

Segurança e Aplicações de Hardware Confiável  
Privacy Systems in the Cloud

Pedro ANTUNES - up201507254

May 31, 2022



Instructor: Bernardo Portela

## 1 Introdução

As aplicações baseadas em serviços de nuvem devem garantir a privacidade das informações dos seus respetivos clientes. Para este efeito, os dados privados de cada cliente têm de ser guardados com recursos a técnicas criptográficas, como um meio de garantir confidencialidade e integridade. Assim, temos necessariamente de utilizar chaves para as operações criptográficas.

Através da utilização da criptografia garantimos a proteção dos dados. No entanto, caso algum ator malicioso consiga obter as chaves dos clientes, ou, caso a máquina no sistema de nuvem onde as chaves são armazenadas estiver infetada, as chaves (secretas) criptográficas destinadas a proteger as informações dos clientes ficam expostas. Consequentemente, as informações dos clientes estão vulneráveis.

Para resolver este problema, precisamos de adquirir um hardware seguro que fique encarregue de guardar as chaves criptográficas de cada cliente no sistema. Um Hardware Security Module fornece um ambiente de computação seguro, onde efetua operações criptográficas e preserva e protege chaves secretas para essas operações. Assumindo que este hardware é resistente a qualquer tipo de manipulação, podemos servir-nos dele para garantir uma âncora de confiança para o nosso sistema.

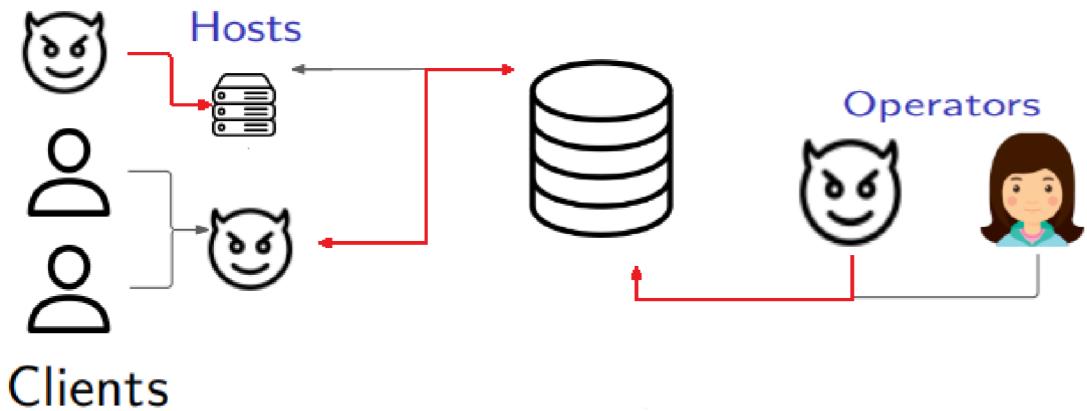
## 2 Identificação de problemas

Para obtermos um sistema de gestão de chaves para clientes que utilizam serviços baseados em cloud, essencialmente a maior preocupação é o armazenamento das chaves em redor deste sistema. Olhando para o sistema de uma forma abrangente, cada cliente pode utilizar um, ou, mais do que um serviço prestado pelo sistema em cloud.

Visto que o sistema em cloud lida com os clientes através do intermédio de serviços, cada cliente teria de estar identificado em cada serviço. As chaves que identificam os clientes em certos serviços teriam de ser criadas e geridas por operadores humanos que estariam encarregues de mapear os clientes nos serviços do sistema.

Logo, para cada cliente era necessário tantas chaves criptográficas quanto os serviços que ele utiliza no sistema de cloud. Isto torna insuportável que qualquer sistema deste género consiga guardar e gerir esta imensidão de chaves, o escalonamento é terrível e torna-se um problema.

O armazenamento das chaves do sistema é de facto, o grande ponto sensível de todo o sistema e, o que deve estar à prova de bala, pois caso o atacante consiga chegar às chaves do sistema, todas as informações dos clientes estão expostas e comprometidas. Para além disto, o atacante deste sistema pode fazer-se passar pelas entidades descritas em cima para forjar ou manipular as chaves dos clientes do sistema.



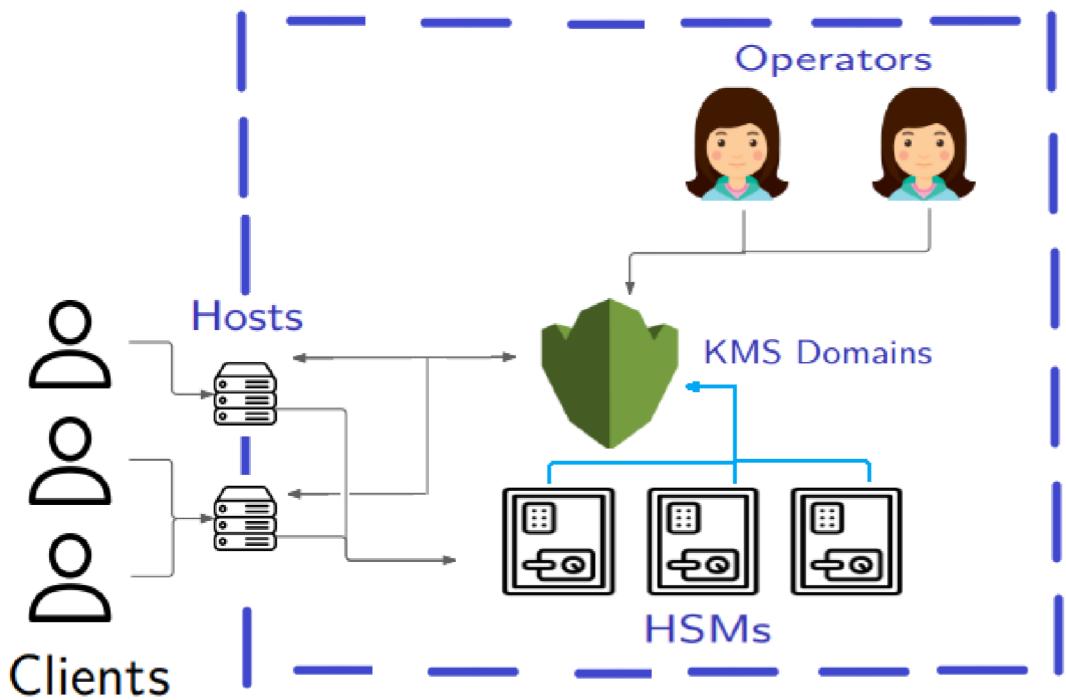
*Figura 1.* Modelo atacante do sistema de gestão de chaves

Através da *Figura 1.*, podemos observar os caminhos que o atacante pode seguir no sistema. Tais como:

- **Clientes**, isto é, um atacante pode tentar fazer-se passar por outro cliente e efetuar pedidos a serviços fingindo ser outra pessoa.
- **Serviços**, isto é, um atacante pode ter controlo de um serviço e pode efetuar pedidos de cifrar ou decifrar ao servidor de gestão de chaves fingindo ser outro serviço.
- **Operadores**, isto é, um atacante pode ter controlo ou influenciar um operador a atribuir uma chave do conhecimento do atacante a um certo cliente.

### 3 Solução

A segurança e proteção dos dados nestes tipos de sistemas em nuvem passa por utilizar um hardware confiável que nos garanta proteção sobre as chaves criptográficas no sistema e ao mesmo tempo que o número de chaves seja escalável para um número significativo de clientes. Para isto, utilizamos um Hardware Security Module como âncora de confiança que desempenha as operações criptográficas e, em simultâneo, desenvolver uma interface sobre o hardware para efetuar a gestão e proteção das chaves dos clientes do sistema do lado exterior do hardware confiável.



*Figura 2.* Desenho do sistema de gestão de chaves

Para a solução deste sistema, consideramos e assumimos que:

- Os clientes são autenticados no sistema de uma forma segura e confiável;
- O sistema de nuvem fornece vários serviços aos clientes;
- Cada serviço do sistema têm associado um par de chaves de criptografia assimétrica, que servirão para identificar o serviço em questão no sistema;
- Existem operadores humanos que têm associados um par de chaves de criptografia assimétrica, que servirão para identificar o operador em questão no sistema;
- Podem existir vários Hardware Security Module disponíveis para desempenhar as operações criptográficas;
- A interface de gestão de chaves utiliza domínios para guardar as chaves criptográficas dos serviços e para controlar as entidades que podem utilizar a sua chave.
- A interface de gestão de chaves mapeia serviços com domínios;
- Os operadores humanos inicializam pedidos de criação de domínios, definem e verificam o controlo de acesso aos domínios criados. Também podem remover domínios do sistema;

- Quando ocorre uma criação de um domínio, esta ação foi decidida e aprovada por um grupo de operadores honestos que pertencerão ao domínio criado.
- Um domínio contém as chaves de domínio associadas às HSMs que pertencem ao domínio e que ficarão encarregues de desempenhar as operações criptográficas sobre esse domínio. Estas chaves de domínio estão cifradas e apenas podem ser decifradas dentro de um HSM que pertence ao domínio.
- Os Hardware Security Modules cifram as chaves de domínio de acordo com as entidades de Hardware Security Modules registadas no domínio. Serão estas entidades que poderão reverter esta operação e recuperar a chave de domínio em texto limpo;
- Um operador pode rodar as chaves de domínio definidas para um certo domínio para alguma chave fresca que origina depois as novas chaves de domínio.
- Um domínio pode ser atualizado de forma a que sejam adicionadas novas entidades, quer HSMs e/ou operadores.
- Um domínio contém uma confiança, que é utilizada para gerir as atualizações de um domínio. Sobre os operadores pertencentes no domínio, é definido o número de um conjunto de operadores em função de que se assegure que, em que qualquer subconjunto possível de operadores do domínio, exista pelo menos 1 operador honesto.
- Quando um domínio é atualizado com novas entidades, estas entidades terão de estar atestadas com assinaturas válidas. O número de assinaturas válidas terá de ser maior ou igual ao número definido para um conjunto de operadores no momento da criação do domínio.
- Assim, qualquer entidade nova adicionada num domínio terá de ter sido atestada por 1 operador honesto.
- Quando existe uma atualização de confiança, uma nova confiança é criada e possui a hash da confiança precedente. Isto faz com que não seja possível forjar novas confianças.
- Os Hardware Security Modules utilizados no sistema estão certificados e em conformidade com Federal Information Processing Standards (FIPS) Validation.
- O algoritmo criptográfico de criptografia simétrica escolhido para operações de cifrar e decifrar informação foi o AES-GCM (Advanced Encryption Standard - Galois/Counter Mode). As chaves utilizadas neste algoritmo são chaves de 256 bits. Este algoritmo produz cifras autenticadas que garantem a confidencialidade e integridade da informação. Assume-se que não é possível forjar mensagens neste algoritmo.
- O algoritmo criptográfico de criptografia assimétrica escolhido para operações de cifrar ou decifrar informação, e assinar ou verificar assinaturas foi o RSA/ECB/OAEPWithSHA-256AndMGF1Padding. As chaves utilizadas neste algoritmo são chaves de 4096 bits. Este algoritmo produz cifras autenticadas que garantem a confidencialidade e integridade da informação e assinaturas digitais que garantem integridade e não-repúdio. Assume-se que não é possível forjar mensagens nem assinaturas neste algoritmo.
- O algoritmo criptográfico de função de compressão escolhido para retornar hashes foi o SHA-256 ( Secure Hashing Algorithm - 256 bits). Assume-se que não é possível encontrar colisões de imagens neste algoritmo.

### 3.1 Fluxo de comunicação

Em termos de fluxo de comunicação do sistema de gestão de chaves, consideramos que:

- Os operadores e os Hardware Security Modules criam domínios na interface de gestão de chaves;
- Os clientes interagem com os serviços na nuvem;
- Os serviços interagem com a interface de gestão de chaves, obtendo assim o seu domínio.
- Os serviços interagem com as Hardware Security Modules pertencentes ao seu domínio, reencaminhando os pedidos criptográficos dos clientes.
- Os operadores atualizam estados de domínios na interface de gestão de chaves através de um conjunto de assinaturas verificadas pelos Hardware Security Modules.

## 4 Validação

O hardware confiável, as HSMs em si, são um pequeno pedaço neste sistema de gestão de chaves. No entanto, desempenham um papel fundamental de garantir que as chaves estão protegidas fora do contexto do hardware confiável.

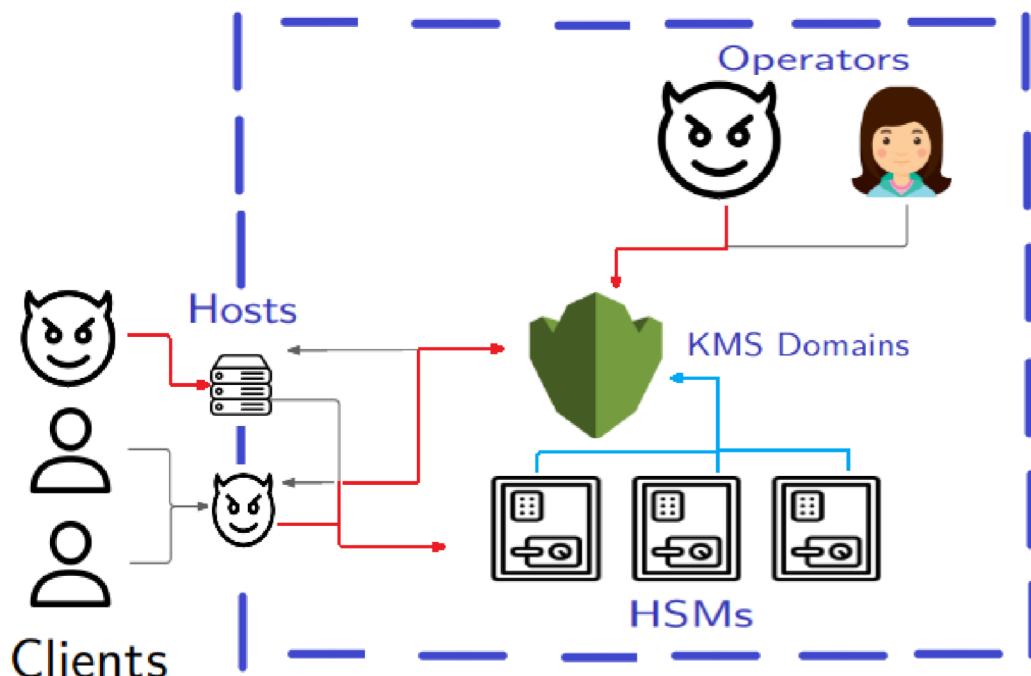


Figura 1. Modelo adversarial do sistema de gestão de chaves

Em termos do modelo atacante a que o sistema está sujeito, assumimos que:

- Ataques de canais laterais (como p.e ataques de tempo direcionados a leitura de informações na cache) não irão acontecer. Logo, o sistema não está preparado para se defender contra este tipo de ataques.
- O atacante não tem poder computacional suficiente para quebrar os algoritmos criptográficos utilizados para cifrar e assinar informações.
- O atacante não consegue manipular um Hardware Security Module (modificar circuitos ou injetar falhas) para falsificar chaves ou outro tipo de informação.
- A implementação de um Hardware Security Module foi formalmente verificada, de modo que não existam estouros na sua implementação que possam despejar informações para fora da HSM.
- Um cliente não consegue fazer-se passar por outro, visto que temos um passo de autenticação para identificar clientes.
- Um serviço não consegue fazer-se passar por outro, visto que utilizamos chaves criptográficas públicas para identificar cada serviço.
- As chaves criptográficas privadas de um serviço ou de um operador estão guardadas localmente e cifradas através de uma frase palavra-passe.
- Um operador pode ser influenciado por um atacante para deturpar ou manipular um certo domínio. No entanto, se no momento da criação de um domínio, este contém uma confiança honesta criada por operadores honesto que definiram um número correto de operadores honestos para que, qualquer operação que ocorra num domínio seja verificada por pelo menos 1 operador honesto, o operador influenciado pelo atacante não poderá realizar ações que deturpem e manipulem os domínios do sistema sem que essas ações sejam validadas por 1 operador honesto também.
- Um atacante pode ganhar acesso físico ou remoto ao armazenamento dos domínios do sistema. No entanto, não poderá decifrar as chaves dos domínios correspondentes a certos serviços. Logo a informação de cada serviço/cliente está protegida.
- Um atacante pode tentar forjar confianças devidamente assinadas por entidades do domínio. No entanto, este método não terá sucesso quando uma HSM tiver que realizar ações sobre um domínio. Isto porque, a confiança honesta inicialmente criada corresponde a uma hash única que serve como identificador. Logo, a HSM sabe que apenas existe a confiança precedente que corresponde ao valor de hash único. O atacante não tem poder computacional para quebrar os algoritmos de compressão utilizados para produzir hashes.

## 5 Prova de Conceito

A prova de conceito foi desenvolvida em Java e para efeitos práticos os serviços de um sistema de cloud, os Hardware Security Modules e os Operadores estão a ser simulados através de classes. A gestão das chaves é feita através de uma estrutura de dados Map, que mapeia os serviços em domínios.

A gestão de chaves tem um servidor de socket onde recebe comunicações através de sockets clientes dos vários componentes.

### 5.1 Gestão de chaves

#### 5.1.1 Domínio

Estrutura de dados que representa um domínio. Armazena a confiança do domínio que serve para controlar os acessos ao domínio. Ainda, armazena as chaves do domínio e uma assinatura produzida por uma entidade HSM dentro do domínio.

#### 5.1.2 Chave de domínio

Estrutura de dados que representa uma chave do domínio. Armazena a chave do domínio cifrada e uma lista da chave necessária para decifrar a chave do domínio cifrada com cada uma das chaves públicas das HSMs pertencentes ao domínio.

#### 5.1.3 Confiança

Estrutura de dados que representa a confiança de um domínio e que serve para controlar quais são as entidades que pertencem ao domínio. Armazena o identificador do domínio, as chaves públicas das entidades do domínio (operadores e HSMs), o núcleo de operadores confiáveis que faz a gestão da confiança do domínio, a hash da confiança precedente do domínio e uma assinatura produzida por uma entidade HSM dentro do domínio.

### 5.2 Componentes

#### 5.2.1 Operadores

Os operadores contém o seu par de chaves para operações criptográficas assimétricas. Realizam a operação de criar, atualizar e remover domínios em função das chaves públicas dos operadores do respetivo domínio e das chaves públicas dos HSMs do respetivo domínio.

#### 5.2.2 Serviços

Os serviços contém o seu par de chaves para operações criptográficas assimétricas e armazenam o controlador do seu domínio. Essencialmente, efetua pedidos de cifrar/decifrar às HSMs do seu domínio.

### **5.2.3 Hardware Security Module**

As HSM contém o seu par de chaves para operações criptográficas assimétricas e armazenam uma chave mestre simétrica. Realizam a operação de gerar chaves simétricas novas para serem utilizadas nas chaves de domínio. Gera chaves de domínio (wrapping). Por fim, realiza operações de cifrar de decifrar mensagem com as chaves de domínio (unwrapping).

Em acrescento, também efetua um papel importante na criação e na atualização dos domínios. No momento da criação de um domínio, um Hardware Security Model tem de assinar as informações desse domínio e da confiança do domínio para que não seja possível forjar domínios falsos.

No momento de atualização de um domínio, um Hardware Security Model tem de verificar se a confiança do domínio é consistente (se está assinada por um HSM pertencente ao estado de confiança atual), verifica se a própria HSM pertence ao estado de confiança antigo e, por último, se cada entidade da nova confiança está na antiga confiança ou então se foi atestada com assinaturas por um conjunto de operadores existentes na confiança (que se assume que exista pelo menos 1 operador honesto neste conjunto). Caso todas as verificações sejam todas bem sucedidas, a confiança do domínio avança para um novo estado e a HSM assina o novo estado de confiança do domínio e gera novas chaves de domínio (caso necessário).

## **6 Conclusão**

Nos dias de hoje, qualquer aplicação segura com funcionalidades relevantes para um cliente, utiliza ou deve utilizar criptografia para proteger quer as informações do lado do cliente quer as informações do lado da entidade prestadora de serviços.

Porém, utilizar apenas criptografia em muitos casos não nos resolvem todos os problemas de segurança nem asseguram que as informações estão protegidas. Como vimos neste trabalho, existem vários vetores de ataque que um adversário do sistema pode utilizar para conseguir ultrapassar métodos criptográficos simples.

Através destes problemas, é que surgem a utilização de um hardware confiável que nos minimizem a quantidade de vetores de ataque que um adversário pode tomar. Existem vários hardwares confiáveis que podem ser utilizados para este meio.

A opção de escolha recai por o que é que queremos proteger na nossa aplicação e no seu meio envolvente. É de salientar que, um hardware confiável na acaba com as ameaças e com os vetores de ataque que um atacante pode realizar. O propósito destes hardwares é que, em conformidade com algumas assunções, fazem com que esses vetores de ataque não tenham sucesso nem consigam deturpar a segurança de uma aplicação. Caso o ataque consiga identificar algum "buraco" nas assunções do sistema, então o nosso sistema que utiliza hardware confiável estará exposto a todos os ataques que estaria caso este hardware não existisse.

Para o contexto de sistema baseados em cloud, vimos que é necessário proteger as chaves criptográficas dos clientes e algo que faça com que o número de chaves no sistema seja escalável (visto que existirão bastantes clientes). Assim, a opção recaiu por um Hardware Security Module que

tem o simples papel de cifrar chaves para o contexto do sistema e fora do contexto do hardware confiável. Estas chaves, as chaves utilizadas pelos clientes, só poderão ser decifradas dentro do contexto do hardware confiável, garantindo assim o secretismo das chaves dos clientes e podendo assim o sistema armazenar estas chaves.