Composition de Sécurité et Crypto - 2009

La composition est décomposée en deux parties :

- Une partie d'étude du Protocole EBICS
- Lecteur Pinpad Carte à Puce

Il est recommandé aux élèves de bien choisir l'ordre des parties selon leurs compétences et rapidités.

Le support de cours est permis. La majeure partie des questions font appel au sens pratique.

1. Etude du Protocole EBICS

Gestion des Clés:

a) L'utilisation d'EBICS en Allemagne est basée sur des clés que chaque client génère et déclare à sa banque par l'envoi par des fichiers de type INI et HIA et l'envoi d'une impression des diverses clés signées manuellement à la banque. Définir un avantage et un inconvénient de cette méthode (5 lignes maximum).

b) L'utilisation d'EBICS en France dans une première phase est basée sur des certificats générés par le client et auto-signés. Est-ce que cette utilisation (dans la première phase) est différente de la façon Allemande et pour quelle raison? (5 lignes maximum).

c) L'utilisation d'EBICS en France dans une seconde phase est basée sur des certificats générés par une autorité de certification. Les certificats sont déclarés par des fichiers de type INI et HIA. Définir un avantage et un inconvénient de l'utilisation des certificats (5 lignes maximum).

d) En France le CFONB (organisme de normalisation bancaire) a spécifié que les clés RSA doivent être de 2048 bits, et que pour chaque usage de sécurité il faut avoir une clé RSA et un certificat. Combien de clés et certificats faut-il par utilisateur en Ebics et quels sont ces clés et certificats?

e) En supposant que les utilisateurs possèdent des cartes à puce sur lesquels sont mis :
   - les clés RSA sous la forme de (S,N,P),
   - Les certificats sous leur forme binaire native,
   - Le certificat de l'autorité.

Question : définir le mapping de la carte à puce en indiquant la taille minimale de la carte et les droits d'utilisation.

f) Certains utilisateurs possèdent des clés et certificats dans le magasin microsoft et ils désirent les ré-utiliser avec les cartes à puce. Que proposez-vous pour permettre cette opération?

Authentification:

g) L'authentification utilise la signature XML/DSIG comme technique de sécurité. Les balises (tags) xml (et leurs balises-filles) entrant dans le calcul de l'XML/DSIG sont ceux ayant l'attribut authenticate=true. Pour quelle raison le contenu de la balise <DataTransfer> qui contient les données du fichier (dans la balise-fille <OrderData>) ne rentre pas dans le calcul de l'authentification ?

h) XML/DSIG utilise la transformation canonique. Quels sont les avantages de la transformation canonique dans une signature XML/DSIG?

Chiffrement:

i) Deux niveaux de chiffrement(pour la confidentialité) sont utilisés. https et le chiffrement du contenu de la balise <OrderData> par AES. Indiquez un avantage et un inconvénient du chiffrement du contenu de la balise <OrderData>. (3 lignes maximum).

j) En supposant que :
   - la vitesse de chiffrement et de 10 Moctets/s,
   - que le temps de dézippage est au maximum de 5 secondes (Pour rappel un fichier est chiffré après zippage),
   - Que le taux de compression des fichiers transmis est de 75%,
   - et que les temps de décodages Base64 sont négligeables,
   - Que le time-out d'une session https est de 30secondes,
   - Que le déchiffrement se fait sur le fichier final reçu après assemblage des divers segments de données de 1 Moctets (dans les phases Transfer)

   Question : quelle est la taille maximale d'un fichier qui peut être transmis et quel est le nombre de segments nécessaires pour transmettre ce fichier?

k) Que proposez-vous pour dépasser la taille maximale de la question précédente ?

l) Comme le fichier a été chiffré en mode CBC, qu'est ce qu'il faut sauvegarder entre la réception de deux segments successifs pour permettre le dépassement de la taille maximale ?

Signature:

m) La signature A006 est basée sur l'utilisation du RSA en mode PSS. Or les spécifications indiquent que la signature n'est pas appliquée directement sur le hash du fichier mais sur le hash du hash du fichier. En vous basant sur la caractéristique la plus importante d'une fonction hash en général, sous quelle condition cette méthode A006 peut affaiblir la signature PSS ?

n) Les signatures du fichier sont transmises dans la phase d'Initialisation avec les identités des signataires (attribut UserId) dans la balise <UserSignatureData>. Est-ce que les signatures peuvent être vérifiées en se basant uniquement sur cette balise ?

o) La balise <Prevalidation> contient le hash du fichier et est transmise lors de la phase d'Initialisation. Comment peut être utilisé le contenu de cette balise pour faire une pré-vérification des signatures ?

p) Est-ce que un résultat positif de la pré-vérification précédente est suffisant ?

q) La phase d'Initialisation contient les balises <OrderType> et <PartnerId>. Citez deux vérifications qui peuvent être implémentées en utilisant ces deux balisent éventuellement avec d'autres balises ou attributs précités.

r) A quelle phase du transfert un fichier peut être considéré comme bien signé ?

Gestion des Erreurs :

s) Quels sont les codes erreurs EBICS pour les cas suivants :
- Mauvaise signature,
- Dépassement de la taille maximale de fichier en déchiffrement trouvée dans le paragraphe Chiffrement (si le serveur ne supporte pas le dépassement),
- Erreur Allocation Mémoire sur le serveur,
- En supposant que dans la phase d'Initialisation d'envoi d'un fichier le client envoie la requête avec <NumSegments>5<NumSegments>. Le serveur reçoit un segment de données avec <SegmentNumber lastSegment= ''true''>4</SegmentNumber>.

2. Lecteur PINPAD Carte à Puce

L'accès à la carte se fait par un lecteur PINPAD (à clavier intégré). La présentation du PIN se fait par un ordre transparent de classe 0xA0 avec instruction=0x20 et le buffer de l'ordre transparent à envoyer est formé de deux zones de syntaxe : T,L,V où T est le type, L (un octet) est la longueur de la valeur V.

1. Zone Formatage de PIN. T=0x52
   - La zone valeur a les octets suivants :
     - Contrôle Byte :

       bit1-bit0 : PIN coding
       00 = BCD (Digit sur 4 bits)
       01 = characters ASCII
       10 = Format 2 PIN Block
       11 = RFU

       bit3-bit2 : 00 = usage future

       bit7-bit4 : length of PIN to be presented on the PIPAD

     - Insertion position Byte: This position is used by the reader to insert the entered PIN into the data to the smart card. Count start with 1 (i.e. insertion position 1 is the first byte)

     - Octets PIN à envoyer à la carte par le lecteur: Le PIN saisi au clavier sera inséré à la position indiquée par l'Insertion Position Byte. **Faire attention au format du fichier PIN.**

2. Zone temporisation de saisie. T=0x80.
   - La zone valeur est constituée d'un octet qui indique le temps en secondes.

Sachant que le fichier PIN contient 8 octets formats de la façon suivante:
0xFF,0xFF, 0xFF,0xFF, $P_1P_2P_3P_4$ avec $P_1P_2P_3P_4$ les codes ASCII du PIN de la carte

Définir la requête transparente à envoyer au lecteur pour la saisie sécurisée du PIN.