École de technologie supérieure
Département de génie logiciel et des TI

Trimestre : Automne 2016
Professeur : Jean-Marc Robert
Chargé de cours : Marc-André Drapeau
Chargé de lab. : Moussa Kaba

GTI 619 - Sécurité des systèmes

LABORATOIRE 5

Module d'authentification

L'authentification est le processus par lequel l'identité d'un utilisateur qui accède à un système ou à un réseau est vérifiée. Il permet notamment de valider qu'un utilisateur est bien la personne qu'il prétend être. L'authentification se fait communément en utilisant un code d'utilisateur et un mot de passe que l'utilisateur est le seul à connaître. Ceux-ci seront transmis à un tiers possédant la liste détaillée des utilisateurs (identifiant, nom, prénom, login, mot de passe, adresse courriel, code permanent, …) qui pourra, avec ces informations, authentifier l'utilisateur. Ce tiers se doit d'être sécurisé, et la communication avec ce dernier, chiffrée. Les mots de passe doivent être conservés secrets, y compris du tiers de confiance (généralement par une fonction de hachage).

La gestion d'une session est le processus par lequel un serveur maintient l'état d'une communication avec un tiers. Il est nécessaire pour le serveur de retenir certaines informations lui permettant de savoir comment réagir aux différentes requêtes (par exemple, les cookies d'une application web). La session est identifiée par un numéro qui est transmis entre le client et le serveur lors des requêtes. Cet identifiant se doit d'être unique pour chaque communication établie entre le serveur et un tiers. De plus, il doit être aléatoire afin d'éviter le vol d'une session en cours et expirer suite à une certaine période d'inactivité. Finalement, les informations dans une session partagée entre un client et un serveur doivent être protégées afin d'éviter toute tentative de piratage.

Pour ce laboratoire, vous devrez implémenter des éléments de sécurité (voir ci-dessous) dans une application que vous développerez en Java, PHP, famille Asp de Microsoft(c#, vb, etc), JS ou Python selon votre choix pour le Back-End. Si vous souhaitez utiliser un langage qui ne se trouve pas dans cette liste, veuillez d'abord valider votre choix avec votre chargé de laboratoire. Cette application peut être ou bien locale ou de type client-serveur.

Il est possible d'utiliser un cadriciel (*Framework*) lors du développement de votre application, y compris pour les fonctions de sécurité, mais vous devez avoir une connaissance suffisante de celui-ci et de vos outils pour certaines explications dans votre laboratoire.

Les Frameworks à utiliser doivent être de type MVC utilisant les langages cités auparavant. L'utilisation des CMS (wordpress, joomla etc) n'est pas acceptée. Si vous n'êtes pas certains de votre choix, veuillez contacter le chargé de laboratoire.

Partie 1 : implémentation d'une gestion d'accès

Cette application devra implémenter une gestion d'accès RBAC (« *Role Base Access Control* »). Ce modèle de gestion d'accès devra contenir les trois rôles suivants :

- Administrateur
- Préposé au cercle
- Préposé au carré

Vous devez créer les trois utilisateurs suivants :

- « Administrateur », qui possèdera le rôle « Administrateur ».
- « Utilisateur1 », qui possèdera le rôle « Préposé au cercle ».
- « Utilisateur2 », qui possèdera le rôle « Préposé au carré ».

Cette application devra également contenir les éléments suivants :

- Une page permettant à l'utilisateur de s'authentifier.
- Une ou des pages d'administration pour configurer les options de sécurité. Les options de sécurité devant être paramétrables sont indiquées dans la partie 2 ci-dessous, sous la colonne « Paramétrable » du tableau. Cette ou ces page(s) devront être visibles seulement pour le rôle « Administrateur »
- Une page qui affiche un cercle, cette page devra être visible par les rôles « Préposé au cercle » et « Administrateur »
- Une page qui affiche un carré, cette page devra être visible par les rôles « Préposé au carré » et « Administrateur »

Partie 2 : implémentation de l'authentification

L'implémentation de votre authentification devra répondre aux objectifs suivants :

Objectif de contrôle	**Description de la mise en œuvre**	**Paramétrable**
Protocole de communication et d'échange d'information.	Selon votre application, les informations d'authentification doivent être protégées.	Non
Une protection contre la force brute. Empêcher qu'une personne tente des milliers de tentatives de connexion dans un court laps de temps. Attention aux dénis de service	Délai par utilisateur en fonction du nombre de tentatives. • Nombre de tentatives maximales. • Délais d'authentification en cas d'atteinte du nombre maximal de tentatives. • Blocage d'accès suite à une deuxième atteinte du nombre maximal de tentatives: nécessité de rejoindre l'administrateur pour définir un nouveau mot de passe.	Oui

Objectif de contrôle	Description de la mise en œuvre	Paramétrable
Une gestion du mot de passe.	Changement de mot de passe : périodiquement ou suite à un événement (oubli, dépassement de la limite de tentatives …)	Oui
Une politique de mot de passe configurable	Complexité du mot de passe (longueur, composé d'au moins une minuscule et une majuscule, d'un caractère spécial et d'un chiffre, etc.) Impossibilité d'utiliser un ancien mot de passe parmi les x derniers.	Oui
Stocker le mot de passe de façon sécuritaire.	Un salt par utilisateur. Complexité : plusieurs passes de hachage, etc. Une « colonne » pour la version de la fonction de hachage.	Non
Journalisation des connexions, des changements reliés à la sécurité.	Connexion réussie et non réussie. Modification du mot de passe.	Non
Réauthentification pour certaines fonctions critiques.	• Redemander le mot de passe pour pouvoir changer le mot de passe (une fois connecté). • Ajout d'un utilisateur. • Etc.	Non

Partie 3 : Gestion de la session

Objectif de contrôle	Description de la mise en œuvre
Protection de l'identifiant de session	Les identifiants de la session doivent être protégés afin d'éviter toute tentative de la régénérer et de la rejouer.
Délais d'inactivité	Selon le niveau de protection, la session devient inactive et l'utilisateur doit se réauthentifier. Par exemple un délai de 20 minutes pour un niveau de protection élevé.
Stockage sécurisé des informations de la session	Pour chaque utilisateur, les informations de la session doivent être sauvegardées dans un endroit sécuritaire. Certains champs de la session doivent être hachés, car ils contiennent des informations confidentielles.

Partie 4 : Authentification forte

Une authentification forte peut être implémentée. Une préconisation de mise en œuvre peut être l'utilisation d'une « GRID CARD ». La réussite de ce défi vous vaudra 5 points sur le rapport du laboratoire.

Informations sur le laboratoire

- Quatre séances sont dédiées à ce laboratoire.
- L'évaluation de l'application se fera pendant la dernière séance de laboratoire.
- Le rapport de laboratoire devra être :
  - remis **à la fin de la journée de démonstation** (sous forme électronique). Le code devrait être aussi envoyé avec votre rapport dans une archive .zip, .rar ou .tar.* (à ne pas mettre en annexe, il faut le code source comme vous l'avez compilé). Un fichier expliquant les configurations et les étapes d'exécution de votre projet doit être inclus dans la remise. Ces explications doivent permettre au chargé de laboratoire d'exécuter votre projet. Vous devez commenter les entêtes des méthodes de vos classes (description, variables d'entrée, ainsi que la sortie).
  - concis et ne pas dépasser 18 pages (max. 1 page pour l'introduction et 1 page pour la conclusion).**Le corps du rapport doit contenir les diagrammes de conception (minimalement votre diagramme de classes et de séquences), votre architecture ainsi que vos choix de design et d'implémentation**.
- Pour la remise du code source, vous pouvez partager votre entrepôt de code avec le chargé de laboratoire.
- Vous devez inclure une copie imprimée de la grille d'évaluation avec le numéro d'équipe, vos noms et vos codes permanents.
- Il y a aura, tel que prévu dans les règles du département, une pénalité de 10% par jour de retard.
- Jusqu'à 10% de la note du rapport pourra être retiré pour la qualité du français (-0.1 points par faute), et jusqu'à 5% pour le non-respect des normes indiquées dans le document de remise.
- **Les éléments exigés dans le rapport sont les suivants** :
  1. Une introduction mettant votre application en contexte, expliquant le but du travail et annonçant le plan.
  2. Une architecture à haut niveau de votre système complet
  3. Une section sur la gestion des accès, qui présentera comment les accès et les rôles sont gérés dans votre application. Incluez un diagramme de classe représentant les classes qui interviennent dans la gestion des accès et indiquez comment votre application gère les accès. Incluez une discussion sur les points forts et les points faibles de votre approche et les améliorations possibles à celle-ci.
  4. Une section sur l'authentification, qui présentera comment l'application gère l'authentification des utilisateurs. Dans cette section, il sera nécessaire de fournir un diagramme de séquence

représentant le processus d'authentification avec les classes impliquées. Aussi, incluez l'algorithme de hachage utilisé vu à un niveau élevé et discutez, comme au point précédent, de votre méthode, de ses points forts, de ses points faibles et des améliorations possibles.

5. Une section sur la gestion des sessions, dans laquelle vous expliquerez comment les sessions sont gérées dans votre application et comment celles-ci sont protégées des interventions extérieures. Discutez, encore une fois, des points forts, des points faibles et des améliorations possibles de votre façon de faire.
6. Une conclusion, qui fera un retour sur les points abordés précédemment et sur vos résultats vis-à-vis les exigences du laboratoire.

Pondération

• Introduction	5%
• Architecture à haut niveau	5%
• Partie 1 – Gestion de l'accès	20%
• Partie 2 – Authentification	20%
• Partie 3 – Session	10%
• Conclusions	5%
• Bon fonctionnement (démonstration)	30%
• Authentification forte	5%

Références utiles :

- https://www.owasp.org/index.php/Authentication_Cheat_Sheet
- http://machinesaredigging.com/2013/10/29/how-does-a-web-session-work/
- http://www.bestwebframeworks.com/

Équipe : No #			
Noms des équipiers : ○ ○ ○ ○		Codes permanents : ○ ○ ○ ○	
Questions	**Grille d'évaluation de la démonstration**	**Points (/30)**	
Partie I :	Page d'authentification fonctionnelle.	/1	/ 7
	Page Administrateur (seulement) pour configuration des paramètres de sécurité.	/1	
	Page qui affiche un cercle : cette page devra être visible par les rôles «Préposé au cercle» et «Administrateur»	/1	
	Page qui affiche un carré : cette page devra être visible par les rôles «Préposé au carré» et «Administrateur»	/2	
	Création des trois utilisateurs : Administrateur, cercles et carré. Possibilité d'ajouter des utilisateurs et de les associer à un rôle.	/2	
Partie II :	Protocole de communication et d'échange d'informations sécurisé.	/2	/15
	Nombre de tentatives maximales.	/2	
	Délai de blocage d'authentification en cas d'atteinte du nombre maximal de tentatives.	/1	
	Blocage d'accès suite au dépassement d'une limite de tentatives : Nécessite de rejoindre l'administrateur pour définir un mot de passe différent.	/1	
	Changement de mot de passe : périodiquement ou suite à un événement (oubli, dépassement de la limite de tentatives …)	/1	
	Complexité du mot de passe (longueur, composé d'au moins une minuscule et une majuscule, d'un caractère spécial et d'un chiffre, etc.).	/1	
	Impossibilité d'utiliser un ancien mot de passe parmi les x derniers.	/1	
	Stocker le mot de passe de façon sécuritaire (salt, plusieurs itération, version de la fonction de hachage).	/2	
	Journalisation des connexions, des changements reliés à la sécurité (p. ex. mot de passe).	/2	
	Réauthentification pour certaines fonctions critiques (changement de mot de passe, par exemple).	/2	
Partie III :	Les identifiants de la session doivent être protégés afin éviter toute tentative de la régénérer.	/2	/ 6
	Selon le niveau de protection, la session devient inactive et l'utilisateur doit se ré-authentifier. Par exemple un délai de 20 minutes pour un niveau de protection élevé.	/2	
	Stockage sécurisé des informations de la session.	/2	
Authentification forte :			/2