Présentation des protocoles RSAES-OAEP et RSASSA-PSS

M2 MIC - Cryptographie asymétrique

Jérémie Nekam et Daniel Resende

université PARIS DIDEROT PARIS 7

Mardi 24 octobre 2017

Sommaire

Sommaire

Sommaire

Sommaire

Sommaire

Introduction

Deux protocoles pour deux utilisations différentes développés par M. Bellare et P. Rogaway :

Introduction

Deux protocoles pour deux utilisations différentes développés par M. Bellare et P. Rogaway :

Chiffrement/déchiffrement : RSAES-OAEP (RSA Encryption Scheme Optimal Asymmetric Encryption Padding),

Introduction

Deux protocoles pour deux utilisations différentes développés par M. Bellare et P. Rogaway :

Chiffrement/déchiffrement : RSAES-OAEP (RSA Encryption Scheme Optimal Asymmetric Encryption Padding),

Signature : RSASSA-PSS (RSASSA Probabilistic Signature Scheme).

Pourquoi utiliser OAEP ?

D. Bleichembacher a trouvé une attaque CCA-2 sur le protocole suivant :

Pourquoi utiliser OAEP ?

D. Bleichembacher a trouvé une attaque CCA-2 sur le protocole suivant :

Definition (PKCS #1 v1)

Soit M le message à chiffrer. On note $EB = 00 \parallel 02 \parallel Padding \parallel 00 \parallel M$

Le schéma OAEP standard

Algorithme 1 Schéma OAEP

Require: Un message m, un aléa r et deux oracles G et H.
Ensure: Un message m' tel que $m' = s \parallel t$.

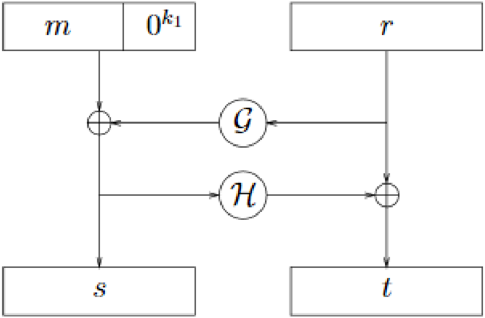


Figure – OAEP

Sommaire

Protocole RSAES-OAEP

Le protocole RSAES-OAEP se décompose en trois parties :

Protocole RSAES-OAEP

Le protocole RSAES-OAEP se décompose en trois parties :

- La génération des clés,

Protocole RSAES-OAEP

Le protocole RSAES-OAEP se décompose en trois parties :

- La génération des clés,
- Le schéma EM-OAEP,

Protocole RSAES-OAEP

Le protocole RSAES-OAEP se décompose en trois parties :

- La génération des clés,
- Le schéma EM-OAEP,
- La primitive RSAEP (resp. RSADP) pour le chiffrement (resp. déchiffrement).

Génération des clés RAES-OAEP

Clés publiques

On garde les mêmes clés (n, e) avec les mêmes propriétés que le RSA classique.

Génération des clés RAES-OAEP

Clés publiques

On garde les mêmes clés (n, e) avec les mêmes propriétés que le RSA classique.

Clés privées

- soit (p, q, d) tel que $e \cdot d = 1 \ mod \ (ppcm(p-1, q-1))$,

Génération des clés RAES-OAEP

Clés publiques

On garde les mêmes clés (n, e) avec les mêmes propriétés que le RSA classique.

Clés privées

- soit (p, q, d) tel que $e \cdot d = 1 \ mod \ (ppcm(p-1, q-1))$,
- soit $(p, q, dP, dQ, qInv)$ où $q \cdot qInv = 1 \ mod \ p$, $e \cdot dP = 1 \ mod \ q$ et $e \cdot dP = 1 \ mod \ q$.

Le schéma EM-OAEP

Algorithme 2 Schéma EM-OAEP

Require: Un message m, un aléa $seed$ et $Hash$ des données spécifiant la fonction de hachage à utiliser

Ensure: Un message EM.

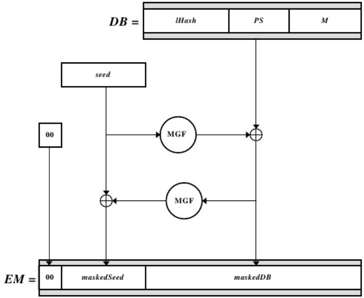


Figure – EM-OAEP

Chiffrement/déchiffrement de RAES-OAEP

RSAEP - Chiffrement

On garde les mêmes paramètres et propriétés que le RSA classique.

Chiffrement/déchiffrement de RAES-OAEP

RSAEP - Chiffrement

On garde les mêmes paramètres et propriétés que le RSA classique.

Algorithme 4 RSADP - Déchiffrement

Require: Un message chiffré c et une clé privé $K = (n, p, q, d)$ ou $(p, q, dP, dQ, qInv)$.
Ensure: Un message clair m
  if c n'est pas une entrée valide **then**
    return ERREUR
  end if
  if $K = (n, p, q, d)$ **then**
    return $m = c^d \ mod \ n$
  end if
  $m_1 = c^{dP} \ mod \ p$
  $m_2 = c^{dQ} \ mod \ q$
  $h = (m_1 - m_2) \cdot qInv \ modp$
  return $m = m_2 + q \cdot h$

Sécurité du protocole

Definition (Sécurité sémantique)

Soit m_0, m_1 deux messages choisies par l'attaquant. Soit c un challenge qui est le chiffré de m_0 ou m_1.
On dit qu'un protocole est sémantiquement sûr si l'attaquant ne peut pas distinguer m_0 ou m_1.

Attaque de Shoup [Sho01]

Proposition

Le protocole OAEP n'est pas totalement sémantiquement sûr.

Idées de preuve pour l'attaque de Shoup [Sho01]

Definition (Xor-malléable)

Soit f une permutation à sens unique avec trappe. On dit que f est xor-malléable, si on a une probabilité non-négligeable de pouvoir calculer $f(t \oplus a)$ en connaissant $f(t)$ et a.

Idées de preuve pour l'attaque de Shoup [Sho01]

Definition (Xor-malléable)

Soit f une permutation à sens unique avec trappe. On dit que f est xor-malléable, si on a une probabilité non-négligeable de pouvoir calculer $f(t \oplus a)$ en connaissant $f(t)$ et a.

Théorème

S'il existe un schéma xor-malléable alors il existe une permutation à sens-unique avec trappe, telle que lorsqu'on utilise OEAP le schéma de chiffrement qui en résulte n'est pas sûr dans un modèle d'oracle aléatoire.

Soit f_0 xor-malléable sur k_0 bits. Soit un A algorithme qui calcule $f_0(t \oplus \delta)$ à partir de $(f_0, f_0(t), \delta)$.
Soit f telle que $f(s \parallel t) = s \parallel f(t)$ une permutation à sens-unique avec trappe où $s \in \{0,1\}^{n+k_1}$ et $t \in \{0,1\}^{k_0}$.
On pose que le schéma OAEP utilise f.
L'attaquant reçoit le chalenge y' , il peut l'écrire tel que $y' = s' \parallel f_0(t')$.
L'attaquant choisi un message arbitraire différent de zéro Δ tel que :

$$\begin{aligned} \mathrm{s} &= \mathrm{s'} \oplus(\delta \parallel 0^{k_1}) \\ \mathrm{v} &= A(\mathrm{f}_0, f_0(t'), H(s) \oplus H(s')) \\ \mathrm{y} &= \mathrm{s}\| \; v \end{aligned}$$

Si y est un chiffrement valide de $x = x' \oplus (\delta 0^{k_1}) et \mathrm{v} = \mathrm{f}_0(t' \oplus H(s) \oplus H(s'))$.

$$\begin{aligned} \mathrm{t} &= \mathrm{t'}\oplus H(s') \oplus H(s) \\ \mathrm{r} &= \mathrm{H(s)}\oplus t \\ &= \mathrm{H(s')}\oplus t' \\ &= \mathrm{r'} \\ \mathrm{z} &= \mathrm{G(r)}\oplus s \\ &= \mathrm{G(r)}\oplus s' \oplus (\delta 0^{k_1}) \\ &= (\mathrm{x'}\oplus\delta) \parallel 0^{k_1} \end{aligned}$$

Si l'attaquant déchiffre y à l'aide d'un oracle aléatoire, alors il obtient x.
Et il peut donc calculer x'.

Idées de preuve pour l'attaque de Shoup [Sho01]

Théorème

Il existe un oracle aléatoire tel qu'une permutation à sens-unique avec trappe existe.

La deux théorèmes précédents nous donnent le corollaire suivant :

Corollaire

Il existe un oracle aléatoire tel que la construction d'OAEP n'est pas sémantiquement sûr.

Réduction de RSA-OAEP

Proposition ([FOPS00])

1. *Le problème d'inverser partiellement la fonction RSA se réduit au problème de sécurité de RSA-OAEP en complexité (en temps) quadratique.*

Réduction de RSA-OAEP

Proposition ([FOPS00])

1. *Le problème d'inverser partiellement la fonction RSA se réduit au problème de sécurité de RSA-OAEP en complexité (en temps) quadratique.*
2. *Le problème d'inverser complètement la fonction RSA se réduit au problème d'inverser partiellement la fonction RSA en complexité (en temps) quadratique.*

Réduction de RSA-OAEP

Proposition ([FOPS00])

1. *Le problème d'inverser partiellement la fonction RSA se réduit au problème de sécurité de RSA-OAEP en complexité (en temps) quadratique.*
2. *Le problème d'inverser complètement la fonction RSA se réduit au problème d'inverser partiellement la fonction RSA en complexité (en temps) quadratique.*

Exemple

Pour une clé de 1024 bits, la complexité de la réduction est de 2^{40}.

Sommaire

Introduction

RSASSA-PSS

RSASSA-PSS est un protocole de signature et de vérification d'un message chiffré par une clé publique RSA .

Fonctionnement

Le Protocole se décompose en :

- un codage EMSA-PSS,
- une signature RSA en 3 parties.

EMSA-PSS

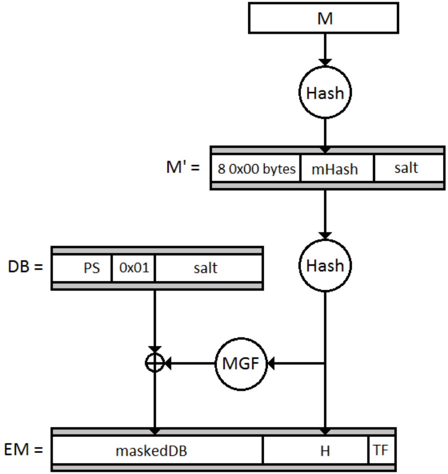


Figure – EMSA-PSS

Signature RSA

Elle se déroule en 3 parties à l'aide de 3 fonctions que nous allons expliquer

- OS2IP,
- RSASP1,
- I2OSP.

Vérification

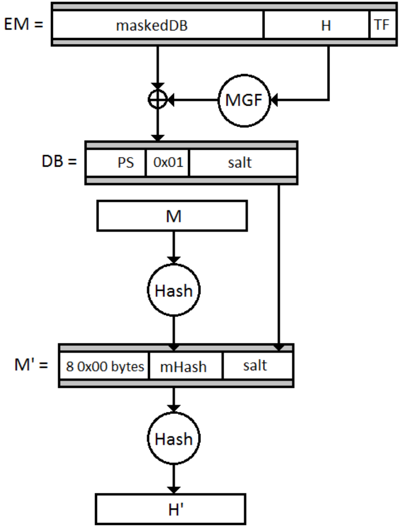
RSASSA-PSS-*Verify*

Elle prend en argument la clé public du signataire, le message signé, et sa signature.

Algorithme 5 RSASSA-PSS-*verify*

Require: Un message signé M et une clé public $K = (n, e)$ et une signature S .
Ensure: $SignatureValid$ ou $Signatureinvalid$.
  if si la taille de S n'est pas k octets **then**
    return $Signature\ invalide$
  end if
  $s =$ OS2IP(S)
  $m =$ RSAVP1((n,e),s) ;
  $EM =$ I2SOP(m , $emlen$) ou $emlem = (tailledenenbit$-1$)/8$
  if EMSA-PSS-$verify$(EM,M,$tailledenenbit$ -1) = "Consitent" **then**
    return $Signature\ Valide$
    return $Signature\ invalide$
  end if

EMSA-PSS-*verify*



Sécurité

La sécurité de ce schéma de Signature comparé autre schéma réside sur le fait qu'il soit probabiliste plutôt que déterministe grâce à la génération aléatoire de $Salt$.

Sommaire

Conclusion/Recommandation

Les deux protocoles ont une sécurité sémantique partielle.

OAEP Il est préférable de ne plus utiliser OAEP, et plutôt REACT ou d'autres protocoles de PKCS $\#1.5$,

PSS L'algorithme est encore utilisé.

Bibliographie

Hieu Phan Duong.
Securite et efficacite des schemas cryptographiques.
2010.

Eiichiro Fujisak, Tatsuaki Okamoto, David Pointcheval, and Jacques Stern.
Rsa-oaep is secure under the rsa assumption.
2000.

RSA Laboratory.
Pkcs 1 v2.2 : Rsa cryptography standard.
2000.

RSA Laboratory.
Rsaes-oaep encryption scheme.
2000.

Victor Shoup.
Oaep reconsidered.
2001.