Commencé le	mardi 13 octobre 2020, 14:41
État	Terminé
Terminé le	mardi 13 octobre 2020, 15:23
Temps mis	42 min 18 s
Note	Pas encore évalué

Question **1**

Correct

Note de 2,00 sur 2,00

Indiquez quelles affirmations sont vraies concernant les attaques CSRF

Veuillez choisir au moins une réponse :

- [ ] a. elles proviennent d'une confusion entre code et données
- [ ] b. elles nécessitent la modification du cookie de session
- [ ] c. il s'agit d'actions effectuées à l'insu d'un utilisateur sans qu'il ait besoin de se connecter à une application web
- [ ] d. elles résultent de la présence d'un cheval de Troie dans le navigateur de la victime
- [x] e. elles sont déclenchées par l'envoi d'une requête depuis une lien web malveillant ✔

Votre réponse est correcte.

La réponse correcte est : elles sont déclenchées par l'envoi d'une requête depuis une lien web malveillant

Question **2**

Correct

Note de 2,00 sur 2,00

Indiquez quelles affirmations sont vraies concernant les vers parmi les phrases suivantes:

Veuillez choisir au moins une réponse :

- [ ] a. un ver ne peut compromettre une application web si le serveur exige qu'un client lui fournisse un jeton (token) secret
- [x] b. un ver est un logiciel malveillant qui se propage lui-même ✔
- [ ] c. un ver nécessite un botnet pour fonctionner
- [ ] d. un ver est une contremesure pour éviter les attaques XSS
- [ ] e. un ver est simplement un bug de programmation

Votre réponse est correcte.

La réponse correcte est : un ver est un logiciel malveillant qui se propage lui-même

Question **3**

Partiellement correct

Note de 1,33 sur 2,00

Une attaque CSRF peut être évitée par :

Veuillez choisir au moins une réponse :

- [x] a. l'utilisation de l'en-tête Origin dans les requêtes HTTP ✔
- [x] b. la validation d'un jeton (token) secret ✔
- [x] c. la connexion à un seul site simultanément dans le navigateur ✔
- [ ] d. des requêtes préparées
- [x] e. l'utilisation de cookies avec l'attribut Secure positionné ✖ non, ceci n'offre qu'une protection contre une attaque MITM
- [ ] f. l'utilisation de tout type de cookie

Votre réponse est partiellement correcte.
Vous avez sélectionné trop d'options.
Les réponses correctes sont : la validation d'un jeton (token) secret, l'utilisation de l'en-tête Origin dans les requêtes HTTP, la connexion à un seul site simultanément dans le navigateur

Question **4**

Partiellement correct

Note de 1,33 sur 2,00

Quelles affirmations suivantes pouvez-vous valider concernant les attaques XSS ?

Veuillez choisir au moins une réponse :

- [x] a. l'application peut être protégée côté client ✔
- [ ] b. désactiver Javascript supprime tout risque tout en conservant une navigation normale
- [x] c. l'application peut être protégée côté serveur ✔
- [ ] d. un jeton (token) secret n'a aucune utilité pour empêcher une attaque XSS

Votre réponse est partiellement correcte.
Vous en avez sélectionné correctement 2.
Les réponses correctes sont : un jeton (token) secret n'a aucune utilité pour empêcher une attaque XSS, l'application peut être protégée côté serveur, l'application peut être protégée côté client

Question **5**

Partiellement correct

Note de 1,33 sur 2,00

Quelles affirmations sur les cookies sont-elles vraies ?

Veuillez choisir au moins une réponse :

- [ ] a. un cookie ne peut être modifié que par un navigateur web ou un serveur web
- [ ] b. un cookie conserve une information concernant un utilisateur dans un serveur web
- [ ] c. un cookie peut prouver qu'un utilisateur s'est authentifié sur un site web
- [x] d. un cookie conserve une information concernant un utilisateur sur un navigateur ✔
- [x] e. un cookie peut compromettre la vie privée d'un utilisateur ✔

Votre réponse est partiellement correcte.

Vous en avez sélectionné correctement 2.
Les réponses correctes sont : un cookie conserve une information concernant un utilisateur sur un navigateur, un cookie peut compromettre la vie privée d'un utilisateur, un cookie peut prouver qu'un utilisateur s'est authentifié sur un site web

Question **6**

Terminer

Noté sur 6,00

Expliquez comment fonctionnent les attaques d'injection, notamment SQL et comment découvrir de telles attaques ? Illustrez avec des exemples.

Les attaques d'Injection SQL permettent de modifier une base de données SQL ou de récupérer des données en exploitent une mauvaise manière de coder la page web : le mélange du code et des données.

Par exemple, on peut avoir une requête SQL dans le code de la page WEB qui attend une entrée utilisateur. Ceci est dangereux, car une requête de la forme "WHERE name = *user_input* " peut être exploitée si le *user_input* est une requête SQL malveillante cherchant à exfiltrer des données.

Il y a 3 types d'Injections SQL, basées sur :

- les erreurs : on injecte des requêtes provoquant des erreurs et on peut en déduire des informations

- les UNION : on peut exfiltrer les données de plusieurs colonnes du tableau de données en une seule requête

- ou injections à l'aveugle : la page WEB ne retourne pas forcément de réponse, ainsi on peut exploiter d'autres informations comme la réaction à des réponses TRUE comme 1=1, FALSE, ou le temps de réponse

Ainsi, on peut vérifier si le code est sensible à de telles attaques en testant avec :

- ' (une *single quote*)

- ' ' (2 *single quote*)

- des caractères spéciaux ou numériques

Question **7**

Terminer

Noté sur 4,00

Quels sont les différents types d'attaques cross-site scripting et comment fonctionnent-elles ? Illustrez avec des exemples.

Il y a 3 types d'attaques XSS :

- Attaques réfléchies (*Reflected XSS*) : L'attaquant va envoyer un lien malveillant à sa victime. Ainsi, lorsqu'elle va exécuter une requête vers le serveur, la réponse du serveur sera renvoyée en écho à l'attaquant (elle est réfléchie)

- Attaques Stockées (*Stored XSS*) : l'attaquant utilise cette fois le serveur. Il va injecter un script directement dans le serveur, et lorsque l'utilisateur va interagir avec le site, il va ainsi activer le script de l'attaquant et lui envoyer des données

- *DOM-based XSS* : l'attaquant touche uniquement à l'interface de la victime. Sans toucher au serveur, il va uniquement modifier l'interface (par exemple, l'URL) pour récupérer des informations. Ainsi, seule l'interface va agir de manière différente et pas le serveur

Question **8**

Partiellement correct

Note de 1,33 sur 2,00

Concernant les injections SQL, quelles affirmations sont vraies dans la suite ?

Veuillez choisir au moins une réponse :

- [ ] a. une injection SQL, si elle est possible, peut parfois permettre d'obtenir un shell en tant que root.
- [ ] b. un outil comme sqlmap ne pourra trouver d'injections SQL si le serveur effectue un filtrage par blacklist (filtrage des caractères dangereux)
- [ ] c. un outil comme sqlmap ne pourra trouver d'injections SQL si le serveur effectue un filtrage par whitelist (entrées de formats prévus à l'avance)
- [x] d. une injection SQL, si elle est possible, peut causer la destruction de la base de données d'une application web ✔
- [x] e. une injection SQL, si elle est possible, peut provoquer une modification des données d'une application web ✔
- [ ] f. une injection SQL, si elle est possible, peut toujours conduire à l'exfiltration de l'ensemble des données d'une application web

Votre réponse est partiellement correcte.

Vous en avez sélectionné correctement 2.
Les réponses correctes sont : une injection SQL, si elle est possible, peut provoquer une modification des données d'une application web, une injection SQL, si elle est possible, peut causer la destruction de la base de données d'une application web, une injection SQL, si elle est possible, peut parfois permettre d'obtenir un shell en tant que root.

Question **9**

Terminer

Note de 4,00 sur 4,00

Expliquez ce qu'est la Same Origin Policy (SOP) et le Cross-Origin Resource Sharing (CORS) ?

On peut avoir différentes ressources ou domaines sur la même page WEB. Ainsi, SOP et CORS sont des restrictions permettant de réguler la communications entre eux.

Same Origin Policy : restreint la manière dont les domaines peuvent communiquer entre eux, et permet la communication entre ceux qui ont la même origine. Ainsi, on va considérer que l'origine est la même si on a le même serveur, même hôte et même port.

CORS : permet le partage des ressources selon certains critères même si les origines sont différentes

Commentaire :