RSA

Teórica #8 de Criptografia Aplicada

Modelo de Criptografia Chave Pública (PKC)

Este modelo representa a cifração das mensagens de um canal de comunicação recorrendo a chaves públicas, que ambas as partes conhecem, e chaves privadas, onde cada agente tem e conhece apenas a sua própria chave.

RSA

Imaginamos o cenário em que duas entidades, Alice e Bob, desejam comunicar entre si. A Alice começa por gerar um par de chaves (1 pública e 1 privada), através do seguinte procedimento:

1. Produz dois "grandes" primos, de peso similar: *p* e *q*;
2. Define `n=p*q`
3. Produz `e < φ(n) = (p − 1)(q − 1)`, onde `(e, φ(n)) = 1`
4. Calcula `d = e ^ −1 (mod φ(n))`

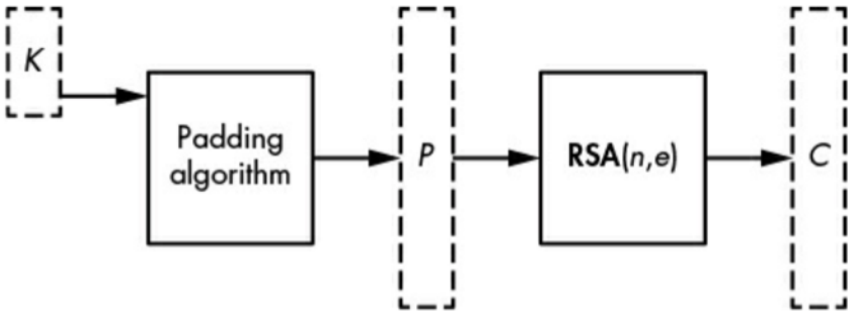
Chave pública = (n, e) Chave privada = (n, d)

Se o Bob deseja enviar uma mensagem *m* à Alice, calcula a mensagem cifrada `c = m^e (mod(n))` e envia-lhe. A Alice decifra a mensagem cifrada através de: `c^d (mod(n)) = (m^e (mod(n)))^d (mod(n)) = m^e*d (mod(n)) = m^kφ(n)+1 = m`

O RSA e na generalidade todos os PKC Ciphers, são fáceis de utilizar mas pouco eficientes, sendo cerca de 1000 mais lentos que cifras simétricas. O facto que também usa uma chave pública, que é *pública*, as cifras PKC são vulneráveis a ataques de dicionário quando a mensagem original tem origem num campo de mensagens conhecidas ou de pequeno período (tamanho).

Optional Asymmetric Encryption Padding (OAEP)

O OAEP é um técnica usada para tornar a cifração com RSA mais forte. Esta consiste na adição de padding à mensagem original, de forma a que seja mais difícil adivinhar a mesma.



Optimal Asymmetric Encryption Padding (OAEP)

Esta técnica utiliza um PRNG (Pseudorandom Number Generator), de forma a garantir as propriedades da indistinguibilidade e não maleabilidade. O OAEP é seguro desde que a função RSA e PRNG sejam seguros e também que as funções hash não sejam muito fracas.

Para cifrar textos com RSA deve-se usar o OAEP.

Assinar com RSA

Para assinar com RSA, um apenas precisa de calcular `m^d (mod(n))`

Sendo que a verificação da assinar é a decifração da mensagem com a chave pública.

Problema de Assinaturas Simples RSA

Um atacante pode falsificar assinaturas de 0, 1 e (n-1), pois:

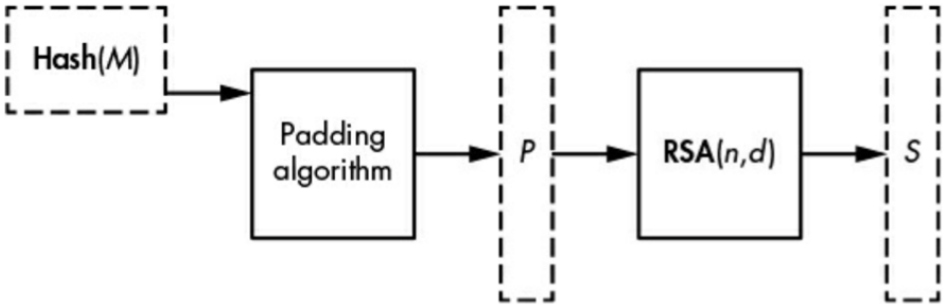
```
0^d (mod(n)) = 1^d (mod(n)) = 1
    = (n – 1)^d (mod(n)) = –1
```

Um atacante pode também realizar um *blinding attack*, dado que se um encontra um valor *r* dado que `r^e * m (mod(n))` é o valor da mensagem plausível de ter sido assinada, então é fácil obter *m^d*, dado que:

```
S = (r^e * m)^d = rm^d
```

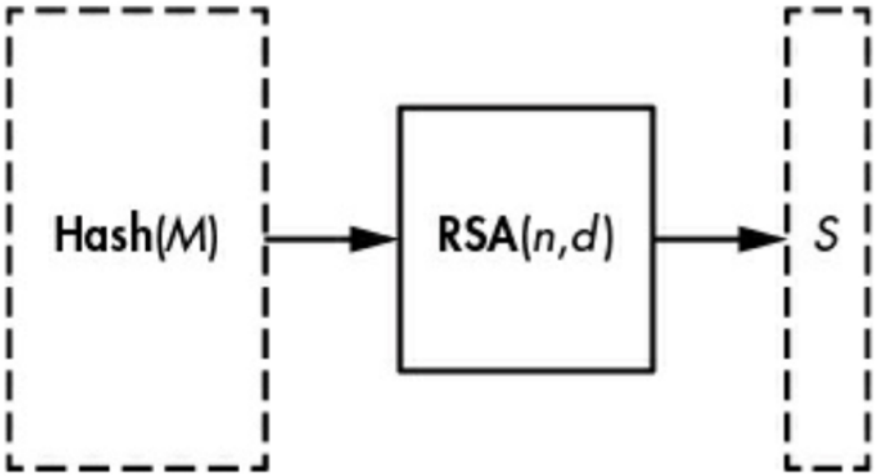
PSS (Probabilistic Signature Scheme) Signature Standard

Esquema de assinaturas com RSA, provado como seguro, mas complexo, o que leva a erros desnecessários de implementação.



Full Domain Hash Signatures (FDHS)

Implementação mais simples que PSS, mas com menores provas de segurança.



Em termos de segurança, PSS é preferível ao FDHS, dado as suas provas de segurança. Contudo, na maioria dos contextos, a aplicação de FDHS não traz grandes perdas de segurança em comparação com o PSS. Contudo, em alguns contextos, PSS é preferível devido à aleatoriedade adicionada por estes, evitando problemas causadas na implementação deste, como por exemplo fault attacks e side channel attacks.