# STI – Analyse de menace

1. Description du système
   1. Objectifs du système

* Permettre aux utilisateurs de communiquer par message
* Réputation : les messages doivent arriver au destinataire voulu et être visibles uniquement par le/les destinataires.
  1. Hypothèses de sécurité
* Réseau interne et administrateurs de confiance
* Système d'exploitation et serveur Web de confiance
  1. Exigences de sécurité :
* Le site Web doit être disponible à 99% du temps (disponibilité)
* Les informations des utilisateurs doivent être scrupuleusement protégées (privacy)
* Les informations accédées par les utilisateurs ne doivent pas pouvoir être tracées (privacy)
* Seul un administrateur doit pouvoir ajouter/modifier/supprimer un utilisateur
* Les mails reçus par un utilisateur ne doivent être visibles que pour lui (privacy)
  1. Eléments du système :
* Base de données des utilisateurs
* Base de données des messages
* Application Web
  1. Rôles des utilisateurs :
* Collaborateur
* Administrateur (même droits que collaborateurs + options administrateur)
* Webmasters (gestion du site)
  1. Actifs à haute valeur :
* Base de données des utilisateurs (données)
  + Confidentialité, sphère privée
  + Un incident nuirait à la réputation du site
  + Plainte possible et perte financière
* Base de données des messages (données)
  + Confidentialité (messages privés entre collaborateur)
  + Intégrité (une modification impliquerait une baisse de réputation)
* Infrastructure
  + Intégrité, disponibilité
  + Un incident serait critique et nuirait à la disponibilité/réputation
  1. DFD

Une image contenant texte, carte, ciel, divers

Description générée automatiquement

1. Identification des sources de menaces
   1. Sources de menace

* Hackers, script-kiddies
  + Motivation : s'amuser, gloire
  + Cible : n'importe quel élément / actif
  + Potentialité : haute
* Cybercrime (spam, maliciels)
  + Motivation : récupération d’information / nuire aux utilisateurs
  + Cible : vol de credentials des utilisateurs, spam des utilisateur, modification / récupération d'informations
  + Potentialité : haute
* Concurrent
  + Motivation : détruire la réputation de l’entreprise, espionnage industriel.
  + Cible : base de données des utilisateurs / base de données des messages
  + Potentialité : Moyenne
* Employé malicieux
  + Motivation : élévation de privilèges, lecture de messages de collaborateurs
  + Cible : Base de données des messages, compte administrateur
  + Potentialité : faible

1. Identification des scénarios d’attaque
   1. Scénario 1 : Vol de compte utilisateur

* Business impact : Haut (réputation, acquisition d’informations confidentielles, usurpation d’identité)
* Sources de menace : Toutes
* Motivations : espionnage industriel (accès aux messages), élévation de privilèges
* Biens ciblés : Base de données des utilisateurs
* Scénarios d’attaque : Mots de passe faible, session hijacking, brute force
  1. Scénario 2 : Attaque de type injection SQL
* Business impact : Haut (réputation, acquisition d’informations confidentielles, suppression/modification de compte utilisateurs et mots de passe)
* Sources de menace : Hacker, cybercrime
* Motivations : Rendre le site indisponible, espionnage industriel (accès aux messages)
* Biens ciblés : base de données des utilisateurs et des messages.
* Scénarios d’attaque : injection SQL
  1. Scénario 3 : Attaque de type XSS
* Business impact : Moyen (réputation, récupération du contenu de cookies)
* Sources de menace : Toutes
* Motivations : exécuter un script chez la victime.
* Biens ciblés : navigateur de la victime.
* Scénario d’attaque : injection de script dans un message. Lors de son affichage, le script va s’exécuter.
  1. Scénario 4 : Attaque de type CSRF
* Business impact : Moyen (récupération du contenu de cookies, action sur un site choisis l’attaquant).
* Source de menace : Hacker, cybercrime
* Motivations : Faire effectuer à la victime une action non désirée sur un site
* Biens ciblés : Navigateur de la victime
* Scénario d’attaque : Dissimulation d’une requête envers un autre site en la faisant passer pour une requête innocente

1. Mitigation des risques :
   1. Scénario 1 : Vol de compte utilisateur

* Mettre en place des mots de passe fort (la politique appliquée ici est un mot de passe avec au minimum 15 caractères. Cela peu sembler faible mais, même si uniquement des lettres minuscules sont utilisées, cela représente 1,677,259,342,285,726,023,680 combinaisons possibles. Et cela permet à un utilisateur de créer un mot de passe qu’il peut facilement mémoriser et qu’il n’a pas besoin de noter.)
* Utiliser le protocole HTTPS pour éviter le vol de cookies.
  1. Scénario 2 : Attaque de type injection SQL
* Utiliser PDO pour faire des requêtes paramétrées. Cela permet à la base de données de pouvoir faire la différence entre le code SQL et les données entrées par l’utilisateur.
  1. Scénario 3 : Attaque de type XSS
* Supprimer les caractères suspects lors de l’envoi de message. La fonction strip\_tags[[1]](#footnote-1) peut être utilisée en PHP.
  1. Scénario 4 : Attaque de type CSRF
* Utiliser des tokens anti CSRF généré aléatoirement pour empêcher la création de requête valide par un attaquant

1. php.net/manual/en/function.strip-tags.php [↑](#footnote-ref-1)