université de BORDEAUX

Master *Sciences Technologies*

Année 2020-2021
Étape : Master Sciences Technologies
Épreuve de : Sécurité des réseaux
Date : 15 Décembre 2020
Heure : 14h

Session de Décembre 2020
UE : 4TIN911U
Durée : 1h
Documents documents authorisés
Épreuve de Monsieur Guermouche

Questions générales

1. Expliquer brièvement ce qu'est une clé PGP et donner quelques exemples d'utilisation de ce genre d'objets. En quoi une clé PGP est différente d'un certificat X509 ?
2. Dans un contexte à mémoire limitée, donner deux approches pour contrer des attaques de type rejeu dans des protocoles qui y sont sensibles.
3. En quoi WPA3 améliore-t-il la sécurité des réseaux sans-fil ?
4. En quoi *Certificate Transparency* règle-t-il les problèmes inhérents aux certificats X509 ? Quels sont les reproches qui pourraient être faits à cette solution ? Quelle alternative vous parait plus pertinente ?
5. Expliquer l'établissemnent d'une connexion IPSec. Qu'est ce qui permet d'assurer la *Perfect Forward Secrecy* ?

Exercice 1

Une entreprise souhaite déchiffrer et analyser le trafic `HTTPS` transitant par sa passerelle. Pour ce faire, elle crée un couple certificat/clé privée correspondant à une autorité de certification (CA) et installe le certificat de cette dernière sur le poste personnel de chaque employé de telle sorte que les navigateurs internet de chaque poste fassent confiance aux certificats émis par cette CA. La passerelle quant à elle aura accès à la clé privée de l'autorité de certification.

1. Expliquer comment la passerelle peut écouter (en clair) le trafic `HTTPS` correspondant à une connexion établie par un navigateur se trouvant à l'intérieur de l'entreprise et se connectant à l'extérieur. Il vous est demandé de bien insister sur ce que fait la passerelle à chaque étape du protocole HTTPS.
2. Est ce que l'employé (ou plus exactement le navigateur) peut se rendre compte que sa connexion `HTTPS` est en train d'être écoutée ? Donner au moins une technique qui permettrait au navigateur de se rendre compte de l'écoute.
3. Dans votre réponse à la question 1 la passerelle utilise la clé privée de l'autorité de certification pour générer un certificat pour le domaine distant vers lequel la connexion est tentée (`banque.fr` par exemple). Supposons que la passerelle génère ce certificat en copiant tel quel le contenu du certificat reçu, les seuls champs à être modifiés étant le nom de la CA, la clé publique et la signature. La passerelle ne fait donc aucune vérification sur le certificat en provenance de `banque.fr` : La vérification du certificat est donc déléguée au navigateur du poste personnel. Expliquer en quoi ce scénario expose l'employé à une attaque de type *man-in-the-middle* effectuée par quelqu'un se trouvant en dehors de l'entreprise. Décrire l'attaque.

Exercice 2

Let's Encrypt est une autorité de certification populaire qui fournit des certificats X.509 pour des sites Web utilisant HTTPS. *Let's Encrypt* est différente de beaucoup d'autres autorités parce qu'elle utilise un algorithme automatique pour valider qu'un utilisateur possède bien le domaine pour lequel il demande un certificat. *Let's Encrypt* valide la propriété du domaine en fournissant un jeton aléatoire que l'administrateur du serveur doit fournir, via HTTP, sur le site Web de son domaine à une adresse spécifiée par *Let's Encrypt* (par exemple, `http://domain.com/.well-known/letsencrypt`).

Pour être plus précis, le client (le demandeur) se connecte d'abord à *Let's Encrypt* et demande un jeton via son API. Il déploie ensuite le jeton sur le serveur HTTP du domaine de telle sorte qu'il soit accessible via le chemin convenu. Puis il demande à ce que *Let's Encrypt* se connecte au domaine et valide le jeton. Si *Let's Encrypt* peut récupérer avec succès le jeton, il émettra un certificat pour le domaine.

1. Un FAI malveillant tente d'acquérir un certificat illégitime pour google.com. Comment le FAI peut-il amener *Let's Encrypt* à lui générer un certificat Google ? Supposons que le client DNS de Let's Encrypt est correctement configuré et non vulnérable aux attaques par injection en aveugle. On supposera que le FAI ne se trouve pas sur le chemin entre google et *Let's Encrypt*.

2. Que peut faire *Let's Encrypt* pour essayer d'empêcher l'attaque que vous avez décrite dans la question précédente ?

   Indice : votre solution n'a pas besoin d'être infaillible, mais elle doit réduire les risques de succès de l'attaque et ne devrait pas augmenter considérablement le temps qu'il faut pour qu'un site obtienne un certificat.

3. Supposons que l'implémentation TCP utilisée par *Let's Encrypt* a un bug qui fait que tous les paquets SYN ont le même numéro de séquence initial. Est-ce qu'un attaquant peut exploiter ce bug pour acquérir un certificat pour un domaine qu'ils ne contrôle pas ? Justifiez votre réponse.

4. Pour réduire les coûts, *Let's Encrypt* envisage de louer de l'espace dans son datacenter à d'autres utilisateurs. Ces autres clients utiliseraient le même sous-réseau et les même routeurs, mais seraient isolés sur différents ports d'un switch moderne. En supposant que *Let's Encrypt* vérifie la présence de token HTTP à partir de ce datacenter, comment un attaquant dans le centre de données pourrait-il acquérir un certificat pour un domaine qu'il ne posséde pas ?

   Indice : Votre attaque doit utiliser le fait que *Let's Encrypt* et l'attaquant sont sur le même réseau.

Exercice 3

Lorsqu'un fichier F est téléchargé sur Dropbox depuis la machine de l'utilisateur, le client Dropbox calcule au préalable un hachage du fichier $H(F)$ et l'envoie au serveur Dropbox. Si le hachage correspond à un hachage connu du serveur (le serveur l'a déjà reçu du même utilisateur ou d'un autre), Dropbox considère que les deux fichiers sont identiques et donc ne déclenche pas le téléchargement du nouveau fichier depuis le client. Lorsque le client veut synchroniser (i.e télécharger) le fichier sur une autre machine, Dropbox va lui envoyer une copie du fichier qu'il a dans son système.

1. Supposons que la fonction de hachage H utilisée par Dropbox est telle que pour tout fichier F et malware M, il soit facile de trouver un suffixe S tel que $H(F) = H(M|S)$ où $|$ fait référence à la concaténation. Décrire comment un attaquant peut utiliser cette propriété pour propager facilement son malware M à plusieurs utilisateurs de Dropbox depuis sa propre machine. Considérer par exemple que l'attaquant a obtenu la copie d'un film très populaire avant sa sortie.

2. Quelle propriété devrait être assurée par la fonction H pour éviter le scénario de la question précédente.

3. Supposons que le tout dernier film (M) du studio `Marvel` est sur le point d'être diffusé dans les salles de cinéma. Le studio veut tester si le film a été piraté et téléchargé sur Dropbox. Expliquer comment le studio peut arriver à effectuer cette vérification en utilisant un simple client Dropbox sans avoir aucun accès privilégié aux serveurs de Dropbox.

4. La fonctionnalité de Dropbox qui a été utilisée pour répondre à la question précédente peut être utilisée pour explorer les fichiers qui sont stockés sur les serveurs Dropbox. Comment pouvez-vous modifier le fonctionnement de Dropbox pour solutionner le problème. Quelles répercutions cette solution peut-elle avoir sur la bande-passante utilisée coté serveurs Dropbox.