Communications Security Establishment
Centre de la sécurité des télécommunications

CANADIAN CENTRE FOR CYBER SECURITY

Facteurs à considérer en matière de cybersécurité pour votre site Web

GESTIONNAIRES

TLP:WHITE

Canada

AVANT-PROPOS

La présente publication intitulée *Facteurs à considérer en matière de cybersécurité pour votre site Web (ITSM.60.005)* est un document NON CLASSIFIÉ publié avec l'autorisation du dirigeant principal du Centre canadien pour la cybersécurité (CCC).

Pour de plus amples renseignements, prière de communiquer avec notre équipe des Services à la clientèle du Centre :

Centre d'appels
contact@cyber.gc.ca
613 949-7048 or 1-833-CYBER-88

DATE D'ENTRÉE EN VIGUEUR

Le présent document entre en vigueur le 6 octobre 2021.

HISTORIQUE DES RÉVISIONS

Révision	Modifications	Date
1	Première version.	6 octobre 2021

ISBN 978-0-660-40570-4
CAT D97-4/60-005-2021F-PDF

APERÇU

Le site Web de votre organisation est un élément crucial de vos activités. Il donne accès à vos services et assure la visibilité de vos produits. Toutefois, les cybermenaces peuvent compromettre votre site Web, ce qui peut nuire à vos fonctions commerciales, à vos revenus et à votre réputation.

Les sites Web font office de porte d'accès à votre organisation par l'entremise d'Internet. Les auteurs de menace peuvent exploiter les vulnérabilités et les mauvaises configurations pour voler, modifier ou supprimer de l'information sensible (p. ex. les portails des fournisseurs, les données des clients, les clients potentiels, les renseignements opérationnels et financiers). Les atteintes à la protection des données vous exposent à des risques en matières légale et réglementaire, et possiblement à des pénalités financières. De plus, si votre site Web est compromis, un auteur de menace pourrait être en mesure de cibler d'autres organisations et personnes avec lesquelles vous êtes affilié.

Le présent document vous présente des pratiques exemplaires en matière de cybersécurité que votre organisation devrait intégrer à la conception et à l'entretien de son site Web. La section 2 comporte une liste de vérification des principales mesures que vos décideurs informatiques et gestionnaires d'équipe de développement Web peuvent mettre en œuvre pour améliorer la sécurité du site Web de votre organisation. Les sections 3 à 9 fournissent davantage de détails sur les facteurs à considérer en matière de cybersécurité se trouvant sur la liste de vérification.

Rappelez-vous cependant que même si vous abordez tous les points de la liste de vérification, il est impossible d'éliminer entièrement les risques (p. ex. il reste un risque résiduel). Les mesures de sécurité que vous prenez dépendent de l'environnement, des exigences et des ressources de votre organisation. Vous devez consulter et adapter l'orientation fournie dans ce document afin de répondre aux besoins et au niveau de tolérance au risque acceptable de votre organisation.

TABLE DES MATIÈRES

LISTE DES FIGURES

LISTE DES TABLEAUX

1 INTRODUCTION

Le présent document vous présente des pratiques exemplaires en matière de cybersécurité que votre organisation devrait intégrer à la conception et à l'entretien de son site Web. La section 2 comporte une liste de vérification des principales mesures que vos décideurs informatiques et gestionnaires d'équipe de développement Web peuvent mettre en œuvre pour améliorer la sécurité du site Web. Vous pouvez utiliser cette liste de vérification de plusieurs façons :

- **Décidez s'il est préférable de construire ou d'acheter** en déterminant si votre organisation a la capacité de construire son site Web et en identifiant les composants devant être externalisés;
- **Concevez les services de votre site Web** en incorporant les contrôles de sécurité appropriés;
- **Passez en revue votre site Web opérationnel** pour cibler les failles de sécurité;
- **Procurez des services Web** provenant de fournisseurs qui utilisent des contrôles de sécurité appropriés pour protéger vos services Web;
- **Définissez des rôles et des responsabilités que tous comprennent** pour assurer la sécurité de votre site Web.

Pour faire suite à la liste de vérification, le présent document décrit plus en détail chacun des facteurs à considérer en matière de cybersécurité. Que vous optiez pour le développement de votre propre site Web ou l'achat de services d'hébergement Web, vous devez veiller à ce que des contrôles soient en place pour protéger la confidentialité, l'intégrité et la disponibilité des données de votre site Web. Votre organisation est légalement responsable de protéger toutes les données des clients qui sont traitées sur votre site Web, même si votre organisation n'héberge pas le site Web.

Beaucoup d'entreprises canadiennes utilisent des services gérés pour leurs opérations. Si vous optez pour un fournisseur de services d'informatique en nuage (FSI) ou un fournisseur de services gérés (FSG) pour vous aider avec vos services Web, nous vous recommandons fortement de consulter les publications suivantes : *Facteurs à considérer par les clients de services gérés en matière de cybersécurité (ITSM.50.030)* [1][1] et *Gestion des risques liés à la sécurité infonuagique (ITSM.50.062)* [2].

1.1 SENSIBILITÉ DES DONNÉES

Avant de concevoir votre site Web, vous devez déterminer les données de grande valeur et sensibles (p. ex. des renseignements personnels, financiers, essentiels aux activités ou exclusifs) pour vous assurer de mettre en œuvre dans le cadre de la conception et de l'architecture de votre site Web les mesures nécessaires pour les protéger. La sensibilité des données se mesure en faisant une évaluation qui tient compte du préjudice possible qu'une incapacité de protéger la confidentialité, l'intégrité et la disponibilité des renseignements pourrait occasionner. La confidentialité permet de protéger les renseignements contre toute divulgation non autorisée. L'intégrité permet de protéger contre toute modification non autorisée. La disponibilité permet d'assurer que les renseignements sont disponibles au besoin.

Lorsque vous classez vos données par catégories, attribuez des niveaux de sensibilité à chacun de ces trois secteurs (confidentialité, intégrité, disponibilité) pour créer un profil de sécurité des données en trois parties. La sensibilité des données est classée comme étant élevée (E), moyenne (M) ou faible (F). Par exemple, envisagez un ensemble de données

[1] Les numéros entre crochets renvoient à des ressources figurant à la section *Contenu complémentaire* du présent document.

qui possède un profil de sécurité des données : E/M/F. Vous pouvez interpréter les trois parties du profil de sécurité des données de manières suivantes :

- Une **compromission de confidentialité des données** (E) a une incidence profonde ou un impact prohibitif;
- Une **compromission d'intégrité des données** (M) a des répercussions importantes;
- Une **compromission de disponibilité des données** (F) a un impact modéré.

La classification des données est une tâche qui peut s'avérer fastidieuse. Par exemple, la sensibilité des données liées aux résultats d'une élection est plus élevée le jour de l'élection que le lendemain de celle-ci. Vous devez classifier vos données en fonction du niveau le plus élevé de préjudice qui pourrait se produire si ces données étaient compromises. Voir l'Annexe 1 du document *La gestion des risques liés à la sécurité des TI : Une méthode axée sur le cycle de vie (ITSG-33)* [3] pour obtenir de plus amples renseignements sur la classification et l'établissement de catégories pour vos données.

Groupe de données	Confidentialité	Intégrité	Disponibilité	Qui a besoin d'accéder	Lieu de stockage
Groupe de données A	Élevée	Élevée	Élevée	Administrateurs de système	Réseau interne
Groupe de données B	Moyenne	Élevée	Moyenne	Finance	Fournisseur de services
Groupe de données C	Moyenne	Faible	Faible	Utilisateurs Invités	Fournisseur de système

Figure 1: Échantillon de profil de sécurité des données

2 LISTE DE VÉRIFICATION DES FACTEURS À CONSIDÉRER EN MATIÈRE DE CYBERSÉCURITÉ

La liste de vérification suivante résume les mesures que vous devriez mettre en œuvre lors du développement et de l'entretien du site Web de votre organisation. À la phase de mise en œuvre, ces mesures fonctionnent ensemble pour s'attaquer aux risques communs d'applications Web, qui sont définis dans le projet intitulé *Top 10 Web Application Security Risks* de l'organisation Open Web Application Security Project (OWASP) [4]. Nous décrivons plus en détail chacune de ces mesures de sécurité tout au long du présent document.

Table 1: Liste de vérification pour concevoir ou fournir des services Web sécuritaires

Section	Mesures de sécurité	Traitée (T), En développement (D), Sans objet (S/O)	Remarques
Architecture sécuritaire			
3.1	Séparer des composants des services Web : • Isoler le serveur Web du réseau • Isoler le serveur d'applications et la base de données		
3.2	Protéger les services Web connectés à Internet : • Installer un serveur mandataire • Utiliser un coupe-feu WAF		
3.3	Mettre en œuvre de composants redondants		
3.4	Isoler les interfaces administratives		
Contrôle d'accès			
4.1	Planifier et tester les contrôles d'accès		
Mécanismes d'authentification			
5.1	Utiliser une politique en matière de mots de passe robustes		
5.2	Permettre une authentification multifacteur		
5.3	Mettre en place un processus sécuritaire de récupération de compte		
5.4	Utiliser des mesures pour protéger les mots de passe, comme : • Utiliser le protocole HTTPS • Utiliser des fonctions de hachage et de salage de mots de passe • Envisager des options d'entrée de mots de passe		

	• N'effectuer aucun codage en dur de justificatifs de base de données et de clés d'interface de programmation d'applications		
5.5	Utiliser des blocages de compte, des délais de connexion et des tests de Turing appelés CAPTCHA (pour *Completely Automated Public Turing Tests to Tell Computers and Humans Apart*) pour différencier les ordinateurs des humains afin d'éviter les attaques par force brute		
Session sécuritaire			
6.1	Utiliser des boîtes à outils de gestion de session		
6.2	Créer des identifiants de session aléatoires qui respectent une longueur minimale		
6.2	Utiliser des indicateurs de sécurité sur les témoins		
6.3	Stocker les données de témoin sur un serveur		
6.4	Mettre un terme aux données de session		
6.6	Ajouter des couches d'authentification de session (jetons CSRF [pour *anti-cross-site request forgery*], réauthentification pour opérations à risque plus élevé)		
Validation d'entrée			
7.1	Valider tôt, mais coordonner		
7.2	Confirmer les longueurs prévues d'entrée		
7.3	Restreindre le format des entrées autant que possible		
7.4	S'assurer que le code de validation est central		
7.5	Filtrer les caractères spéciaux		
7.6	Masquer les messages d'erreur de langage SQL pour tous les utilisateurs		
7.7	Bloquer les multiples instances de paramètre		
7.8	Valider après l'encodage		
7.9	Sécuriser le téléversement de fichiers		
7.10	Tester la logique métier		
7.11	Effectuer des tests de pénétration		
Configuration			
8.1	Désactiver l'exploration des répertoires		
8.2	Supprimer les fichiers du répertoire Web inutiles		
8.3	Verrouiller les composants de services Web		
8.4	Désactiver la mise en cache des justificatifs d'identité		
8.5	Utiliser des scanneurs de vulnérabilité		
8.6	Automatiser le déploiement		
Opérations			

9.1	Surveiller les activités sur le site Web		
9.2	Élaborer un plan d'intervention en cas d'incident		
9.3	Établir une stratégie d'application de correctifs		
9.4	Promouvoir la sensibilisation à la sécurité		

3 ARCHITECTURE SÉCURITAIRE

Vos services Web reçoivent et répondent à des demandes provenant d'utilisateurs sur Internet. Pour garantir la sécurité permanente de vos services Web et des données sensibles que vous recueillez, traitez et stockez, vous devez vous assurer de concevoir une architecture sécuritaire. Dans le cadre du présent document, le terme *architecture* fait référence au degré de sécurité et d'efficacité avec lequel vos services Web et leurs composants sous-jacents sont agencés pour fournir un service. Une architecture sécuritaire comporte des principes comme la séparation et la redondance pour assurer la protection de vos composants de services Web contre des compromissions.

3.1 SÉPARATION DES COMPOSANTS DES SERVICES WEB

Lorsque vous concevez vos services Web, vous devez distinguer ou séparer vos composants de services Web les uns des autres. Ce faisant, vous êtes assuré que si un composant est compromis, les autres composants seront protégés. Par exemple, si un auteur de menace réussit à accéder à une partie de votre infrastructure, il ne pourra pas avoir accès aux autres composants.

Un service Web simple comporte les composants suivants : un serveur Web, une application et une base de données. Le serveur Web accepte et formate les demandes du navigateur client sur Internet, puis il achemine ces demandes vers l'application pour qu'elles soient traitées. Le serveur Web reçoit également des réponses de l'application et il les présente au navigateur client (p. ex. la couche présentation). L'application effectue le traitement principal dans le service Web, appliquant ainsi la logique métier aux demandes (p. ex. la couche application). La base de données est votre référentiel de données persistant qui héberge vos données sensibles (p. ex. la couche persistante).

Une conception simple peut permettre que tous les composants soient réunis sur un seul serveur Web. Ce concept fonctionne d'un point de vue fonctionnalité, mais il laisse vos données vulnérables face à des menaces. Si le serveur Web est compromis, et que tout est stocké sur un serveur Web, vos données sont alors également compromises. Afin de sécuriser votre architecture, utilisez des techniques comme la séparation physique (lorsque la situation le permet), des coupe-feux, des politiques en matière de sécurité et des contrôles d'accès. Si vous faites affaire avec un fournisseur de services, c'est à vous à déterminer comment ce fournisseur doit séparer les composants de services Web.

3.1.1 ISOLER LE SERVEUR WEB DU RÉSEAU

Le serveur Web est la partie frontale de votre application Web. Le serveur Web est considéré comme le *Web frontal*, ce qui signifie qu'il est visible et qu'il donne aux utilisateurs un accès par Internet. Votre serveur Web communique avec le navigateur de l'utilisateur; il recueille les entrées utilisateur et envoie le résultat à l'utilisateur. Le serveur Web traite les données minimales. Les données de l'utilisateur sont plutôt dirigées vers l'application pour être traitées.

Les auteurs de menace ciblent les composants du Web frontal, comme votre serveur Web, parce qu'ils représentent des points d'entrée à votre réseau et à vos données.

Vous auriez avantage à séparer physiquement votre serveur Web des autres composants de service Web, notamment vos bases de données et votre serveur d'applications. Placez votre serveur Web sur un différent serveur physique et appliquez des contrôles de sécurité entre celui-ci et le reste du réseau. Vous devriez également mettre en place des coupe-feux, des systèmes de détection d'intrusion (SDI) ou des systèmes de prévention d'intrusion (IPS pour *intrusion prevention systems*),

des politiques en matière de sécurité et des autorisations pour protéger votre réseau. S'il n'est pas envisageable de séparer physiquement votre serveur Web, vous devriez mettre en œuvre des contrôles de sécurité compensatoires afin d'atteindre un niveau de risque acceptable pour votre organisation.

Si vous faites appel à un FSG ou à un FSI pour héberger vos services Web, la séparation physique peut se révéler difficile si le fournisseur de services est doté d'une infrastructure multilocation. Au sein d'une infrastructure multilocation, des clients multiples (p. ex. vous et d'autres partenaires payants) partagent la même infrastructure physique. Dans ce cas, les comptes sont séparés par des contrôles logiques, comme c'est le cas avec les autorisations.

Dans un scénario classique, les pratiques exemplaires exigent une séparation physique de tous les composants des services principaux en les plaçant sur différents serveurs physiques et en mettant en œuvre des contrôles physiques, comme un coupe-feu, entre chacun.

Dans une infrastructure multilocation partagée, l'argument commercial en faveur de la séparation physique peut être plus difficile à justifier en raison des coûts plus élevés. Toutefois, la pratique exemplaire est de séparer physiquement votre serveur Web des autres composants du réseau, comme votre serveur d'applications et votre base de données.

3.1.2 ISOLER LE SERVEUR D'APPLICATIONS ET LA BASE DE DONNÉES

Le serveur d'applications est l'arrière-plan de l'application Web. Les données sont acheminées du serveur Web au serveur d'applications, où elles sont traitées en tant que logique métier d'application (c.-à-d. les règles qui déterminent comment les données sont créées, stockées ou modifiées). Les données sont stockées dans une base de données ou sont réacheminées vers le serveur Web. La base de données est le référentiel de données pour l'application Web.

Vous devez séparer votre serveur d'applications de votre serveur de base de données. Si un auteur de menace arrive à s'introduire dans votre réseau et est en mesure d'attaquer les composants de celui-ci, vous devez faire en sorte que l'accès aux données sensibles soit le plus difficile possible pour cet auteur. Comme il est recommandé, vous devriez également mettre en œuvre des contrôles propres au système, comme l'application d'autorisations d'accès au serveur d'application et à la base de données, et l'installation de coupe-feux entre les composants.

Examinez les relations entre des composants d'arrière-plan : une base de données contient des données non structurées, un système de gestion du contenu structure les données, et un moteur de recherche cherche des index pour organiser les données. En isolant ces composants et en appliquant les contrôles de sécurité appropriés, il vous est possible d'empêcher un auteur de menace d'avoir accès aux données si un de ces composants est compromis. Par exemple, si un auteur de menace accède au serveur du système de gestion de contenu, il peut voir des modèles, mais pas des données. Si cet auteur accède à la base de données, il ne pourra que voir des données non structurées qui ne sont pas faciles à lire. Si l'auteur de menace accède aux composants du moteur de recherche, mais pas aux données formatées, il pourrait voir une partie de la fonction d'indexation, mais pas les données qu'il recherche.

Vous devez vous concentrer sur la séparation de l'application Web, car il s'agit d'une cible de grande valeur étant donné qu'elle réunit beaucoup des composants d'arrière-plan. Toutefois, s'il n'est pas possible de séparer les données, vous devriez mettre en œuvre des protections supplémentaires. En voici quelques exemples : exiger une authentification pour chacun des systèmes qui fournit des données à l'application, mettre en œuvre une *intégrité par deux personnes* (p. ex. deux personnes doivent se connecter pour exécuter des actions), permettre une authentification multifacteur et surveiller l'accès au système.

3.2 PROTÉGER LES SERVICES WEB CONNECTÉS À INTERNET

Les auteurs de menace ciblent les composants du Web frontal, car ils servent de points d'entrée à votre réseau. Lorsque la situation le permet, vous devez cerner et arrêter les demandes malveillantes avant qu'elles n'atteignent votre réseau.

Il vous est possible de protéger vos services Web connectés à Internet en utilisant les techniques suivantes :

- Insérez un serveur mandataire à l'avant de vos services Web;
- Utilisez un coupe-feu d'applications Web (WAF pour *Web Application Firewall*), qui permet de filtrer, de surveiller et de bloquer le trafic du protocole HTTP;
- Mettez en œuvre des contrôles d'accès et des méthodes d'authentification (voir les sections 4 et 5);
- Surveillez et utilisez un SDI ou un IPS (voir la section 9).

3.2.1 INSTALLER UN SERVEUR MANDATAIRE

Un serveur mandataire fournit une couche de sécurité en agissant comme intermédiaire entre vos services et les demandes des clients. Il peut agir comme un coupe-feu et un filtre pour les demandes des clients. Vous devriez configurer un serveur mandataire devant vos services Web puisqu'il peut fournir les fonctions suivantes :

- **Authentifier les utilisateurs :** Un mandataire agit comme contrôle de sécurité lors de l'authentification. Il peut authentifier les utilisateurs avant qu'ils n'accèdent au serveur Web de votre réseau.
- **Augmenter la performance** : Un mandataire met en cache le contenu pour plusieurs clients pour que l'accès aux données couramment demandées se fasse rapidement.
- **Appliquer la traduction NAT :** Un mandataire dissimule votre réseau de l'Internet, ce qui limite la surface d'attaque.
- **Surveiller :** Un mandataire peut faciliter un chemin d'accès connu pour vos clients. C'est sur chemin que doivent se concentrer vos activités de surveillance et de protection d'intrusion.
- **S'occuper du déchargement du protocole SSL** : Certains dispositifs mandataires peuvent aussi s'occuper du déchargement du protocole SSL pour atténuer le fardeau du chiffrement et du déchiffrement depuis les serveurs Web. En utilisant des accélérateurs matériels intégrés, les mandataires Web peuvent effectuer plus efficacement la terminaison et l'accélération SSL, ce qui améliore la performance et permet aux serveurs Web d'axer les cycles de l'unité centrale de traitement (UCT) sur le traitement des demandes Web.

Précisez parmi les services mandataires mentionnés ci-dessus ceux dont vous avez besoin. Si vous avez besoin de services d'authentification, vous devriez envisager d'adopter des exigences en matière de gestion de certificat et de cycle de vie de certificat.

Vous pourriez aussi faire appel à un FSG pour les services mandataires. Dans ce cas, le service mandataire repose dans l'infrastructure du fournisseur. Avant de choisir un FSG, passez en revue les services mandataires offerts.

3.2.2 UTILISER UN COUPE-FEU WAF

Vous devriez installer un coupe-feu WAF entre vos services Web et Internet. Un WAF est un coupe-feu au niveau de l'application qui filtre, surveille et bloque le trafic HTTP. Il permet d'inspecter les paquets de protocoles TCP/IP

(*Transmission Control Protocol/Internet Protocol*) sur la connexion avant qu'ils n'atteignent le serveur Web. Voici quelques exemples des fonctions qu'offre le coupe-feu WAF :

- **Suivre les sessions** : Un coupe-feu WAF surveille l'information relative au jeton de session pour détecter toute trace de trafiquage.
- **Ralentir les attaques par déni de service (DoS) :** Un WAF peut identifier des signes d'une attaque DoS possible, ce qui permet au réseau de réagir et de minimiser les dommages.
- **Appliquer des correctifs :** Un coupe-feu WAF peut offrir une solution temporaire à court terme pour corriger des applications Web. Lorsqu'une vulnérabilité est trouvée dans une application Web, il faudra peut-être un certain temps avant de pouvoir mettre en œuvre un correctif en raison de vos processus de développement et d'assurance de la qualité. Après avoir identifié un problème, vous pouvez établir les règles relatives au coupe-feu WAF de façon à bloquer une attaque jusqu'à ce que vous puissiez mettre en œuvre le correctif sur l'application Web.
- **Remédier aux anomalies de connexion :** Un coupe-feu WAF peut régler des problèmes d'anomalies de connexion dans l'application Web. Il peut de plus inspecter les paquets avant que les messages ne parviennent à l'application Web. Cette fonctionnalité permet également de décharger certaines tâches de connexion du serveur d'applications.

Précisez parmi les services WAF mentionnés ci-dessus ceux dont vous avez besoin. Il est à noter que la configuration d'un coupe-feu WAF peut s'avérer complexe. Vous devez définir les règles WAF de façon aussi précise que possible. Toutefois, si vous établissez des règles trop strictes, vous risquez de bloquer du trafic légitime ou de générer beaucoup de faux positifs, ce qui risque d'encombrer vos connexions rendant ainsi difficile la détection de problèmes réels.

3.3 METTRE EN ŒUVRE DES COMPOSANTS REDONDANTS

Vous devriez intégrer la redondance à votre concept afin d'éviter un point unique de défaillance pour vous aider à reprendre efficacement les activités et à minimiser les répercussions lorsque survient un problème. Tous les composants matériels peuvent, à un certain moment, s'exposer à un risque de défaillance. La redondance (c.-à-d. la reproduction) est un principe architectural qui fait en sorte que des ressources multiples remplissent la même fonction. En intégrant des redondances aux composants de vos services Web, vous pouvez assurer la disponibilité des systèmes et des services, et réduire le risque de perdre des données. De plus, la redondance permet de réduire l'incidence d'attaques DoS, qui peuvent cibler des composants du Web frontal dans le but de perturber intentionnellement des services. Il vous faudrait privilégier la création de redondances pour les composants du Web frontal.

Votre plan de redondance doit d'abord être axé sur les composants du Web frontal, comme les serveurs mandataires, mais il doit aussi tenir compte des composants internes, comme les serveurs Web, les équilibreurs de charge, les bases de données et les serveurs d'applications. Si un équilibrage de charge est nécessaire, vous devez déterminer les mécanismes de distribution d'équilibrage de charge mis à votre disposition (p. ex. source IP, port source, quintuple) et vous assurer que ces mécanismes sont compatibles avec vos serveurs Web et applications.

Veillez à sauvegarder régulièrement vos données. Il ne suffit pas de créer des composants redondants et des sauvegardes; vous devez les tester rigoureusement pour vous assurer de leur bon fonctionnement.

3.4 SÉPARER LES INTERFACES ADMINISTRATIVES

Vous devez séparer l'interface administrative de votre site Web. Les utilisateurs administratifs ou privilégiés ont un niveau d'accès élevé à votre réseau. Lors de l'attribution de privilèges à des utilisateurs, vous devez toujours tenir compte de l'impact que pourrait avoir la compromission de leurs comptes. Vous devez également vous prémunir d'interfaces de connexion distinctes pour séparer les utilisateurs généraux des utilisateurs administratifs ou privilégiés. Les interfaces de connexion pour les applications Web sont connectées à Internet, ce qui en fait des cibles pour les attaques. Des problèmes liés à l'interface de connexion utilisateur peuvent rapidement dégénérés si un administrateur utilise cette même interface. S'il survient un problème avec votre site Web, c'est à votre administrateur que revient la tâche de le régler.

Vous devez séparer l'interface administrateur de votre site Web. Dans le cas des environnements de travail à distance, les utilisateurs administratifs ou privilégiés doivent utiliser un réseau privé virtuel (RPV) pour se connecter et avoir accès à votre réseau.

De plus, vous devriez stocker des justificatifs utilisateur administratifs distincts des justificatifs d'utilisateur général; par exemple, dans une instance de base de données différente.

4 CONTRÔLE D'ACCÈS

Le contrôle d'accès fait référence aux règles et aux techniques utilisées pour s'assurer que les entités (p. ex. les systèmes, les processus et les gens) n'ont accès qu'aux systèmes et aux données dont ils ont besoin pour effectuer des fonctions autorisées. Les contrôles d'accès définissent qui peut accéder à quelles ressources sur votre site Web et ils limitent l'information que ces personnes peuvent voir et utiliser. Le contrôle d'accès est un élément central de la confidentialité, de l'intégrité et de la disponibilité des données. Que vous hébergiez vous-même votre site Web, ou qu'il soit hébergé par un fournisseur de services, votre organisation est responsable de sa politique en matière de contrôle d'accès.

4.1 PLANIFIER ET TESTER LES CONTRÔLES D'ACCÈS

L'accès à une application ou à un réseau nécessite une planification du contrôle d'accès. Lorsque vous définissez les contrôles d'accès, songez à appliquer le principe du droit d'accès minimal et le principe du refus par défaut. Le principe du droit d'accès minimal s'assure que les entités ont l'accès minimal dont elles ont besoin pour effectuer les tâches qui leur sont autorisées. Dans le principe du refus par défaut, les demandes sont refusées à moins que la source d'une demande ait reçu l'autorisation de votre organisation conformément à la liste autorisée.

Le contrôle d'accès est compliqué dans le cadre d'une application Web. Il y a en jeu de nombreuses couches de mécanismes lorsqu'un accès est donné aux ressources ou aux données. Les couches de contrôle d'accès de votre application Web doivent comprendre ce qui suit :

- **Contrôle d'accès par URL sur le serveur Web ou la plateforme Web :** Refuser certains éléments URL. Si vous désirez que l'accès à votre site Web se fasse uniquement de certains emplacements géographiques, cela peut comprendre des protections de géolocalisation.
- **Système de fichier et autorisation de serveur :** Pour traverser un serveur Web, une autorisation de système de fichier et une autorisation de serveur Web doivent être examinées.
- **Contrôle d'accès (logique métier) à l'application :** Contrôle d'accès pour définir ce que l'utilisateur peut et ne peut pas faire dans les fonctions d'application.
- **Contrôle d'accès à la couche de données :** Complète le contrôle d'accès à l'application en contrôlant l'accès aux éléments de la base de données.
- **Contrôle d'accès à l'application (couche de présentation) :** Contrôle les limites de l'application qu'un utilisateur peut afficher.

Lorsque vous concevez des contrôles d'accès pour chacune des couches, vous devez toujours tenir compte de la défense en profondeur et de l'identité ainsi que du rôle de l'utilisateur. Définissez les rôles et les fonctions que chaque rôle peut effectuer. Pour chaque fonction, vous devez être aussi précis que possible en définissant ce que les entités peuvent faire et voir (p. ex. lecture, mise à jour ou suppression de données), et ce à quoi les entités peuvent accéder dans votre base de données. Vous devez créer une matrice de contrôle d'accès qui utilise des règles ou des rôles pour lier des ressources à des objets. Par exemple, créez un tableau qui permet de distinguer les rôles administratifs, comme les administrateurs de système et les administrateurs d'application. Un tableau supplémentaire devrait être préparé pour décrire les données qui peuvent être vues. Il est essentiel de vous assurer que vos contrôles d'accès sont correctement mis en œuvre.

Assurez-vous de définir les exigences permettant de passer en revue et de mettre à jour les privilèges régulièrement (p. ex. lorsqu'un utilisateur change de rôle ou quitte l'organisation).

5 MÉCANISMES D'AUTHENTIFICATION

L'authentification fait référence aux mécanismes qui assurent la vérification de l'identité d'un utilisateur. L'authentification fonctionne de concert avec une autorisation. Lorsqu'un utilisateur prouve son identité, il obtient uniquement l'accès aux ressources dont il a besoin.

Avec des demandes d'authentification qui proviennent de l'Internet non fiable, vous devez vous assurer de vérifier avec certitude les identités afin de protéger votre site Web et vos renseignements sensibles. Votre approche à l'égard de l'authentification d'une application Web doit comprendre les éléments suivants :

- une politique en matière de mots de passe robustes;
- une récupération de compte sécuritaire;
- une authentification de certificat;
- des mécanismes de protection de mot de passe intégrés;
- autorisation de verrouillage de compte, de délais de connexion et de tests CAPTCHA.

5.1 ÉTABLIR UNE POLITIQUE EN MATIÈRE DE MOTS DE PASSE ROBUSTES

Établissez une politique en matière de mots de passe robustes qui comporte un nombre minimum de caractères et des dates d'expiration pour les mots de passe. Vous ne devriez pas autoriser d'indices de mot de passe, car ils permettent plus facilement aux auteurs de menace de deviner les mots de passe. Nous croyons que les normes suivantes produisent le meilleur effet dissuasif :

- les phrases passe doivent comporter au moins 4 mots et 15 caractères;
- les mots de passe doivent comporter au moins 12 caractères;
- les codes secrets et les numéros d'identification personnelle (NIP) ne doivent être utilisés que si les phrases passe et les mots de passe ne peuvent pas être utilisés.
    - Dans la mesure du possible, utilisez des NIP générés de façon aléatoire.

Assurez-vous de bien tester la mise en œuvre de vos mots de passe. Pour obtenir de plus amples renseignements sur les pratiques exemplaires en matière de mots de passe, consultez le document *Pratiques Exemplaires de Création de Phrases de Passe et de Mots de Passe (ITSAP.30.032)* [5].

Les attaques de bourrage de justificatifs deviennent plus fréquentes. Lors d'une attaque de bourrage de justificatifs, des auteurs de menace se sont servis de justificatifs volés pour tenter de se connecter à d'autres sites Web et systèmes jusqu'à ce que des correspondances soient trouvées. Les attaques de bourrage de justificatifs fonctionnent parce que les gens ont tendance à réutiliser leurs mots de passe pour plusieurs sites Web.

Dans le but de protéger vos clients contre des attaques de bourrage de justificatifs, encouragez-les à ne pas réutiliser leurs mots de passe et empêchez-les d'utiliser des mots de passe qui ont déjà fait l'objet d'une fuite. Ne permettez pas aux utilisateurs d'entrer leurs adresses de courriel comme noms d'utilisateur. Lorsque vous permettez que des adresses de courriel soient utilisées comme noms d'utilisateur, vous donnez aux auteurs de menace une combinaison plausible de nom d'utilisateur et de mot de passe.

Il existe des ressources en ligne qui permettent d'identifier des mots de passe compromis, dont le site *Have I Been Pwned* et son service *Pwned Passwords*. Par l'entremise de ce service, les utilisateurs peuvent voir si certaines de leurs informations ont fait l'objet d'une atteinte à la protection des données. Ce service a été créé après que le National Institute for Standards and Technology (NIST) ait formulé des directives recommandant spécifiquement que les mots de passe fournis par l'utilisateur soient vérifiés par rapport aux atteintes à la protection des données existantes.

En raison des exigences relatives aux mots de passe qui sont de plus en plus complexes, les utilisateurs peuvent avoir de la difficulté à se rappeler de mots de passe complexes sans les écrire, ce qui pose un risque à la sécurité des mots de passe. Les sites Web utilisent de plus en plus de services de gestion des mots de passe qui proviennent de fournisseurs qui peuvent non seulement stocker et gérer les mots de passe, mais aussi générer pour les utilisateurs des mots de passe complexes. Pour avoir des conseils de sécurité sur les gestionnaires de mots de passe, consultez le document *Conseils de sécurité sur les gestionnaires de mots de passe (ITSAP.30.025)* [6].

5.2 PERMETTRE UNE AUTHENTIFICATION MULTIFACTEUR

Nous recommandons fortement de permettre l'authentification multifacteur (MFA pour *multi-factor authentication*) pour tous les types de comptes (p. ex. administrateurs, utilisateurs). La MFA oblige les utilisateurs à avoir au moins deux facteurs d'authentification différents pour se connecter à leurs comptes. Comme l'indique notre publication, *Guide sur l'authentification des utilisateurs dans les systèmes de technologie de l'information (ITSP.30.031 v3)* [7], il existe trois principaux types de facteurs d'authentification :

1. Quelque chose que vous savez (p. ex. un NIP, un mot de passe ou une phrase passe);
2. Quelque chose que vous avez (p. ex. un jeton d'authentification matériel, une carte intelligente);
3. Quelque chose que vous êtes (p. ex. une biométrie telle une empreinte digitale).

Remarque : On reconnaît de plus en plus le besoin de facteurs d'authentification supplémentaires pour renforcer davantage l'authentification, comme pour ce qui touche votre emplacement ou le comportement des utilisateurs (p. ex. ce que vous faites).

Il est plus facile de mettre en œuvre un facteur d'authentification de type 2 (c.-à-d. quelque chose que vous avez), parce que cette mesure n'oblige pas l'utilisateur (ou composant de navigateur) de connaître le facteur ou de s'en souvenir (p. ex. un mot de passe ou un NIP), et vous pouvez intégrer l'entrée à un formulaire HTML. Toutefois, il n'est peut-être pas possible de fournir un jeton d'authentification matériel à tous les clients du site Web. Vous pourriez plutôt émettre des jetons d'authentification à des utilisateurs ou à des administrateurs spécifiques.

Beaucoup de fournisseurs offrent des solutions propriétaires pour les jetons d'authentification multifacteur. Dans la plupart des cas, le fournisseur donne l'interface de programmation d'applications pour différents langages de réalisation de programme que le développeur intègre à l'application d'authentification. Si vous utilisez une de ces solutions, assurez-vous qu'elle emploie au moins deux des types de facteurs d'authentification énumérés ci-dessus. Vous devez également vous assurer que le fournisseur puisse prendre en charge une distribution électronique du second facteur.

L'authentification multifacteur n'est pas une solution parfaite, mais cette mesure fait en sorte que les auteurs de menace doivent travailler plus forts pour arriver à pirater un compte. Il peut arriver que des auteurs de menace s'approprient des sessions utilisateur ou compromettent plus d'un support pour faire obstacle aux mécanismes d'authentification hors bande.

Toutefois, en utilisant des jetons par utilisateur et des jetons sur le point d'expirer, vous serez en mesure de limiter ou d'éliminer le laps de temps au cours duquel un auteur de menace peut se servir des identifiants de connexion ciblés.

En outre, l'authentification multifacteur ne peut empêcher les attaques de l'intercepteur. Un auteur de menace peut également envoyer un message d'hameçonnage comportant un lien vers une page de connexion frauduleuse pour inciter le destinataire à entrer ses justificatifs d'ouverture de session.

5.3 METTRE EN PLACE UN PROCESSUS SÉCURITAIRE DE RÉCUPÉRATION DE COMPTE

Il vous faut définir un processus de récupération de compte sécuritaire. Si un client perd un ou ses deux facteurs d'authentification, ce processus de récupération lui permet de réinitialiser ses mots de passe et de se connecter à ses comptes. Vous devez mettre en œuvre de manière rigoureuse le processus de récupération de compte afin de minimiser les risques que posent les auteurs de menace qui se font passer pour des clients légitimes. Vous pouvez également aviser le client chaque fois qu'une connexion anormale se produit (p. ex. à partir d'un nouveau dispositif ou d'un nouvel emplacement).

Votre processus de reprise de compte doit comprendre au minimum les étapes suivantes :

1. Demandez à l'utilisateur de s'identifier (p. ex. nom d'utilisateur, numéro de client, adresses de courriel);
2. Posez des questions de sécurité privée;
   - Choisissez avec soin ces questions et évitez de poser des questions qui reposent sur des réponses basées sur de l'information accessible au public. Utilisez plutôt des questions ayant trait à des transactions antérieures ou à des détails sur d'autres comptes;
3. Générez un courriel contenant un lien ponctuel périssable;
   - Ce courriel fournit un deuxième facteur d'authentification en communiquant avec l'utilisateur sur un autre support;
4. Permettez à l'utilisateur d'entrer un nouveau mot de passe;
   - L'utilisateur doit respecter les exigences en matière de complexité de mot de passe;
   - L'utilisateur entre le nouveau mot de passe deux fois pour valider le changement;
5. Avisez l'utilisateur que les détails du mot de passe ou du compte ont changé.

5.4 UTILISER DES MESURES POUR PROTÉGER LES MOTS DE PASSE

En plus d'une politique en matière de mots de passe robustes, vous devriez appliquer des mesures supplémentaires pour protéger les mots de passe. Lorsque les utilisateurs se connectent à votre site Web, ne permettez pas l'envoi de mots de passe en texte clair. Exigez plutôt l'utilisation du protocole HTTPS et du hachage. Vous devez revoir vos pratiques de codage pour vous assurer que les justificatifs et les clés d'interface de programmation d'applications ne puissent jamais être codés en dur dans vos services Web.

5.4.1 UTILISER LE PROTOCOLE HTTPS

Vous devriez obliger l'utilisation du protocole HTTPS pour les données en transit. Bien que l'authentification de base HTTP soit un mode d'authentification largement utilisé et qu'il soit pris en charge par de nombreux navigateurs et serveurs, elle

comporte une faille critique. Le protocole HTTP envoie des données, dont des justificatifs d'identité, qui permettent à n'importe qui de lire ces données. Ce texte clair peut être affiché dans la ligne de l'URL. Le protocole HTTP est vulnérable aux attaques par réinsertion et à l'écoute clandestine. Le protocole HTTPS n'est pas différent du protocole HTTP. Le protocole HTTPS nécessite un accès tunnellisé chiffré TLS/SSL à travers le protocole HTTP. Le protocole TLS/SSL chiffre les données d'authentification pour empêcher les attaques par réinsertion et par écoute clandestine. Nous recommandons l'utilisation du protocole TLS 1.3 ou une version supérieure. Pour obtenir des recommandations supplémentaires sur le chiffrement de données en transit, consultez *Algorithmes cryptographiques pour l'information NON CLASSIFIÉ, PROTÉGÉ A et PROTÉGÉ B (ITSP.40.111)* [8].

5.4.2 UTILISER LA FONCTION DE HACHAGE ET DE SALAGE DE MOTS DE PASSE

Conformément à la recommandation qui se trouve dans le document *Special Publication 800-63-3 Digital Identity Guidelines* [9] du NIST, vous devez générer le hachage et le salage de mots de passe à l'aide d'une fonction adéquate de hachage à sens unique.

Une fonction de hachage entre le mot de passe, une valeur de salage et une valeur de coût pour générer un hachage de mot de passe. Une valeur de salage fait référence à une chaîne aléatoire de caractères qui sont ajoutés au mot de passe et que seul le serveur connaît. Les valeurs de salage doivent avoir une longueur d'au moins 32 bits. Le salage des mots de passe se fait avant que ne soit effectuée la fonction de hachage pour rendre le mot de passe plus long et ainsi plus difficile à deviner si un auteur de menace venait à intercepter le fichier de hachage de mot de passe sur le serveur. L'objectif de la valeur de coût est de rendre les attaques par supposition de mot de passe onéreuses, voire prohibitives pour les auteurs de menace.

N'utilisez que des algorithmes de hachage approuvés conformément à l'ITSP.40.111 [8] et un générateur de bits aléatoires sécurisé par chiffrement pour générer des valeurs de salage. D'autres recommandations sont données dans la publication NIST SP 800-63-3 [9]. Ne stockez pas les valeurs de hachage et les valeurs de salage au même endroit dans le serveur. Lorsqu'elles sont séparées dans des endroits distincts, il devient plus difficile pour un auteur de menace d'intercepteur les deux.

Le hachage à lui seul ne garantit pas une protection suffisante. Vous pourriez également envisager de faire ce qui suit :

- Chiffrer les hachages stockés et utiliser le protocole HTTPS pour chiffrer les mots de passe en transit vers le serveur;
- Isoler les valeurs de hachage, les valeurs de salage et les clés de chiffrement les unes des autres sur le serveur;
- Stocker les mots de passe côté client dans un format qui résiste aux attaques hors ligne.

5.4.3 ENVISAGER DES OPTIONS D'ENTRÉE DE MOTS DE PASSE

Afin de soutenir l'utilisation de gestionnaires de mots de passe, permettez la fonctionnalité couper-coller dans les champs de mot de passe. Vous pourriez aussi permettre aux utilisateurs de voir les mots de passe pour qu'ils puissent vérifier ce qu'ils tapent.

5.4.4 N'EFFECTUER AUCUN CODAGE EN DUR DE JUSTIFICATIFS DE BASE DE DONNÉES ET DE CLÉS D'INTERFACE DE PROGRAMMATION D'APPLICATIONS

Dans le cadre de vos pratiques exemplaires en matière de codage, avant d'entamer le développement, vous devriez préciser que les justificatifs de base de données et les clés d'interface de programmation d'applications ne doivent pas être faire l'objet d'un codage en dur. Étant donné que les justificatifs sont requis tout au long du code, la gestion des justificatifs et leurs mises à jour sont complexes et donnent souvent lieu à des erreurs lorsqu'il est question de codage en dur. De plus, les applications Web accèdent à la base de données d'arrière-plan avec la base de données et les justificatifs d'interface de programmation d'applications. Si une application Web est compromise et que son code est exposé avec des justificatifs intégrés, les auteurs de menace peuvent alors avoir accès à votre base de données.

Vous devriez plutôt placer les justificatifs dans un environnement distinct et faire une référence à l'exécution à partir de l'application. Vous devriez également exiger que toutes les pages de connexion utilisent une interface de programmation d'applications standard.

Des solutions sous forme de logiciel de gestion des justificatifs sont offertes pour permettre l'accès aux justificatifs uniquement en cas de besoin. Attendez-vous à ce que les clients utilisent ces solutions pour gérer leurs justificatifs. Ne bloquez pas l'option couper-coller à la connexion.

5.5 UTILISER L'AUTORISATION DE VERROUILLAGE DE COMPTE, DE DÉLAIS DE CONNEXION ET DE TESTS CAPTCHA

Lors d'une attaque par force brute, un auteur de menace essaie à plusieurs reprises, par tâtonnement, de se connecter jusqu'à ce qu'il trouve la bonne combinaison de justificatifs. Pour empêcher les attaques par force brute, vous ne devriez pas permettre aux utilisateurs d'utiliser leur adresse de courriel comme nom d'utilisateur (voir la sous-section 5.3). Vous devez également mettre en œuvre des contrôles de sécurité complémentaires, notamment les verrouillages de compte, les délais de connexion et les tests CAPTCHA.

Vous devez surveiller les tentatives de connexion manquées pour identifier les attaques par force brute potentielles et aviser votre administrateur de la situation le plus rapidement possible. Le recours à un coupe-feu WAF peut vous aider dans le cadre de vos activités de détection et d'intervention.

Avec la fonction de verrouillage de compte, les utilisateurs se voient fixer un nombre maximal de tentatives de connexion avant que leurs comptes soient verrouillés. Ce seuil donne aux utilisateurs ou aux auteurs de menace un nombre limité d'occasions de se connecter avant que le compte soit gelé, les forçant automatiquement à suivre la procédure de récupération de compte. Vos développeurs doivent passer en revue les mécanismes d'authentification en place. Si vous utilisez un mécanisme d'authentification de base ou condensé, vous devez le personnaliser de façon à prendre en charge le verrouillage de compte. Quelques fournisseurs offrent des serveurs d'authentification d'arrière-plan dotés d'une fonctionnalité de verrouillage de compte intégrée, dont le protocole Lightweight Directory Access Protocol (LDAP), le système d'exploitation Active Directory (AD) et le serveur SQL.

Ajoutez un délai à chaque tentative de connexion. Lors d'attaques par force brute, beaucoup d'auteurs de menace utilisent des outils automatisés. Un délai de 0,5 seconde suffit habituellement pour interrompre ses outils.

Vous devriez également ajouter un test CAPTCHA ou un mécanisme similaire pour confirmer qu'un utilisateur n'est pas un robot après un ou deux échecs de tentatives de connexion. Par exemple, un mécanisme CAPTCHA peut demander à l'utilisateur d'observer une image et d'entrer les caractères qu'il voit. Vous n'avez pas à développer ce mécanisme. Il est possible de l'acheter sur le marché et de l'intégrer à votre interface. Vous pouvez vous procurer la famille de tests CAPTCHA et contacter des fournisseurs de solutions de rechange pour discuter de la possibilité d'utiliser les mécanismes CAPTCHA.

6 SÉCURISER LES SESSIONS

Une session est basée sur un échange d'information entre au moins deux entités, comme deux dispositifs ou un utilisateur et un serveur Web. Si les sessions ne sont pas traitées de façon sécuritaire, les auteurs de menace peuvent interrompre ou s'approprier les sessions pour intercepter les données ou se faire passer pour des utilisateurs authentifiés.

La gestion de session est le processus d'amorce, de contrôle, de maintien et d'arrêt des échanges entre au moins deux entités. Lors d'une session, un identifiant unique est marqué pour toutes les demandes HTTP faites par un utilisateur à partir du moment où il se connecte jusqu'à ce qu'il mette fin à la session. La gestion de session autorise au serveur à reconnaître l'utilisateur entre les demandes. Lorsqu'un utilisateur est authentifié, il reçoit un identifiant de session pour qu'il n'ait pas à s'authentifier à chaque demande.

Le suivi de session permet de conserver certaines données entre les sessions. Le serveur peut faire le suivi des données d'un utilisateur, et ce, d'une session à l'autre (p. ex. montrer l'historique d'achats de l'utilisateur).

Un témoin est un paquet de données qui contient l'information sur la session de l'utilisateur. Lorsqu'un utilisateur visite un site Web, le serveur renvoie une source d'information au navigateur de l'utilisateur. Le navigateur stocke cette information et la retourne au serveur dans chaque communication subséquente à ce site.

Ces concepts contribuent grandement à la fonctionnalité des sites Web, mais ils augmentent également les risques. Les auteurs de menace peuvent voler ou saboter les sessions d'utilisateur au moyen d'attaques comme celles concernant le script intersites (XSS) ou le fichier CSRF. Lors d'une attaque XSS, un auteur de menace compromet un serveur Web et injecte un code malveillant dans un site Web. Lorsque les utilisateurs visitent le site Web en question, leurs navigateurs exécutent ce script malveillant, mettant à risque les témoins, les jetons de session ou l'information sensible. Lors d'une attaque de type CSRF, les utilisateurs sont amenés par la ruse à exécuter des interventions indésirables dans leurs navigateurs comme se déconnecter, télécharger de l'information sur le compte ou téléverser des témoins.

6.1 UTILISER DES BOÎTES À OUTILS DE GESTION DE SESSION

De langages de programmation comme Hypertext Preprocessor (PHP), .NET et Active Server Pages (ASP) sont dotés de fonctions de gestion de session intégrées. Il n'est donc pas nécessaire de réinventer des mécanismes de gestion de session à moins de disposer d'un spécialiste interne qui se charge de cette tâche. L'utilisation de produits déjà établis vous permet de tester et de mettre à jour plus facilement ces mécanismes. Il est recommandé de passer en revue ces boîtes à outils pour vous assurer qu'elles répondent aux objectifs de votre application et qu'elles sont en mesure de prendre en charge d'éventuelles améliorations de service. Si vous optez pour une boîte à outils, prenez sois de consigner les configurations de sécurité pour la répétabilité.

Remarque : Étant donné que les boîtes à outils ne sont pas par nature sécurisées, vous devez suivre les directives données dans les sous-sections 6.2 à 6.6 pour sécuriser les sessions.

6.2 UTILISER DES IDENTIFIANTS DE SESSION ALÉATOIRES QUI RESPECTENT UNE LONGUEUR MINIMALE

Vous devriez utiliser des identifiants de session aléatoires pour empêcher que les auteurs de menace extrapolent le prochain jeton de session. Si les jetons de session sont basés sur des valeurs séquentielles ou utilisent des données comme des horodateurs lors de leur création, ils peuvent alors être extrapolés.

Vous devriez également mettre en place une longueur minimale pour les identifiants de session afin d'empêcher les attaques par force brute. L'OWASP recommande une longueur minimale de 128 bits.

6.3 UTILISER DES INDICATEURS DE SÉCURITÉ SUR LES TÉMOINS

Pour assurer la sécurité des témoins, utilisez les indicateurs de témoin suivants :

- **Secure** : L'indicateur *Secure* fait savoir au navigateur que ce témoin doit être envoyé uniquement dans des connexions sécurisées, comme une connexion TLS. L'utilisation du protocole TLS permet de chiffrer l'en-tête du témoin pour qu'il ne puisse pas être lu.
- **HTTPOnly** : L'indicateur *HTTPOnly* demande au navigateur de refuser d'accorder un accès au témoin à partir d'une composante JavaScript. Certaines attaques, comme le script intersites, peuvent arriver à contrôler le langage JavaScript du navigateur de la victime et à envoyer le témoin à des sites malveillants.
- **SAME-SITE**. L'indicateur *SAME-SITE* permet aux témoins d'être envoyés uniquement au même domaine d'où provenaient les témoins.

6.4 STOCKER LES DONNÉES DE TÉMOIN SUR UN SERVEUR

Lors de l'authentification de l'utilisateur, votre serveur Web génère un identifiant de session qu'il envoie au navigateur de l'utilisateur. Toutes les demandes subséquentes au serveur Web seront comprises dans cette valeur, ce qui permet au serveur de reconnaître l'utilisateur sans avoir à l'authentifier à nouveau. Toutefois, beaucoup de services Web font également le suivi des sessions, y compris des préférences d'un utilisateur. Vous devrez stocker les données de session sensibles sur vos serveurs de services Web et ne stocker qu'une quantité minimale des données stockées dans le navigateur de l'utilisateur. Le navigateur d'un utilisateur est moins sécurisé. Cela est souvent dû aux vulnérabilités (p. ex. un logiciel périmé), que les auteurs de menace peuvent exploiter par des attaques XSS et CSRF. De plus, il est possible que des ordinateurs portables et des dispositifs mobiles soient volés ou perdus.

6.5 METTRE UN TERME AUX DONNÉES DE SESSION

Les auteurs de menace peuvent utiliser des données de jetons supprimés pour se faire passer pour des utilisateurs par l'entremise d'une attaque CSRF. Il est recommandé d'appliquer des mécanismes de temporisation pour limiter le laps de temps au cours duquel les auteurs de menace peuvent utiliser les justificatifs de session. De plus, il arrive que les utilisateurs ne se déconnectent pas après leurs sessions. Vous devriez mettre en œuvre un mécanisme d'expiration de session qui se produit après une période définie (p. ex. après 15 minutes) d'inactivité de la part de l'utilisateur.

6.6 AJOUTER DES COUCHES D'AUTHENTIFICATION DE SESSION

Vous devriez créer des couches d'authentification de session. L'authentification n'est pas une activité unique dans les applications Web sécuritaires et modernes. Elle devrait être prise en compte au-delà de la page d'accueil et tout au long de la session au cours de laquelle l'utilisateur interagit avec l'application. Une forme d'authentification peut être effectuée à chaque demande en utilisant une combinaison de comportements de l'utilisateur, d'attributs, de minutage des demandes, d'adresses IP et de détails de navigateur pour déterminer la légitimité des demandes. Vous devriez envisager d'ajouter un mécanisme d'authentification complémentaire lorsque des fonctions à risque élevé sont effectuées, ainsi que les principes de l'authentification multifacteur qui ont été présentés à la section 5.2.

Vous devriez mettre en œuvre des jetons anti-CRSF chaque fois qu'une forme d'authentification est soumise. Vous devez également vous assurer que l'application authentifie de nouveau l'utilisateur lorsqu'il effectue des opérations à risque élevé (p. ex. des demandes de réinitialisation d'un mot de passe). Ce type d'activité permet d'atténuer les attaques XSS et CSRF. Pour obtenir de plus amples renseignements sur la validation d'entrée, consultez la section 7 ci-dessous.

7 VALIDATION D'ENTRÉE

La validation d'entrée est le processus qui permet de vérifier que les utilisateurs et les applications n'entrent que des données correctement formées, comme dans les champs, les formulaires ou les demandes d'information. Les applications Web sont complexes en raison du nombre de parties mobiles entre la source de la demande dans le navigateur du client, Internet et la ressource de destination d'arrière-plan de votre infrastructure Web. Toutes les entrées sur votre site Web doivent être considérées comme étant non sécurisées et, dans la mesure du possible, elles doivent être contrôlées. La validation d'entrée est une méthode d'atténuation efficace des risques liés au développement Web.

Vous pouvez valider des entrées dans les secteurs suivants de votre service Web :

- **Navigateur de l'utilisateur :** Lancement de la demande de l'utilisateur;
- **WAF** : Filtrage avant que la demande atteigne votre réseau de service Web;
- **Serveur Web :** Couche de présentation de vos services Web;
- **Application de la logique métier :** Traitement de la demande;
- **Base de données :** Stockage rémanent.

Si vous utilisez un fournisseur de services, assurez-vous de revoir l'application Web et d'effectuer votre propre test de fonctionnalité. La validation d'entrée doit faire partie de votre processus de développement, et le propriétaire de l'application Web est toujours responsable du risque.

7.1 VALIDER TÔT, MAIS COORDONNER

Une pratique exemplaire veut que vous validiez les entrées le plus tôt possible dans le cadre du processus de traitement pour réduire les contraintes que doivent subir vos serveurs. Tenez compte de tous les secteurs où peut se produire une validation. Par exemple, pensez aux numéros de carte de crédit. Si vous acceptez que les numéros de carte de crédit soient une entrée, la validation débute au navigateur de l'utilisateur, au lieu d'attendre l'application de la logique métier (comme une longueur prévue). La validation à partir du navigateur de l'utilisateur évite qu'un bruit inutile passe par votre infrastructure Web. Toutefois, la validation au niveau du navigateur de l'utilisateur est limitée, et elle peut être contournée du côté de l'utilisateur.

Vous devriez compléter la validation du navigateur à l'aide de méthodes de validations plus fortes. Par exemple, le filtrage fournit un complément flexible permettant d'ignorer les demandes non valides. Règle générale, la validation de l'application Web est plus sécuritaire. Vous devriez valider la partie frontale (comme un formulaire) et la partie en arrière-plan (comme les données transmises par un service), et surtout, ne vous fiez pas uniquement à un de ces deux éléments.

7.2 CONFIRMER LES LONGUEURS D'ENTRÉE PRÉVUES

Le dépassement de la mémoire tampon se produit lorsque les entrées pour une fonction dépassent la mémoire tampon allouée pour les données, ce qui affecte de façon imprévue l'application Web. Vos services Web peuvent également s'en ressentir par une exposition potentielle de données sensibles. Consultez le référentiel *National Vulnerability Database* du

NIST pour obtenir de plus amples renseignements dans la section *CVE-2014-0160 Detail* [10], qui décrit l'exposition à la vulnérabilité Heartbleed découverte en 2014.

Vous pouvez minimiser le dépassement de la mémoire tampon en imposant des limites d'entrée conformes aux longueurs d'entrée prévues et à la capacité de la mémoire tampon. Testez la prévention du dépassement de la mémoire tampon à l'exécution ou dans le cadre des révisions de code. Assurez-vous que votre application Web est en mesure de bien prendre en charge le dépassement de la mémoire tampon et qu'aucune modification inattendue ne se produit dans la base de données. Vous pouvez également utiliser les langages Java, Perl et PHP, qui sont dotés de protections intégrées contre le dépassement de la mémoire tampon.

7.3 S'ASSURER QUE LE CODE DE VALIDATION EST CENTRAL

La validation d'entrée est courante et peut se produire presque n'importe où dans votre code. Ainsi, cette validation peut s'avérer passablement difficile à tester efficacement. Centralisez votre code de validation dans les fonctions au lieu de l'éparpiller sur l'ensemble du code. Identifiez clairement dans votre code l'endroit où se produit la validation. Ceci permet de réutiliser la validation d'entrée pour des entrées semblables et de faciliter une analyse efficace de votre code. Cette précision doit se faire à même vos normes de codage, avant que ne débute le développement. Il faut en outre que les révisions de code vérifient l'efficacité de la validation d'entrée.

7.4 RESTREINDRE LE FORMAT DES ENTRÉES AUTANT QUE POSSIBLE

Dans un champ à structure libre, l'utilisateur peut entrer le texte qu'il désire. Ce type d'entrée nécessite la validation d'entrée la plus complexe en raison du risque accru de voir un élément malveillant s'infiltrer. Pour éviter ce risque, vous devriez limiter le plus possible le format et restreindre les entrées client à structure libre lorsque cela est possible. Utilisez plutôt, si vous le pouvez, des champs à options limitées, comme des listes déroulantes, des boutons radio ou des cases à cocher, qui sont plus faciles à tester. Ne demandez pas les mêmes données plus d'une fois. Confirmez les entrées pour vous assurer d'obtenir le type d'entrée prévue.

7.5 FILTRER LES CARACTÈRES SPÉCIAUX

Le filtrage de combinaisons de caractères spéciaux peut permettre de vous protéger contre différents types de cybermenaces et d'attaques, comme l'injection d'entrées malveillantes. Lorsque cela est possible, utilisez les cadres et les bibliothèques qui existent depuis un certain temps et auxquels la communauté a contribué, au lieu de créer des filtres à partir de rien.

L'injection de ligne de commande d'un système d'exploitation se produit lorsque des applications permettent aux utilisateurs d'envoyer des arguments de ligne de commande dans le système d'exploitation avec des privilèges de service. Cette façon de faire est particulièrement risquée étant donné que les auteurs de menace peuvent utiliser une injection de code de système d'exploitation malveillante pour élever les privilèges ou mener des attaques DoS. Afin d'atténuer cette attaque, vous devez valider les contraintes d'entrée en bloquant des caractères spéciaux comme les points-virgules (;), les barres obliques (/), les barres verticales (|) et les mots-clés d'application ou de base de données.

Lors d'attaques par injection d'en-tête HTTP, l'utilisateur de l'application est ciblé contrairement à l'application Web elle-même. Lorsque l'utilisateur se connecte à l'application Web, la réponse HTTP réagit en fonction de qui est l'utilisateur. Par

exemple, lors d'une connexion à un site bancaire, la réponse HTTP accepte les entrées comme l'ID utilisateur, et redirige celui-ci vers une page déterminée qui contient ses renseignements bancaires. Toutefois, un auteur de menace peut se servir de caractères spéciaux pour créer de nouveaux éléments en-têtes, qui peuvent être utilisés pour usurper les témoins d'un utilisateur ou rediriger cet utilisateur vers un site non valide. Ce moyen est souvent utilisé pour faciliter les attaques XSS et CSRF. Afin d'atténuer ce type d'attaque, vous devriez valider les contraintes d'entrée en bloquant les caractères spéciaux, comme « \r » « \n » et d'autres commandes d'édition d'en-tête HTTP.

L'injection de code SQL se produit lorsqu'un auteur de menace ajoute un code SQL dans le champ d'entrée pour afficher ou modifier des données de la base de données. Grâce à cette méthode d'attaque, un auteur de menace peut modifier ou détruire une base de données. L'auteur de menace n'a besoin que de compétences SQL de base pour exécuter l'attaque. Pour atténuer ce type d'attaque, il vous faut bloquer les mécanismes d'échappement (p. ex. les guillemets ou les guillemets doubles) et les caractères spéciaux associés au code SQL (p. ex. SUPPRIMER, METTRE À JOUR) lorsque cela est possible et selon les autorisations de l'application Web.

7.6 CACHER AUX UTILISATEURS LES MESSAGES D'ERREUR SQL

Les messages d'erreur SQL peuvent fournir de nombreuses données relatives à votre base de données, notamment des demandes d'information pour des champs et des bases de données spécifiques. Ne permettez pas aux utilisateurs de voir les messages d'erreur, car cela peut donner des indications aux auteurs de menace quant à savoir à quoi ressemble votre base de données.

7.7 BLOQUER LES INSTANCES À PLUSIEURS PARAMÈTRES

Dans la mesure du possible, veuillez bloquer les instances à plusieurs paramètres afin de filtrer les demandes avant que le serveur d'applications les traite. La mise en œuvre dans le code dépend de la plateforme d'application que vous utilisez. Certaines plateformes ont une protection implicite contre les paramètres multiples du même nom. Vous pouvez également limiter vos entrées de la partie frontale et utiliser un coupe-feu WAF pour bloquer certains mots-clés propres à des instructions SQL.

Le protocole HTTP permet l'utilisation de plusieurs paramètres ayant le même nom; ainsi, des valeurs multiples du même nom de paramètre peuvent se retrouver dans une seule demande. Cette procédure est gérée différemment selon les plateformes de serveur (p. ex. ASP.NET, JSP, PHP), en utilisant les deux valeurs ou simplement la première valeur ou la seconde valeur, ce qui peut prêter à confusion. Le comportement des applications peut changer, et la validation d'entrée peut être contournée.

Il est recommandé de tester ces validations durant la phase de développement au moyen de révisions de code et d'essais de développement.

7.8 VALIDER APRÈS L'ENCODAGE

Vous devriez définir l'encodage sur les pages Web et vous assurer que toutes les pages prennent en charge le même encodage. Si possible, désactivez le mappage ajusté et normalisez les données par encodage avant de valider les données. Vos clients et vos plateformes logicielles peuvent se trouver à travers le monde. Ainsi, il peut y avoir confusion entre des

caractères appartenant à des langues différentes qui peuvent se ressembler beaucoup, mais dont la signification est complètement différente pour votre application.

Utilisez, si possible, le format transformé d'Unicode 8 (UTF-8) de la convention Unicode Standard. Le codage UTF-8 est le plus répandu dans le monde pour les pages Web, et il prend en charge la compatibilité descendante. Actuellement, la plus récente version du codage UTF-8 est la version v.12.1. Utilisez des bibliothèques à normalisation standard. La plupart des fournisseurs de logiciels modernes utilisent Unicode comme langage universel pour identifier clairement les caractères, pendant que les données sont traitées dans différentes plateformes et divers dispositifs, pour empêcher l'altération des données.

Effectuez des essais dans le cadre du processus de développement, en utilisant les révisions de code et les essais à l'exécution.

7.9 SÉCURISER LE TÉLÉVERSEMENT DE FICHIERS

Le téléversement de fichiers est une forme d'entrée dans votre application Web. Dans de nombreux cas, le téléversement de fichiers peut s'inscrire dans le cadre du contenu de votre site Web. Il y a un risque que le téléversement puisse comporter un contenu exécutable ou un certain type de maliciel. De plus, un téléversement plus volumineux que prévu peut entraîner des problèmes de disponibilité pour votre application Web.

Vous devez chercher à savoir pourquoi le téléversement du fichier est nécessaire et s'il existe d'autres formats pouvant prendre en charge ces données, comme des champs textes et zones de texte. Bien que ces formats de rechange nécessitent plus d'entrées de l'utilisateur final, ils assurent un stockage plus sécuritaire. De plus, il est plus facile d'effectuer des recherches sur des données lorsqu'elles sont entrées dans ces formats, car le contenu demeure conforme (p. ex. en mode texte). Vous pourriez par exemple songer à une demande d'emploi. L'entrée par formulaire offre plus d'avantages à l'examinateur, car il peut effectuer des recherches sur des mots-clés, et au candidat, car il n'a pas à s'inquiéter que le logiciel de lecture ne soit pas en mesure d'analyser le contenu de son CV.

S'il est nécessaire de téléverser un fichier, vous devriez filtrer le téléchargement de ce fichier en fonction de l'extension de fichier et des limites de taille, et analyser le contenu. Vous pouvez utiliser la commande « fichier », qui vérifie les types de fichiers et permet de comprendre le type de contenu. Pour extraire les métadonnées ou le contenu pour effectuer une inspection plus approfondie, vous pouvez également utiliser des outils et des bibliothèques complémentaires pour l'extraction du fichier en format texte ou Extensible Markup Language (XML). Pour ce qui est des graphiques, vous pouvez utiliser des outils pour supprimer des métadonnées et détecter des anomalies.

7.10 TESTER LA LOGIQUE MÉTIER

La logique métier fait référence aux décisions opérationnelles de votre organisation et à la façon selon laquelle ces décisions sont traduites en logique de codage sur le serveur d'applications. Des risques peuvent être engendrés si les décisions opérationnelles ne sont pas claires ou ne sont pas traduites adéquatement pour la logique de codage. Des failles dans la logique métier peuvent donner accès à des entrées erronées dans votre espace, ce qui peut occasionner bon nombre de problèmes.

Vous devez bien définir vos exigences et cas d'utilisation, y compris les cas d'utilisation malveillante. Lorsque vous définissez des exigences liées à la sécurité, analysez le processus opérationnel dans son ensemble. Testez la logique métier pendant et après la phase de développement pour régler les cas d'utilisation et d'utilisation malveillante. Vous devez vous assurer que les cas d'utilisation règlent toutes occurrences d'accès simultanés, qui se produisent lorsque plusieurs processus doivent avoir accès à la même ressource au même moment.

7.11 EFFECTUER DES TESTS DE PÉNÉTRATION

Les tests de pénétration sont nécessaires pour déterminer jusqu'à quel point votre validation d'entrée est résiliente face à des activités malveillantes. Les tests de pénétration, ou le piratage contrôlé, impliquent des testeurs qui sont autorisés à tenter de trouver et d'exploiter les vulnérabilités de sécurité. Il existe différents types de tests de pénétration (p. ex. les testeurs reçoivent plus ou moins d'informations sur vos systèmes); toutefois, vos testeurs ne devraient pas faire partie de votre équipe de développement pour qu'ils puissent simuler les comportements des auteurs de menace.

8 CONFIGURATION SÉCURITAIRE

La configuration sécuritaire est un sujet très vaste, plus particulièrement lorsque l'on tient compte des configurations de sécurité sur des technologies individuelles. Le présent document se consacre aux configurations générales spécifiquement pour assurer la sécurité des services Web. Si vous cherchez des recommandations relatives aux configurations sécuritaires, nous vous recommandons de communiquer avec le fournisseur. Les configurations de sécurité recommandées par un fournisseur sont habituellement de bonnes données de référence que vous pouvez appliquer et ensuite adapter aux besoins de votre organisation. Pour obtenir de plus amples renseignements sur la pratique exemplaire en matière de configuration sécurisée générique, consultez les diverses publications proposées sur notre site Web (www.cyber.gc.ca).

8.1 DÉSACTIVER L'EXPLORATION DES RÉPERTOIRES

Pour éviter de couler de l'information potentiellement sensible aux utilisateurs, désactivez l'option d'exploration des répertoires qui se trouve sur votre serveur Web. L'exploration de répertoires se produit lorsque vous pouvez voir et accéder aux dossiers ainsi qu'aux fichiers qui composent le site Web plutôt que de le voir comme il devrait être présenté aux utilisateurs. Ces dossiers contiennent de l'information qui n'est pas censée être vue par le public, notamment des images, des scripts ou des sauvegardes. L'exploration des répertoires est habituellement mise en œuvre par répertoire, et vous devez désactiver cette fonction pour chacun des répertoires.

Il vous faut également créer un fichier index.htm vide dans chaque répertoire. Si le fichier de base, comme index.htm ou index.php n'est pas disponible, alors par défaut, n'importe qui peut voir tous les fichiers et les sous-répertoires énumérés dans le navigateur. Testez chaque répertoire et sous-répertoire.

8.2 SUPPRIMER LES FICHIERS DU RÉPERTOIRE WEB INUTILES

Dans le cadre de votre politique en matière de déploiement sécurisé, vous devez passer en revue vos répertoires Web avant et après le déploiement de votre site Web afin d'atténuer les risques associés aux fuites de données. Les répertoires Web peuvent contenir de l'information qui n'est pas censée être vue par le public, notamment des images, des scripts ou des sauvegardes.

Il vous faut également définir des exigences pour établir le cycle de vie des fichiers répertoires Web. Et vous devez aussi supprimer des répertoires Web les fichiers d'opérations Web inutiles. Ciblez les anciennes versions des fichiers (multiples fichiers default.asp), code source (.bak), fichiers de sauvegarde (pouvant contenir des mots de passe) et fichiers de contrôle de source (.svn et .git). Afin de faciliter l'identification de ces fichiers, vous pouvez activer l'option *Heure du dernier accès* sur le système de fichiers, ce qui vous donne un accès visuel aux fichiers rarement utilisés, voire jamais. Faites une vérification lors de vos révisions de code.

8.3 VERROUILLER LES COMPOSANTS DE SERVICES WEB

Afin de réduire votre exposition aux menaces, vous devez verrouiller différents composants de vos services Web.

Vous devriez privilégier le verrouillage, ou le renforcement, de vos hôtes de service Web en supprimant les ports, les fichiers et les codes inutilisés, ainsi que les comptes inutilisés à cycle de vie. Si vos serveurs Web restent exposés, il y a un risque

accru que les auteurs de menace puissent exploiter ces ports et services. Ces mesures peuvent améliorer les protections offertes par les coupe-feux et les politiques de sécurité.

Vous devriez également supprimer tout module d'extension dans vos systèmes de gestion de la configuration (CMS pour *Configuration Management System*). Plus vous avez des modules d'extension, plus grande est votre surface d'attaque, plus particulièrement si les modules d'extension ne constituent pas des modules officiels.

De plus, renforcez le serveur SQL et supprimez toutes les procédures stockées qui sont inutilisées ainsi que les comptes par défaut qui se trouvent dans votre base de données. Surveillez la connexion sortante du serveur SQL pour détecter une intrusion.

8.4 DÉSACTIVER LA MISE EN CACHE DES JUSTIFICATIFS D'IDENTITÉ

Les navigateurs donnent la possibilité d'enregistrer les mots de passe des utilisateurs pour des raisons de commodité. Il faut toutefois noter que le stockage de mots de passe sur le navigateur de l'utilisateur n'est pas sécuritaire. En cas de vol du dispositif mobile ou de l'ordinateur portable d'un utilisateur, et si un auteur de menace arrive à accéder au site Web, les justificatifs de cet utilisateur seront automatiquement affichés. Lors du codage en format HTML, assurez-vous de désactiver l'élément *Remplissage autom. (autocomplete)*, qui indique au navigateur de désactiver la mise en cache des informations d'identification pour ce formulaire.

8.5 UTILISER DES SCANNEURS DE VULNÉRABILITÉ

Vous devriez intégrer l'analyse des vulnérabilités à votre processus de déploiement sécurisé. Des outils tels que Zed Attack Proxy (ZAP) de source ouverte de l'OWASP sont couramment utilisés pour identifier des vulnérabilités propres aux applications Web. Vous pouvez donc avoir recours à ces outils pour trouver des vulnérabilités, qui peuvent également servir aux testeurs de pénétration pour déterminer les points d'entrée dans vos services Web.

8.6 AUTOMATISER LE DÉPLOIEMENT

Il peut s'avérer compliquer de gérer les configurations sécuritaires de toutes les sections mobiles de votre site Web. Une pratique exemplaire consiste à automatiser les environnements. Cette pratique permet de s'assurer que les environnements sont conçus de façon répétable (p. ex. ils peuvent être redéployés au besoin) et automatisée (p. ex. Sans intervention humaine). C'est à vous que revient la tâche de définir les rôles et les responsabilités quant à l'exécution de la gestion de la configuration et de la vérification.

Vous pouvez utiliser des conteneurs et des modèles de configurations de sécurité. Pour restreindre la surface d'attaque, vous devriez créer un conteneur dans lequel on met uniquement une image basée sur un serveur Web. En utilisant des modèles et des conteneurs, vous pouvez vous assurer que les nouvelles versions de serveur Web ne comportent pas de bogues du jour zéro. Lors du déploiement, rappelez-vous de désactiver les fonctionnalités de connexion à distance pour que le protocole SSH ne soit pas autorisé en production.

9 SÉCURISER LES OPÉRATIONS

Une fois vos services Web établis, vous devez continuellement en assurer la sécurité. Vos méthodes de surveillance doivent évoluer au rythme des changements technologiques et à mesure que vous découvrez de nouvelles cybermenaces. Passez en revue périodiquement vos modèles de contrôle d'accès pour vous assurer qu'ils reflètent les changements apportés à l'accès au compte (p. ex. lors des changements de personnel). Vous devez valider de nouveau les données d'entrée lorsque vous apportez des changements à votre site Web et passer en revue vos configurations de sécurité pour chacune des nouvelles versions provenant du fournisseur.

Les auteurs de menace peuvent se faire passer pour votre organisation pour cibler vos clients. L'hameçonnage, le rançongiciel et le piratage psychologique sont des attaques très répandues. Ces auteurs peuvent aussi fournir de faux renseignements pour recueillir de l'information de vos clients ou les diriger vers un site Web ou un lien frauduleux ou faux qui injecte un code malveillant dans leurs systèmes. Une mystification peut avoir des répercussions négatives sur votre présence sur le Web et votre réputation. L'hameçonnage, le rançongiciel et le piratage psychologique sont des attaques très répandues.

9.1 EFFECTUER DES ACTIVITÉS DE SURVEILLANCE

Définissez votre stratégie de surveillance de façon à identifier des attaques potentielles et des anomalies propres aux risques liés aux applications Web mentionnés dans le projet *OWASP Top 10* [4]. Dans le cadre des activités de surveillance pour votre application Web, tenez compte des composants de l'architecture Web mentionnés à la Section 3 du présent document. Surveillez le trafic sur votre réseau ainsi que toutes tentatives infructueuses de connexion. Analysez les registres et avisez les parties concernées de toutes activités inhabituelles en temps opportun pour ainsi assurer d'une intervention efficace.

9.2 ÉLABORER UN PLAN D'INTERVENTION EN CAS D'INCIDENT

Élaborez un plan d'intervention en cas d'incident, qui comprend des guides opérationnels sur les incidents, afin de définir les interventions que votre organisation dot mettre en œuvre lors de la détection d'un incident. Par exemple, vous devriez inclure des étapes permettant d'identifier, de communiquer, de contenir et de corriger les incidents. Il vous faut bien définir les rôles et les responsabilités pour être en mesure d'intervenir en cas d'incident, plus particulièrement si plusieurs équipes ou organisations sont impliquées dans la prise en charge de vos exigences de surveillance. Ce plan doit être rigoureusement testé pour assurer son efficacité.

Si vous faites appel à un FSG ou à un FSI, travaillez de concert avec votre fournisseur de services pour établir un plan commun d'intervention en cas d'incident et assurez-vous d'avoir accès aux données appropriées et d'être avisé adéquatement. Assurez-vous de coordonner les essais avec votre fournisseur de services. Identifiez l'information reçue et déterminez des délais raisonnables pour vous permettre d'intervenir judicieusement.

Pour obtenir une liste complète des fonctionnalités de surveillance de la sécurité, consultez le catalogue de contrôle de la sécurité à l'Annexe 3a du guide ITSG-33 [3], et plus particulièrement les familles de contrôle de la sécurité : vérification et imputabilité (AU pour *auditing and accountability*) et intervention en cas d'incident (IR pour *incident response*).

9.3 ÉTABLIR UNE STRATÉGIE D'APPLICATION DE CORRECTIFS

La gestion des correctifs est le processus qui permet d'acquérir, de tester et d'installer des correctifs et des mises à niveau pour vos systèmes. La stratégie de gestion des correctifs pour vos applications Web sera semblable à la stratégie d'application de correctifs du reste de votre infrastructure. Toutefois, vous devez tenir compte et établir l'ordre de priorité des composants de l'architecture qui sont exposés à Internet.

Il est recommandé de s'abonner à des publications sur les vulnérabilités et les applications (p. ex. la liste *Common Vulnerabilities and Exposures* [CVE pour vulnérabilités et expositions courantes] de MITRE) pour que vous puissiez identifier les vulnérabilités à mesure que celles-ci deviennent connues. Il existe plusieurs outils reconnus et à faible coût que vous pouvez utiliser pour analyser les vulnérabilités dans votre réseau. Tenez aussi compte des vulnérabilités dans les dépendances et les conteneurs de bibliothèque. Une autre option est de faire un balayage des composants logiciels anciens et désuets à mettre à jour. Si votre organisation n'est pas en mesure de mettre rapidement en œuvre des correctifs, réexaminez vos règles de coupe-feu pour réduire les risques.

Si vous utilisez des services gérés ou des services infonuagiques, travaillez en collaboration avec votre fournisseur de services pour appliquer des correctifs de sécurité et des mises à jour, et ce, en temps opportun pour respecter vos exigences. Pour ce qui est des services infonuagiques, l'entité responsable de l'application des correctifs dépend de votre modèle de service infonuagique (p. ex. dans un modèle de logiciel en tant que service, le FSG est chargé des mises à jour et de l'application des correctifs).

9.4 PROMOUVOIR LA SENSIBILISATION À LA SÉCURITÉ

Malgré l'évolution des technologies et des cybermenaces, vous pouvez continuer de protéger vos systèmes, vos renseignements et vos clients contre des compromissions en insistant sur la promotion de la sensibilisation à la sécurité. Il vous faut promouvoir les pratiques exemplaires comme le fait de ne pas cliquer sur des liens ou de ne pas télécharger des fichiers. Ces pratiques se font en changeant votre façon de communiquer avec vos clients. Par exemple, si vous devez communiquer avec un client pour discuter de son compte, ne joignez pas de lien ou de fichier téléchargeable dans votre communication. Demandez plutôt au client de se connecter à votre site Web ou de vous contacter à l'aide des coordonnées qui se trouvent sur votre site. Vous pouvez aussi lui suggérer de vérifier les communications que vous lui avez envoyées et de vous signaler toutes activités suspectes.

Vous pouvez également diriger les clients vers nos documents de la campagne Pensez cybersécurité qui sont conçus pour sensibiliser les Canadiens à la sécurité en ligne.

10 PARTENARIATS

Les avantages et les difficultés que pose la création de sites Web sécuritaires sont partagés par toutes les organisations présentes sur Internet. Vous trouverez ci-dessous des ressources que vous pouvez utiliser pour développer un site Web sécuritaire.

10.1 PARLER AVEC DES PAIRS

Quel que soit le secteur de l'industrie, les problèmes et les préoccupations auxquels vous êtes confrontés ne sont pas nécessairement uniques. On comprend bien qu'il n'est pas facile de discuter d'enjeux relatifs à la sécurité avec des concurrents. Cependant, il faut demeurer à l'affût des occasions qui se présentent pour discuter d'options de produits ou de fournisseurs, ainsi que des défis que pose le développement. Vous pourriez être agréablement surpris de constater à quel point vous pourriez en apprendre de l'expérience des autres.

10.2 OWASP

L'OWASP Foundation est une fondation sans but lucratif qui vise à améliorer la sécurité des logiciels. Vous êtes invité à consulter le projet OWASP Top 10 [4] pour connaître la liste des risques critiques de sécurité touchant les applications Web.

10.3 CENTRE ANTIFRAUDE DU CANADA

Si votre organisation a été victime de fraude, par exemple si un auteur de menace s'est fait passer pour votre organisation, communiquez avec le service de police le plus près de chez vous et faites un signalement en ligne en passant par le Système de signalement de cas de fraude au Centre antifraude du Canada.

10.4 CONTACTEZ-NOUS

Pour obtenir de plus amples renseignements sur la cybersécurité, visitez notre site Web (cyber.gc.ca) ou communiquez avec notre équipe des Services à la clientèle du Centre :

Centre d'appel
contact@cyber.gc.ca
613 949-7048 ou 1-833-CYBER-88

11 CONTENU COMPLÉMENTAIRE

11.1 LISTE DES ABRÉVIATIONS

Terme	Définition
AD	Système d'exploitation Active Directory
IPA	Interface de programmation d'applications
ASP	Langage de programmation Active Server Pages
AU	Vérification (*Auditing*) (famille de contrôle de sécurité)
CAPTCHA	Test de sécurité dont l'acronyme anglais est *Completely automated public Turing tests to tell computers and humans apart*
CMS	Système de gestion de la configuration (*Configuration Management System*)
UCT	Unité centrale de traitement
CSRF	Falsification de requête intersites (Cross-site request forgery)
FSI	Fournisseur de services infonuagiques
CVE	Vulnérabilités et expositions courantes (*Common vulnerabilities and exposures*)
DoS	Attaque par déni de service
GC	Gouvernement du Canada
E	Élevée (en termes de sensibilité des données)
HTTP	Protocole de transfert hypertexte (*Hypertext Transfer Protocol*)
HTTPS	Protocole Hypertext Transfer Protocol Secure
SDI	Système de détection d'intrusion
IP	Protocole IP
IPS	Système de prévention d'intrusion (*Intrusion prevention system*)
IR	Intervention en cas d'incident (*Incident Response*) (famille de contrôle de sécurité)
TI	Technologies de l'information
F	Faible (en termes de sensibilité des données)
LDAP	Protocole Lightweight Directory Access Protocol
M	Moyenne (en termes de sensibilité des données)
MFA	Authentification multifacteur (*Multi-factor authentication*)
FSG	Fournisseur de services gérés
NAT	Traduction d'adresses de réseau (*Network Address Translation*)
NIST	National Institute of Standards and Technology (États-Unis)
OWASP	Communauté Open Web Application Security Project

Terme	Définition
PHP	Langage de programmation Hypertext Preprocessor
NIP	Numéro d'identification personnel
SQL	Langage Structured Query Language
SSL	Protocole Secure Sockets Layer
TCP/IP	Protocoles Transmission Control Protocol/Internet Protocol
TLS	Protocole de sécurité de la couche de transport (*Transport Layer Security*)
TLS/SSL	Protocole de sécurité de la couche de transport et protocole de couche de sockets sécurisés (*Transport Layer Security/Secure Sockets Layer*)
URL	Localisateur de ressources uniforme (U*niform resource locator*)
RPV	Réseau privé virtuel
WAF	Coupe-feu d'applications Web (*Web application firewall*)
XML	Langage de balisage extensible (*Extensible Markup Language*)
XSS	Script intersites (*Cross-site scripting*)
ZAP	Outil Zed Attack Proxy (scanneur de sécurité OWASP)

11.2 GLOSSAIRE

Terme	Définition
Contrôle d'accès	Attestation confirmant que seul un accès autorisé est donné aux biens (tant physiques qu'électroniques). Pour les biens de TI, les contrôles d'accès peuvent être nécessaires pour les réseaux, les systèmes et l'information (p. ex. limiter l'accès à certains systèmes à des utilisateurs, limiter les privilèges du compte).
Privilèges administratifs	Les autorisations qui permettent à un utilisateur d'exécuter certaines fonctions sur un système ou un réseau, comme l'installation d'un logiciel et la modification de paramètres de configuration.
Architecture	La disposition des services Web et de leurs composants sous-jacents pour assurer un service sécuritaire et efficace.
Authentification	Un processus ou une mesure visant à vérifier l'identité d'un utilisateur.
Disponibilité	La capacité pour les bonnes personnes à accéder à la bonne information ou aux bons systèmes lorsque cela est nécessaire. La disponibilité est appliquée aux actifs informationnels, aux logiciels et aux matériels (l'infrastructure et ses composantes). La définition implique que la disponibilité comprend la protection des biens contre un accès non autorisé et une compromission.
Injection de code	Insertion d'un code malveillant dans un programme informatique en exploitant une faille dans le programme ou dans la façon dont ce programme interprète les données saisies par les utilisateurs.
Compromission	La divulgation intentionnelle ou non d'information qui a une incidence négative sur sa confidentialité, son intégrité ou sa disponibilité.
Confidentialité	La capacité à protéger de l'information sensible pour en empêcher l'accès à des personnes non autorisées.
Témoin	Un paquet de données qui contient l'information sur la session de l'utilisateur.
Cyberattaque	L'utilisation de supports électroniques pour interrompre, manipuler, détruire, ou obtenir un accès à un système informatique, à un réseau ou à un dispositif.
Attaque par déni de service	Toute activité qui rend un système inaccessible aux utilisateurs légitimes ou qui provoque des retards aux opérations et aux fonctions du système.
Chiffrement	Transformation de données d'un format vers un autre pour cacher leur contenu et empêcher un accès non autorisé.
Hachage	Processus par lequel un algorithme mathématique est utilisé à partir de données pour produire une valeur numérique qui est représentative de ces données [10].
Site Web hôte	Il héberge tous les fichiers nécessaires à la gestion d'un site Web et il assure la connexion à Internet.
Validation d'entrée	Le processus qui permet de vérifier que les utilisateurs et les applications n'entrent que des données correctement formées, comme dans les champs, les formulaires ou les demandes d'information.
Intégrité	La capacité à protéger de l'information pour qu'elle ne puisse pas être modifiée ou supprimée par inadvertance ou lorsqu'il ne le faut pas. L'intégrité aide à confirmer que l'information est ce qu'elle prétend être.
Droit d'accès minimal	Le principe en vertu duquel une personne ne reçoit que l'ensemble des privilèges dont elle a besoin pour accomplir des tâches autorisées. Ce principe permet de limiter les dommages que peut causer une utilisation accidentelle, incorrecte ou non autorisée d'un système d'information.
Authentification multifacteur	Une forme d'authentification qui exige au moins deux facteurs d'authentification différents pour vérifier l'identité affirmée. Les trois facteurs les plus répandus sont : (1) quelque chose que vous savez (p. ex. un mot de passe),

Terme	Définition
	(2) quelque chose que vous avez (p. ex. un jeton d'authentification physique) et (3) quelque chose que vous êtes (p. ex. une biométrie).
Tests de pénétration	Un test au moyen duquel des spécialistes du piratage contrôlé sont autorisés à tenter de trouver et d'exploiter des vulnérabilités se trouvant dans les réseaux ou les systèmes d'une organisation.
Redondance	Un principe architectural qui s'assure que diverses ressources remplissent la même fonction pour ainsi maintenir la disponibilité des systèmes et des services.
Risque résiduel	La probabilité et l'incidence qu'une menace persiste, même après que les contrôles de sécurité aient été implantés.
Session	Un échange d'information entre au moins deux entités, comme deux dispositifs ou un utilisateur et un serveur Web.
Gestion de session	Le processus d'amorce, de contrôle, de maintien et d'arrêt des échanges entre au moins deux entités. Lors d'une session, un identifiant unique est marqué pour toutes les demandes HTTP faites par un utilisateur à partir du moment où il se connecte jusqu'à ce qu'il mette fin à la session. La gestion de session autorise au serveur à reconnaître l'utilisateur entre les demandes.
Authentification à deux facteurs	Un type d'authentification multifacteur utilisé pour confirmer l'identité d'un utilisateur.
Intégrité par deux personnes	Une exigence selon laquelle au moins deux personnes autorisées doivent être en mesure de détecter des procédures de sécurité incorrectes ou non autorisées en rapport à la tâche effectuée [10].
Réseau privé virtuel	Un réseau de communication privé généralement utilisé au sein d'une entreprise, ou par plusieurs entreprises ou organisations différentes pour communiquer par un réseau plus vaste. Les communications RPV sont habituellement chiffrées ou encodées afin d'isoler les échanges du trafic d'utilisateurs se déroulant sur des réseaux publics.

11.3 RÉFÉRENCES

Numéro	Référence
1	Centre canadien pour la cybersécurité, *Facteurs à considérer par les clients de services gérés en matière de cybersécurité (ITSM.50.030)*, octobre 2020.
2	Centre canadien pour la cybersécurité, *Gestion des risques liés à la sécurité infonuagique (ITSM.50.062)*, mars 2019.
3	Centre canadien pour la cybersécurité, *ITSG-33 IT – La gestion des risques liés à la sécurité des TI : Une méthode axée sur le cycle de vie*, novembre 2012.
4	Open Web Application Security Project, *Top 10 Web Application Security Risks*, 2017.
5	Centre canadien pour la cybersécurité, *Pratiques exemplaires de création de phrases de passe et de mots de passe (ITSAP.30.032)*, septembre 2019.
6	Centre canadien pour la cybersécurité, *Conseils de sécurité sur les gestionnaires de mots de passe (ITSAP.30.025)*, septembre 2019.
7	Centre canadien pour la cybersécurité, *Guide sur l'authentification des utilisateurs dans les systèmes de technologie de l'information (ITSP.30.031 v3)*, avril 2018.
8	Centre canadien pour la cybersécurité, *Algorithmes cryptographiques pour l'information NON CLASSIFIÉ, PROTÉGÉ A et PROTÉGÉ B (ITSP.40.111)*, août 2016.
9	National Institute of Standards and Technology, *Special Publication 800-63-3 Digital Identity Guidelines*, juin 2017.
10	National Institute of Standards and Technology, National Vulnerability Database. *CVE-2014-0160 Detail*, avril 2014.
11	National Institute of Standards and Technology, Computer Security Resource Centre. *Glossary*, sans date.