesmo em dispositivos com limitações de recursos.
- Nós também utilizamos uma ferramenta de verificação automática para validar as propriedades de segurança do AdC.
- Por fim, os resultados experimentais mostram que o AdC pode ser executado com baixo custo em dispositivos com restrições computacionais e com custo desprezíveis em dispositivos computacionalmente poderosos.