Union internationale des télécommunications

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

X.1211

(09/2014)

SÉRIE X: RÉSEAUX DE DONNÉES, COMMUNICATION ENTRE SYSTÈMES OUVERTS ET SÉCURITÉ

Sécurité du cyberespace – Cybersécurité

Techniques pour prévenir les attaques sur le web

Recommandation UIT-T X.1211

RECOMMANDATIONS UIT-T DE LA SÉRIE X

RÉSEAUX DE DONNÉES, COMMUNICATION ENTRE SYSTÈMES OUVERTS ET SÉCURITÉ

RÉSEAUX PUBLICS DE DONNÉES	X.1–X.199
INTERCONNEXION DES SYSTÈMES OUVERTS	X.200–X.299
INTERFONCTIONNEMENT DES RÉSEAUX	X.300–X.399
SYSTÈMES DE MESSAGERIE	X.400–X.499
ANNUAIRE	X.500–X.599
RÉSEAUTAGE OSI ET ASPECTS SYSTÈMES	X.600–X.699
GESTION OSI	X.700–X.799
SÉCURITÉ	X.800–X.849
APPLICATIONS OSI	X.850–X.899
TRAITEMENT RÉPARTI OUVERT	X.900–X.999
SÉCURITÉ DE L'INFORMATION ET DES RÉSEAUX	
Aspects généraux de la sécurité	X.1000–X.1029
Sécurité des réseaux	X.1030–X.1049
Gestion de la sécurité	X.1050–X.1069
Télébiométrie	X.1080–X.1099
APPLICATIONS ET SERVICES SÉCURISÉS	
Sécurité en multidiffusion	X.1100–X.1109
Sécurité des réseaux domestiques	X.1110–X.1119
Sécurité des télécommunications mobiles	X.1120–X.1139
Sécurité de la toile	X.1140–X.1149
Protocoles de sécurité	X.1150–X.1159
Sécurité d'homologue à homologue	X.1160–X.1169
Sécurité des identificateurs en réseau	X.1170–X.1179
Sécurité de la télévision par réseau IP	X.1180–X.1199
SÉCURITÉ DU CYBERESPACE	
Cybersécurité	**X.1200–X.1229**
Lutte contre le pollupostage	X.1230–X.1249
Gestion des identités	X.1250–X.1279
APPLICATIONS ET SERVICES SÉCURISÉS	
Communications d'urgence	X.1300–X.1309
Sécurité des réseaux de capteurs ubiquitaires	X.1310–X.1339
ECHANGE D'INFORMATIONS SUR LA CYBERSÉCURITÉ	
Aperçu général de la cybersécurité	X.1500–X.1519
Echange concernant les vulnérabilités/les états	X.1520–X.1539
Echange concernant les événements/les incidents/l'heuristique	X.1540–X.1549
Echange de politiques	X.1550–X.1559
Heuristique et demande d'informations	X.1560–X.1569
Identification et découverte	X.1570–X.1579
Echange garanti	X.1580–X.1589
SÉCURITÉ DE L'INFORMATIQUE EN NUAGE	
Aperçu de la sécurité de l'informatique en nuage	X.1600-X.1601
Conception de la sécurité de l'informatique en nuage	X.1602-X.1639
Bonnes pratiques et lignes directrices concernant la sécurité de l'informatique en nuage	X.1640-X.1659
Mise en oeuvre de la sécurité de l'informatique en nuage	X.1660-X.1679
Sécurité de l'informatique en nuage (autres)	X.1680-X.1699

Pour plus de détails, voir la Liste des Recommandations de l'UIT-T.

Recommandation UIT-T X.1211

Techniques pour prévenir les attaques sur le web

Résumé

La Recommandation UIT-T X.1211 décrit les techniques pouvant atténuer les attaques sur le web qui se produisent lorsque les vulnérabilités des serveurs d'hébergement de site web sont exploitées et que des codes malveillants sont injectés, lesquels peuvent ensuite infecter les ordinateurs des utilisateurs. Plusieurs appendices décrivent ces attaques et les mesures à prendre pour y faire face.

Historique

Edition	Recommandation	Approbation	Commission d'études	ID unique*
1.0	ITU-T X.1211	2014-09-26	17	11.1002/1000/12154

Mots clés

Attaque sur le web, contenu suspect, injection SQL, logiciel espion, prévention, vulnérabilité.

* Pour accéder à la Recommandation, reporter cet URL http://handle.itu.int/ dans votre navigateur Web, suivi de l'identifiant unique, par exemple http://handle.itu.int/11.1002/1000/11830-en.

AVANT-PROPOS

L'Union internationale des télécommunications (UIT) est une institution spécialisée des Nations Unies dans le domaine des télécommunications et des technologies de l'information et de la communication (ICT). Le Secteur de la normalisation des télécommunications (UIT-T) est un organe permanent de l'UIT. Il est chargé de l'étude des questions techniques, d'exploitation et de tarification, et émet à ce sujet des Recommandations en vue de la normalisation des télécommunications à l'échelle mondiale.

L'Assemblée mondiale de normalisation des télécommunications (AMNT), qui se réunit tous les quatre ans, détermine les thèmes d'étude à traiter par les Commissions d'études de l'UIT-T, lesquelles élaborent en retour des Recommandations sur ces thèmes.

L'approbation des Recommandations par les Membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution 1 de l'AMNT.

Dans certains secteurs des technologies de l'information qui correspondent à la sphère de compétence de l'UIT-T, les normes nécessaires se préparent en collaboration avec l'ISO et la CEI.

NOTE

Dans la présente Recommandation, l'expression "Administration" est utilisée pour désigner de façon abrégée aussi bien une administration de télécommunications qu'une exploitation reconnue.

Le respect de cette Recommandation se fait à titre volontaire. Cependant, il se peut que la Recommandation contienne certaines dispositions obligatoires (pour assurer, par exemple, l'interopérabilité et l'applicabilité) et considère que la Recommandation est respectée lorsque toutes ces dispositions sont observées. Le futur d'obligation et les autres moyens d'expression de l'obligation comme le verbe "devoir" ainsi que leurs formes négatives servent à énoncer des prescriptions. L'utilisation de ces formes ne signifie pas qu'il est obligatoire de respecter la Recommandation.

DROITS DE PROPRIÉTÉ INTELLECTUELLE

L'UIT attire l'attention sur la possibilité que l'application ou la mise en œuvre de la présente Recommandation puisse donner lieu à l'utilisation d'un droit de propriété intellectuelle. L'UIT ne prend pas position en ce qui concerne l'existence, la validité ou l'applicabilité des droits de propriété intellectuelle, qu'ils soient revendiqués par un membre de l'UIT ou par une tierce partie étrangère à la procédure d'élaboration des Recommandations.

A la date d'approbation de la présente Recommandation, l'UIT n'avait pas été avisée de l'existence d'une propriété intellectuelle protégée par des brevets à acquérir pour mettre en œuvre la présente Recommandation. Toutefois, comme il ne s'agit peut-être pas de renseignements les plus récents, il est vivement recommandé aux développeurs de consulter la base de données des brevets du TSB sous http://www.itu.int/ITU-T/ipr/.

TABLE DES MATIÈRES

Recommandation UIT-T X.1211

Techniques pour prévenir les attaques sur le web

1 Domaine d'application

La présente Recommandation traite des techniques pour prévenir les attaques sur le web. Elle décrit les scénarios utilisés pour distribuer des logiciels malveillants via le web ainsi que les techniques fonctionnelles et les fonctions à utiliser pour prévenir les attaques sur le web.

2 Références

Aucune.

3 Termes et définitions

3.1 Termes définis ailleurs

La présente Recommandation emploie les termes suivants définis ailleurs:

3.1.1 actif [b-ISO/CEI 27000]: tout ce qui a de la valeur pour l'organisation.

NOTE – Il existe de nombreux types d'actifs, à savoir:

a) les informations;

b) les logiciels, par exemple les programmes informatiques;

c) les matériels, par exemple les ordinateurs;

d) les services;

e) les personnes, et leurs qualifications, compétences et expérience; et

f) les biens immatériels, par exemple la réputation et l'image.

3.1.2 instance d'attaque [b-UIT-T X.1544]: attaque précise contre une application ou un système donné, qui cible les vulnérabilités ou les failles de ce système.

3.1.3 schéma d'attaque [b-UIT-T X.1544]: représentation abstraite des méthodes d'attaque courantes observées dans le monde contre des applications ou des systèmes (par exemple, l'injection SQL, "l'attaque de l'homme du milieu" ou le détournement de session, etc.).

NOTE – Un même schéma d'attaque peut être associé à de nombreuses instances d'attaque différentes.

3.1.4 langage de balisage hypertexte (HTML, *hypertext markup language*) [b-UIT-T M.3030]: système de codage d'informations très diverses (textes, graphiques, résultats de consultation de bases de données) destinées à être affichées par les navigateurs web. Certains codes spéciaux, appelés balises, sont intégrés dans le document, afin que le navigateur sache comment restituer les informations.

3.1.5 logiciel malveillant [b-ISO/CEI 27033-1]: logiciel conçu expressément pour endommager ou perturber un système, et nuire à la confidentialité, à l'intégrité et/ou à la disponibilité.

3.1.6 technique d'obfuscation [b-NIST SP 800-83]: moyen utilisé pour construire un virus qui soit difficile à détecter.

3.1.7 informations d'identification personnelle (PII, *personally identifiable information*) [b-UIT-T X.1252]: toute information: a) identifiant ou permettant d'identifier la personne à laquelle elle se rapporte, de se mettre en rapport avec elle ou de la localiser; b) permettant d'obtenir des informations d'identification ou les coordonnées d'une personne; ou c) étant ou pouvant être directement ou indirectement liée à une personne physique.

3.1.8 menace [b-UIT-T X.800]: violation potentielle de la sécurité.

3.1.9 domaine de sécurité [b-UIT-T T.411]: ensemble de ressources associé à une politique de sécurité unique.

3.1.10 autorité de domaine de sécurité [b-UIT-T X.810]: autorité de sécurité qui est responsable de la mise en oeuvre d'une politique de sécurité pour un domaine de sécurité.

3.1.11 politique de sécurité [b-UIT-T T.411]: ensemble de règles qui spécifient les procédures et services nécessaires pour maintenir le niveau voulu de sécurité d'un ensemble de ressources.

3.1.12 signature [b-NIST SP 800-83]: ensemble de caractéristiques d'instances connues de logiciels malveillants qui peut être utilisé pour identifier des logiciels malveillants connus et certaines nouvelles variantes de logiciels malveillants connus.

3.1.13 logiciel espion [b-NIST SP 800-83]: logiciel malveillant conçu pour porter atteinte à la vie privée d'un utilisateur.

3.1.14 extension de navigateur web [b-NIST SP 800-83]: mécanisme utilisé pour afficher ou exécuter certains types de contenu via un navigateur web.

3.2 Termes définis dans la présente Recommandation

Les termes suivants sont définis dans la présente Recommandation:

3.2.1 anomalie: motif dans les données qui n'est pas conforme au comportement attendu.

3.2.2 téléchargement intempestif (*drive-by download*): schéma d'attaque sur le web qui est perpétré lorsqu'un utilisateur consulte un site web et qui exploite les vulnérabilités du navigateur pour lancer le téléchargement et l'installation automatiques de logiciels malveillants à l'insu de l'utilisateur ou sans sa permission.

3.2.3 attaque sur le web: schéma d'attaque qui est perpétré contre des sites web légitimes, entraînant l'injection dans une application de codes malveillants, lesquels peuvent ensuite infecter les ordinateurs des utilisateurs qui consultent ces sites web, ou qui utilise les vulnérabilités des sites web pour lancer des attaques contre les systèmes informatiques des utilisateurs qui consultent ces sites web, ce qui se produit sans intervention de logiciels malveillants.

3.2.4 système de protection contre les attaques sur le web: ensemble de systèmes qui détecte les vulnérabilités, les logiciels malveillants ou les codes malveillants intégrés dans un site web légitime et informe l'administrateur web du résultat de la détection, l'objectif ultime étant de les supprimer.

NOTE – Les activités de détection peuvent être programmées ou peuvent être déclenchées par des événements survenus dans le réseau ou par des demandes émanant d'autres systèmes.

3.2.5 ordinateur zombie: ordinateur qui a été compromis et dont le contrôle a été pris par un attaquant qui a installé des logiciels malveillants tels que des virus informatiques, des chevaux de Troie ou des botnets, lesquels peuvent être utilisés pour perpétrer des attaques malveillantes, comme la diffusion de spams de courrier électronique ou le lancement d'attaques de type déni de service.

4 Abréviations et acronymes

La présente Recommandation utilise les abréviations et acronymes suivants:

CAPEC liste et classification des schémas d'attaque courants (*common attack pattern enumeration and classification*)

CSRF falsification de requête intersites (*cross-site request forgery*)

CWE liste des failles courantes (*common weakness enumeration*)

DDoS déni de service réparti (*distributed denial of service*)

DOM modèle d'objet de document (*document object model*)

HTML langage de balisage hypertexte (*hypertext markup language*)

HTTP protocole de transfert hypertexte (*hypertext transfer protocol*)

ID identité

IODEF format d'échange de descriptions d'objet concernant les incidents (*incident object description exchange format*)

LDAP protocole simple d'accès à l'annuaire (*lightweight directory access protocol*)

MITM intercepteur (*man-in-the-middle*)

OS système d'exploitation (*operating system*)

OWASP projet ouvert concernant la sécurité des applications web (*open web applications security project*)

PC ordinateur personnel (*personal computer*)

PII informations d'identification personnelle (*personally identifiable information*)

PUI programme considéré (*program under inspection*)

SNS service de réseau social (*social network service*)

SQL langage de requête structuré (*structured query language*)

SSRF falsification de requête côté serveur (*server-side request forgery*)

S/W logiciel (*software*)

URI identificateur uniforme de ressource (*uniform resource identifier*)

URL localisateur uniforme de ressource (*uniform resource locator*)

XSPA attaque de port intersites (*cross-site port attack*)

XSS exécution de script intersites (*cross-site scripting*)

5 Conventions

Aucune.

6 Présentation générale

Un logiciel malveillant, utilisé pour nuire aux ressources d'information, est défini comme étant un logiciel conçu expressément pour endommager ou perturber un système, et nuire à la confidentialité, à l'intégrité et/ou à la disponibilité. Parmi les logiciels malveillants, figurent les virus informatiques, les vers, les chevaux de Troie, les logiciels espions, les logiciels publicitaires, la plupart des outils de dissimulation d'activité (rootkits), et d'autres programmes malveillants.

Les attaques sur le web sont des attaques perpétrées contre des sites web légitimes présentant des vulnérabilités, qui peuvent conduire à l'injection dans ces sites web de codes malveillants, lesquels peuvent ensuite infecter les ordinateurs des utilisateurs qui consultent ces sites web. Les codes malveillants peuvent prendre plusieurs formes: il peut s'agir d'une balise iframe cachée qui oriente l'utilisateur vers un site malveillant, ou d'applications malveillantes rédigées dans un langage informatique (par exemple un script ou des appliquettes). Comme exemples types de vulnérabilités utilisées pour lancer des attaques sur le web, on peut citer l'injection SQL et la falsification de requête intersites.

Le schéma d'attaque de la falsification de requête intersites [b-CAPEC-62] est un type d'attaque sur le web consistant à transmettre des commandes non autorisées ou à demander l'exécution d'actions indésirables sur un site web de confiance à l'insu de l'utilisateur lorsque celui-ci est connecté sur ce site. Le schéma d'attaque de l'injection de code en langage de requête structuré [b-CAPEC-66] est un autre type d'attaque sur un site web reposant sur une base de données, qui consiste pour l'attaquant à ajouter un code en langage de requête structuré (SQL) dans une zone de saisie de formulaire web pour accéder à des ressources ou modifier des données. Ce type d'attaque est utilisé pour dérober des informations figurant dans une base de données dont les données ne sont en principe pas mises à disposition et/ou pour accéder aux serveurs d'une organisation par l'intermédiaire de l'ordinateur qui héberge la base de données. Une balise iframe [b-iframe] est une balise utilisée pour insérer dans le document en langage de balisage hypertexte (HTML) actuel un document invisible, et ainsi tromper l'utilisateur en le faisant cliquer sur le document invisible par l'intermédiaire du clickjacking [b-CAPEC-103].

Ces derniers temps, les attaques sur le web ont nettement augmenté en raison d'une utilisation grandissante de dispositifs informatiques par les utilisateurs finals et de l'augmentation du nombre de sites web infectés par des logiciels malveillants.

Par exemple, des techniques anti-virus pourraient être appliquées au niveau des serveurs et des pare-feu pour applications web pourraient être mis en oeuvre au niveau des proxys afin d'assurer une mise en oeuvre rentable de ces techniques.

En ce qui concerne les attaques sur le web, il se peut que les administrateurs des sites web ne soient pas au courant du piratage des sites web, de leur infection par des codes malveillants et de leur utilisation pour diffuser des codes malveillants. De plus, il se peut que les utilisateurs ne soient pas non plus au courant de l'infection de leurs ordinateurs par des codes malveillants provenant des sites qu'ils ont visités. L'installation de logiciels antivirus permet d'éviter certains incidents, mais n'offre pas de solutions définitives.

Les raisons expliquant l'augmentation des attaques sur le web sont les suivantes:

- les téléchargements intempestifs depuis des sites web grand public sont en augmentation;
- certaines attaques font l'objet d'une forte obfuscation et de modifications dynamiques, rendant inefficaces les solutions classiques de détection et de prévention des logiciels malveillants;
- certaines attaques visent les extensions de navigateur web des utilisateurs finals;
- des attaques par injection SQL sont utilisées pour infecter des sites web grand public;

- des annonces publicitaires malveillantes redirigent les utilisateurs vers des sites web malveillants; et
- le nombre d'instances uniques et ciblées de logiciels malveillants augmente considérablement.

7 Techniques du système de protection contre les attaques sur le web

7.1 Techniques générales

Les techniques suivantes sont caractéristiques du système de protection contre les attaques sur le web:

- conception modulable, robuste et résistante;
- exploitation à travers de multiples domaines de sécurité, dont chacun est géré par un administrateur de sécurité responsable; et
- échange d'informations concernant les vulnérabilités des sites web ou les sites web infectés par des logiciels malveillants (sites web dans lesquels une balise i-frame invisible redirige les utilisateurs vers un site web infecté par des logiciels malveillants [b-CAPEC-103]);

NOTE – Pour l'échange d'informations, on pourrait utiliser le format existant d'échange de descriptions d'objet concernant les incidents (IODEF) [b-UIT-T X.1541].

- fonctionnement selon l'un ou l'autre des deux types de modèles de déploiement possibles: un modèle centralisé et un modèle réparti. Dans le modèle centralisé, toutes les informations sur les sites web infectés par des logiciels malveillants et sur les types de logiciels malveillants devraient être communiquées à un serveur centralisé, qui devrait les conserver et les gérer. Dans le modèle réparti, à chaque domaine de sécurité devrait être affecté un agent responsable et les informations sur les sites web infectés par des logiciels malveillants et sur les types de logiciels malveillants devraient être échangées entre les différents agents responsables répartis;
- configuration hiérarchisée pour faciliter l'évolutivité.

7.2 Techniques fonctionnelles

Les techniques fonctionnelles suivantes sont caractéristiques du système de protection contre les attaques sur le web:

- distinction entre les logiciels malveillants connus et les contenus web légitimes et protection contre l'installation de logiciels malveillants dans les sites web;
- détection de balises iframe invisibles qui redirigent l'utilisateur vers des sites web infectés par des logiciels malveillants;
- détection des vulnérabilités susceptibles d'être utilisées pour lancer des attaques types sur le web, telles que l'injection SQL, la falsification de référence intersites, etc., comme décrit dans l'Appendice IV;
- fonction d'analyse basée sur la signature ou fonction d'analyse équivalente permettant de détecter les logiciels malveillants connus dans les sites web;
- fonction d'analyse basée sur le comportement permettant d'identifier les logiciels malveillants inconnus;
- signalisation à l'administrateur d'un site web de toute infection de son site par des logiciels malveillants afin de les supprimer;

- détection des logiciels malveillants obfusqués au moyen du fractionnement de chaîne, du codage de chaîne, du codage de chaîne personnalisé, de la modification de comportement de script, de l'obfuscation de fonctions de modification de modèle d'objet de document (DOM), de la dissimulation de liens derrière des services publics, et de redirections vers d'autres sites web;
- détection des logiciels malveillants susceptibles d'être utilisés pour lancer des attaques par falsification de référence intersites dans les sites web;
- évaluation des comportements de logiciels malveillants suspects dans les sites web;
- signalisation aux utilisateurs qui visitent des sites web de toute infection de ces sites;
- signalisation à l'administrateur de sécurité lorsqu'un logiciel malveillant est détecté dans un site web, du risque d'utilisation de ce site web pour lancer des attaques sur le web;
- échange d'informations sur les listes noires des sites web malveillants; et
- identification des vulnérabilités présentes dans les sites web (injection SQL, exécution de script intersites, etc.) et signalisation à l'administrateur de ces sites web des vulnérabilités identifiées dans ces sites.

7.3 Techniques de gestion

Les techniques de gestion suivantes sont caractéristiques du système de protection contre les attaques sur le web:

- prise en charge de la gestion de la sécurité sur la base de politiques de sécurité lorsque le système est déployé dans différents domaines de sécurité;
- interface de gestion unifiée avec un système de gestion centralisé;
- prise en charge de la gestion de la confiance et acceptation des données d'événements concernant des attaques uniquement lorsque ces données proviennent de domaines de sécurité fiables;
- prise en charge de la gestion des ressources et protection contre la surcharge; et
- prise en charge de la gestion de l'exploitation et de la maintenance, y compris de la gestion de la configuration du système, de la gestion de la journalisation, de la surveillance de l'état du système, etc.

7.4 Techniques de sécurité et de confidentialité

Les techniques de sécurité et de confidentialité suivantes sont caractéristiques du système de protection contre les attaques sur le web:

- prise en charge de la confidentialité, de l'authentification de l'origine des données et de l'intégrité des informations échangées via l'interface de communication entre différents domaines de sécurité;
- protection contre la fuite des informations d'identification personnelle (PII) traitées par le système de prévention des attaques sur le web;
- résistance face aux diverses attaques sur le réseau, par exemple les attaques DDoS; et
- fonctionnalité d'audit permettant de repérer les utilisations abusives par des entités non autorisées des informations que le système collecte.

8 Fonctions du système de protection contre les attaques sur le web

Le système de protection contre les attaques sur le web devrait comporter au moins les fonctions suivantes (liste non exhaustive):

- la détection de toutes les vulnérabilités connues dans les sites web;
- la détection des sites web contenant des logiciels malveillants qui sont utilisés pour la distribution de tels logiciels;
- la notification à l'administrateur des sites web qui contiennent des logiciels malveillants et présentent des vulnérabilités connues susceptibles d'être exploitées par des attaquants;
- la collecte des informations nécessaires concernant les vulnérabilités des sites web et les logiciels malveillants qu'ils contiennent;
- le partage d'informations concernant les sites web infectés par des logiciels malveillants et ceux qui sont utilisés pour la distribution de logiciels malveillants entre des entités fiables dans un domaine de sécurité et entre plusieurs domaines;
- la mise en oeuvre de la politique de sécurité du système proprement dit dans un domaine; et
- la protection du système proprement dit contre toutes les attaques.

9 Format d'échange d'informations

L'échange d'informations d'analyse des logiciels malveillants (par exemple énumération et caractérisation des attributs des logiciels malveillants) devrait être renforcé. Les responsables chargés de la mise en oeuvre de la présente Recommandation peuvent utiliser [b-UIT-T X.1546] pour échanger des informations d'analyse des logiciels malveillants.

Appendice I

Scénarios utilisés pour les attaques sur le web

(Cet appendice ne fait pas partie intégrante de la présente Recommandation.)

I.1 Scénario relatif à l'infection par des logiciels malveillants

La Figure I-1 décrit un scénario type d'attaques sur le web.

1. L'attaquant compromet un site web légitime qui présente des vulnérabilités puis installe un logiciel malveillant ou un script qui est utilisé pour attaquer l'ordinateur de l'utilisateur ou installe des balises afin de rediriger l'ordinateur de l'utilisateur visitant ce site vers un site web contenant un logiciel malveillant et de l'attaquer.
2. Lorsqu'un utilisateur, une victime, visite le site web qui a été compromis par l'attaquant, son ordinateur est attaqué par le logiciel malveillant intégré ou est redirigé vers un autre site web contenant un logiciel malveillant destiné à l'attaquer.
3. La présence, dans le navigateur de l'ordinateur de l'utilisateur, de vulnérabilités susceptibles d'être exploitées par le logiciel malveillant en question peut conduire à l'installation dudit logiciel dans l'ordinateur, qui en est alors infecté à l'insu ou sans la permission de l'utilisateur.
4. Le logiciel malveillant installé dans l'ordinateur de l'utilisateur pourrait être utilisé pour lancer des attaques massives de type déni de service réparti (DDoS) ou pour dérober des informations personnelles telles que l'identité (ID) et le mot de passe et les transmettre à l'attaquant.

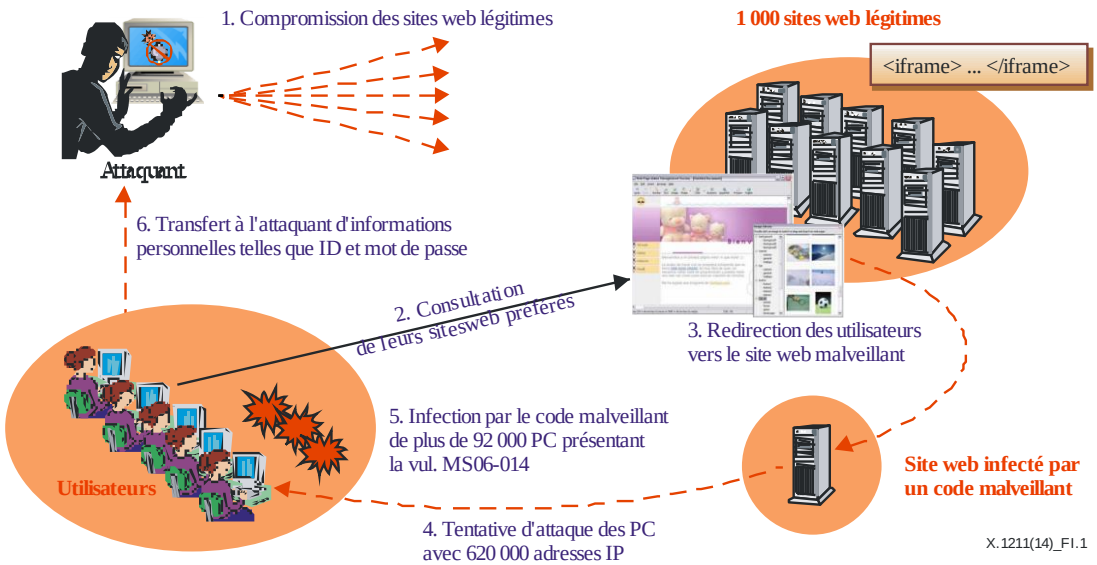

Figure I-1 – Scénario type d'attaques sur le web

I.2 Falsification de requête intersites (CAPEC-62)

La falsification de requête intersites (CSRF) peut amener un utilisateur à soumettre à son insu une ou plusieurs requêtes HTTP à un site web vulnérable auquel il fait confiance. Une attaque type par falsification de requête intersites peut alors compromettre l'intégrité de données, et donner à l'attaquant la possibilité de modifier les informations stockées sur un site web vulnérable.

Lorsqu'un site web exige l'authentification de l'utilisateur, souvent l'utilisateur n'a pas besoin de saisir son mot de passe pour chaque requête HTTP. En revanche, le site web identifie l'état d'authentification de l'utilisateur entre plusieurs requêtes HTTP au moyen de jetons tels que des cookies de session ou l'en-tête d'autorisation HTTP. Toutefois, un problème se pose: les navigateurs web mémorisent le jeton associé à un URL et rattachent automatiquement le jeton lorsqu'une nouvelle demande HTTP est adressée au site web, même si elle n'est pas à l'initiative de l'utilisateur. La falsification CSRF tire parti du comportement du navigateur. Elle nécessite simplement qu'un utilisateur visite un site web malveillant dans lequel figure une logique JavaScript qui adresse des demandes HTTP (potentiellement cachées) à un autre site web (par exemple la banque de l'utilisateur), demandes HTTP qui pourraient être autorisées par le site web en raison de la présence des jetons. La falsification CSRF permet de lancer divers types d'attaques, par exemple l'envoi de courriels depuis un service de messagerie sur le web, le postage d'un commentaire sur un blog pour le compte de l'utilisateur, la modification de la liste de contacts de l'utilisateur dans un service de réseau social (SNS), ou la modification des paramètres dans un routeur domestique.

I.3 Attaques de port intersites/falsification de requête côté serveur

Les attaques de port intersites/la falsification de requête côté serveur (XSPA/SSRF) constituent une méthode d'utilisation abusive des applications web qui traitent les URL utilisés dans un navigateur web. Une attaque XSPA/SSRF vise généralement l'intranet de l'application vulnérable. Elle peut entraîner un scannage de ports, une compromission de la confidentialité des données, l'exécution de code non autorisée et l'exploitation de ressources vulnérables de l'intranet.

Une application est considérée comme vulnérable aux attaques XSPA/SSRF lorsqu'elle ne valide pas les résultats reçus d'un serveur distant et les données saisies par l'utilisateur final. A titre d'exemple, une application qui télécharge une image depuis un URL fourni par un utilisateur pourrait accéder à une ressource de l'intranet lorsque l'utilisateur poste l'URL, par exemple 'http://localhost/secret.txt'. Dans certains cas, des structures spéciales d'URI peuvent être utilisées de manière à ce qu'une application vulnérable envoie une demande à des services spéciaux tels que 'https','gopher','ftp','ldap'. Des structures propres à un langage telles que 'php://fd','php://memory' pourraient également être utilisées.

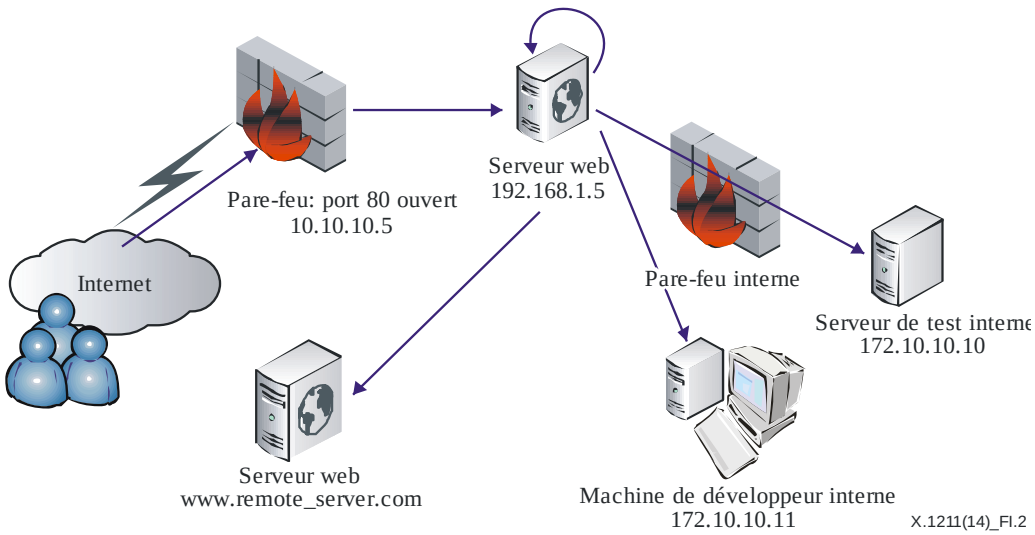

Figure I-2 – Scénario type des attaques de port intersites/de la falsification de requête côté serveur

I.4 Injection SQL (CAPEC-66)

Un scénario type d'injection SQL repose sur une mauvaise vérification des données d'entrée des applications web. Les canaux d'entrée sont divers: demandes HTTP GET et POST, cookies de navigateurs, contenu de données XML, entrées de fichiers, ou autres.

L'entrée visée est alors injectée dans une requête SQL. Un exemple type d'injection SQL dans le paramètre HTTP "GET" est le suivant:

- Dans la requête originale suivante – "SELECT title, content FROM table1 WHERE id=%d", "id" est le paramètre visé.

Dans des conditions normales, "id" est un entier naturel, mais en raison d'une absence de vérification, l'attaquant pourrait fournir l'entrée suivante à la place d'un nombre:

- %d = "1 UNION SELECT user, password FROM secret_table".

Il en résulterait un accès non autorisé à "secret_table", ce qui pourrait entraîner une divulgation de données sensibles directement dans la sortie du serveur.

En fonction de la base de données SQL mise en place, une telle attaque peut avoir pour conséquence:

- la divulgation de données sensibles de la base de données ou du système de fichiers;
- la perte ou la modification de données;
- l'injection de portes dérobées ou l'élévation de privilèges; et
- la propagation de logiciels malveillants auprès des utilisateurs finals se rendant sur le site.

I.5 Détection des logiciels malveillants dans les sites web

Les techniques utilisées pour détecter les logiciels malveillants peuvent être classées en deux catégories: la détection basée sur les anomalies et la détection basée sur la signature [b-NA].

Dans le cas de la détection basée sur les anomalies, les critères utilisés pour déterminer si un programme considéré est malveillant sont liés à ce qui constitue un comportement normal. La détection basée sur la spécification est un type particulier de détection basée sur les anomalies. Les techniques de détection basée sur la spécification utilisent une spécification ou un ensemble de règles de comportement valable afin de déterminer si un programme considéré est malveillant. Les programmes ne respectant pas l'ensemble de règles ou la spécification sont considérés comme malveillants.

Dans le cas de la détection basée sur la signature, les critères utilisés pour déterminer si un programme considéré est malveillant sont liés à la caractérisation de ce qui est connu comme étant malveillant. L'efficacité d'une méthode de détection basée sur la signature dépend de la caractérisation ou de la signature d'un comportement malveillant.

Chacune des techniques de détection peut employer une approche parmi les trois suivantes: statique, dynamique ou hybride, qui correspondent à des façons différentes de collecter les informations pour détecter les logiciels malveillants. L'analyse statique utilise la syntaxe ou les propriétés structurelles du programme (statique)/processus (dynamique) considéré (PUI) pour déterminer si celui-ci est malveillant. Par exemple, dans le cas de la détection basée sur la signature, une approche statique consisterait à utiliser uniquement les informations structurelles (par exemple la séquence d'octets), tandis qu'une approche dynamique utilisera les informations relatives à l'exécution du PUI (par exemple les systèmes visibles sur la pile d'exécution).

D'une manière générale, une approche statique tente de détecter les logiciels malveillants avant l'exécution du programme considéré, alors qu'une approche dynamique tente de détecter un comportement malveillant pendant ou après l'exécution du programme.

Il existe des techniques hybrides qui combinent les deux approches. Dans ce cas, on utilise à la fois des informations statiques et des informations dynamiques pour détecter les logiciels malveillants.

On recense plusieurs techniques de détection des logiciels malveillants dans les sites web, qui sont décrites dans l'Appendice III.

Appendice II

Méthode utilisée pour infecter les ordinateurs des utilisateurs avec des logiciels malveillants

(Cet appendice ne fait pas partie intégrante de la présente Recommandation.)

Le présent Appendice a pour objet de décrire les scénarios types susceptibles d'être utilisés par des attaquants afin d'en faciliter la compréhension par les administrateurs.

La première étape d'une attaque sur web consiste à installer et exécuter divers codes malveillants sur l'ordinateur d'un utilisateur, par exemple des enregistreurs de frappe ou des rootkits (qui peuvent transformer l'ordinateur de l'utilisateur en ordinateur zombie ou entraîner une fuite d'informations sensibles relatives à l'utilisateur vers les attaquants).

L'objectif de l'attaque pourrait être atteint soit par l'exploration de plusieurs vulnérabilités connues dans différents éléments logiciels accessibles depuis un navigateur (par exemple des éléments de système d'exploitation accessibles depuis un navigateur par le biais d'ActiveX, etc.), soit au moyen de techniques d'ingénierie sociale consistant à leurrer les utilisateurs et à leur faire installer et exécuter des logiciels malveillants dans leur système, ainsi qu'à tenter de dérober les justificatifs de l'utilisateur au moyen de techniques d'hameçonnage ou d'attaques par exécution de script intersites lancées dans un iframe caché.

Un certain nombre de techniques sont utilisées pour infecter l'ordinateur d'un utilisateur avec des logiciels malveillants: exploitation de l'élément ActiveX, techniques d'ingénierie sociale, codec manquant, techniques basées sur des outils de suppression des logiciels malveillants, et attaques par falsification de requête intersites. Pour plus de précisions, on pourra se reporter à la référence [b-NTobjectives]. De plus, on trouvera dans la référence [b-UIT-T X.1544] une liste de schémas d'attaque courants et leur classification.

Appendice III

Exemples types de techniques d'obfuscation

(Cet appendice ne fait pas partie intégrante de la présente Recommandation.)

Les contenus malveillants injectés utilisent des techniques d'obfuscation afin de dissimuler les logiciels malveillants à la fois pour l'oeil humain et pour les logiciels de détection de vulnérabilité [b-UIT-T X.1520]. Les techniques d'obfuscation sont relativement efficaces pour les raisons suivantes:

- Bon nombre d'administrateurs de sites web sont méfiants à l'idée de supprimer des codes de script qu'ils ne comprennent pas.
- Les administrateurs de base de données ont du mal à nettoyer les bases de données infectées, car ils ne savent pas quelles structures ils doivent rechercher.
- De nombreuses méthodes de détection sont fondées sur la recherche d'expressions ou d'autres chaînes ordinaires, et ont donc du mal à identifier le code HTML obfusqué.

Il existe plusieurs méthodes d'obfuscation: fractionnement de chaîne, codage de chaîne, codage de chaîne personnalisé, modification de comportement de script, obfuscation de fonctions de modification de modèle d'objet de document (DOM), dissimulation de liens derrière des services publics, et redirection vers d'autres pages. Pour plus de précisions, on pourra se reporter à la référence [b-NTobjectives].

Appendice IV

Techniques de prévention des attaques sur le web

(Cet appendice ne fait pas partie intégrante de la présente Recommandation.)

Le présent Appendice a pour objet de présenter plusieurs techniques de détection des logiciels malveillants dans les sites web [b-NTobjectives]. Pour détecter les contenus malveillants, on peut recourir à la concordance de signature de contenu, à une liste noire des sites malveillants, ou à une analyse du comportement de contenu suspect au moyen d'algorithmes propriétaires.

IV.1 Suppression des vulnérabilités dans les sites web

La solution la plus simple consiste à supprimer les vulnérabilités dans les sites web, y compris l'injection SQL et l'exécution de script intersites. Si l'attaquant ne parvient pas à insérer des contenus malveillants dans un site web, un navigateur client n'exécutera pas les logiciels malveillants insérés dans ledit site. Par conséquent, la solution la plus efficace pour prévenir les attaques sur le web consiste à supprimer toutes les vulnérabilités dans les sites web.

IV.2 Concordance de signature

Compte tenu du nombre de techniques d'obfuscation et d'outils d'automatisation qui existent pour obfusquer les logiciels malveillants, il est impossible de détecter les contenus malveillants dans les sites web en utilisant une méthode de détection basée sur la signature. On sait très bien que les attaquants peuvent automatiser le codage de contenus malveillants avec une nouvelle clé pour chaque site web, de sorte que la signature des logiciels malveillants créée est différente pour chaque site web.

Toutefois, le contenu des logiciels malveillants en clair n'est pas modifié fréquemment, de sorte que les logiciels malveillants présents dans le site web peuvent être détectés au moyen d'une signature. Si le contenu des logiciels malveillants en clair est obtenu par décodage des logiciels malveillants codés et que la signature des logiciels malveillants en clair est calculée à partir de ceux-ci, on peut alors détecter les logiciels malveillants en comparant la signature calculée des logiciels malveillants avec une liste regroupant toutes les signatures du contenu des logiciels malveillants connues à l'avance.

IV.3 Liste noire de sites

L'inscription sur liste noire des sites web malveillants figure parmi les techniques de détection les plus précieuses. Même si des contenus malveillants peuvent être complètement hébergés sur un site web légitime (et ne nécessitent pas de télécharger automatiquement des scripts ou iframes présents sur un site malveillant, d'où une dissimulation de leur connexion avec le site malveillant), il est nécessaire d'échanger certaines données avec le site web malveillant pour mener à bien l'attaque prévue. Cet échange de données nécessaire peut prendre de nombreuses formes différentes: le script malveillant doit télécharger des logiciels malveillants présents sur un site web malveillant, ou doit envoyer au site malveillant des données privées collectées dans le système de l'utilisateur, etc. En tout état de cause, le script malveillant doit établir une connexion avec un site malveillant.

Si un algorithme permet de détecter que des ressources externes font partie des sites inscrits sur liste noire, on peut suspecter que le site web contient des logiciels malveillants. Par conséquent, toute détection de site inscrit sur liste noire indiquera la présence de contenus malveillants sur une page faisant l'objet d'une analyse.

IV.4 Détection des techniques d'obfuscation

La présence dans un site web de contenus codés au moyen de techniques d'obfuscation pourrait constituer un bon indice du caractère malveillant du site web. Par exemple, si un site web contient une longue chaîne codée, il pourrait s'agit de contenus malveillants. Toutefois, même si la longue chaîne codée est suspecte, on ne peut affirmer que le site web comporte des contenus malveillants tant que la chaîne n'a pas été décodée et que son rôle n'a pas été analysé.

IV.5 Evaluation du comportement des contenus suspects

La solution la plus efficace consiste à analyser le comportement des contenus suspects. Si l'activité de certains contenus est suspecte, ce peut être un indice qu'il s'agit de contenus malveillants. Parmi les comportements types qui pourraient être considérés comme malveillants, on peut citer l'accès au disque dur local, l'instanciation d'un objet d'application d'interprétation et le téléchargement de contenus exécutables externes (ou l'accès à ces contenus).

Appendice V

Exemples types de risques pour la sécurité des applications, recensés par l'OWASP

(Cet appendice ne fait pas partie intégrante de la présente Recommandation.)

Le projet ouvert concernant la sécurité des applications web (OWASP), communauté ouverte à tous, s'occupant des outils, techniques et méthodes de sécurité sur le web et regroupant des entreprises de premier plan, des structures d'enseignement et des particuliers du monde entier, a publié la liste des dix principaux risques pour la sécurité des applications [b-OWASP] et CWE [b-UIT-T X.1524] CWE-928: Weaknesses in OWASP Top Ten [b-CWE], recensés dans le Tableau V.1.

Tableau V.1 – Les dix principaux risques pour la sécurité des applications, recensés par l'OWASP

Risque	Agent de menace	Vecteur d'attaque	Vulnérabilité	Impact sur le plan technique	Impact au niveau de l'entreprise	Références
A-1 – Injection	Toute personne qui peut envoyer des données non fiables au système, y compris les utilisateurs externes, les utilisateurs internes et les administrateurs.	L'attaquant envoie de simples attaques basées sur du texte qui exploitent la syntaxe d'un interpréteur cible. Presque toutes les sources de données peuvent être des vecteurs d'injection, y compris les sources internes.	Les failles d'injection surviennent quand une application envoie des données non fiables à un interpréteur. Les failles d'injection sont très fréquentes, surtout dans un code ancien. On les retrouve souvent dans les requêtes SQL, LDAP ou XPath, commandes OS, arguments de programme, etc. Les failles d'injection se détectent facilement via l'examen du code, difficilement via le test. Scanners et fuzzers peuvent aider les attaquants à trouver les failles d'injection.	L'injection peut résulter en une perte ou une corruption de données, une perte de droits, ou un refus d'accès. L'injection peut parfois mener à une prise de contrôle totale du serveur.	Prendre en compte la valeur pour l'entreprise des données affectées et la plate-forme exécutant l'interpréteur. Toutes les données pourraient être dérobées, modifiées ou supprimées. La réputation de l'entreprise pourrait-elle en pâtir?	CWE-77; CWE-78, CWE-89, CWE-90, CWE-91, CWE-929

Tableau V.1 – Les dix principaux risques pour la sécurité des applications, recensés par l'OWASP

Risque	Agent de menace	Vecteur d'attaque	Vulnérabilité	Impact sur le plan technique	Impact au niveau de l'entreprise	Références
A-2 –Violation d'authentification et de gestion de session	Prendre en compte les attaquants externes anonymes ainsi que les utilisateurs ayant leur propre compte qui peuvent tenter de dérober des comptes d'autres utilisateurs. Prendre aussi en compte les utilisateurs internes voulant camoufler leurs actes.	L'attaquant exploite des fuites/failles dans les fonctions d'authentification ou de gestion de session (par exemple comptes, mots de passe, ID de session exposés) pour usurper l'identité des utilisateurs.	Développer correctement un système personnalisé d'authentification ou de gestion de session est difficile. En conséquence, ce type de système a souvent des failles dans des domaines tels que la déconnexion, la gestion de mots de passe, l'expiration de temporisation, la fonction "se souvenir de moi", la question secrète, la mise à jour de compte, etc. Trouver de telles failles s'avère parfois difficile, chaque implémentation étant unique.	Avec de telles failles, des attaques contre une partie voire la totalité des comptes peuvent être menées à bien. L'attaquant peut ensuite faire tout ce que la victime pouvait faire. Les comptes à privilèges sont souvent ciblés.	Prendre en compte la valeur pour l'entreprise des données ou des fonctions applicatives touchées. Prendre aussi en compte l'impact pour l'entreprise de la divulgation de la vulnérabilité au grand public.	CWE-256, CWE-287, CWE 311, CWE-319, CWE-384 CWE-522, CWE-523, CWE-613, CWE-620, CWE-640, CWE-930

Tableau V.1 – Les dix principaux risques pour la sécurité des applications, recensés par l'OWASP

Risque	Agent de menace	Vecteur d'attaque	Vulnérabilité	Impact sur le plan technique	Impact au niveau de l'entreprise	Références
A-3 –Exécution de script intersites (XSS)	Prendre en compte toute personne qui peut envoyer des données non fiables au système, y compris les utilisateurs externes, les utilisateurs internes et les administrateurs.	L'attaquant envoie des scripts basés sur du texte qui exploitent l'interpréteur dans le navigateur. Presque toutes les sources de données peuvent être des vecteurs d'attaque, y compris les sources internes telles qu'une base de données.	XSS est la faille la plus répandue dans les applications web. Les failles XSS ont lieu lorsqu'une application inclut des données fournies par l'utilisateur dans une page envoyée au navigateur, sans validation ou échappement correct de ce contenu. Il en existe trois types connus: 1) XSS stockée; 2) XSS réfléchie; et 3) XSS basée sur DOM. La détection de la plupart des failles XSS est assez simple par test ou analyse de code.	L'attaquant peut exécuter des scripts dans le navigateur de la victime pour détourner des sessions, défigurer des sites, insérer du contenu hostile, rediriger l'utilisateur, détourner le navigateur de l'utilisateur au moyen d'un logiciel malveillant, etc.	Prendre en compte la valeur pour l'entreprise du système affecté ainsi que de l'ensemble des données qu'il traite. Prendre aussi en compte l'impact pour l'entreprise de la divulgation de la vulnérabilité au grand public.	CWE-79 CWE-931

Tableau V.1 – Les dix principaux risques pour la sécurité des applications, recensés par l'OWASP

Risque	Agent de menace	Vecteur d'attaque	Vulnérabilité	Impact sur le plan technique	Impact au niveau de l'entreprise	Références
A-4 –Références directes non sécurisées à des objets	Prendre en compte les différents types d'utilisateurs de votre système. Certains d'entre eux ont-ils un accès partiel à certains types de données du système?	L'attaquant, qui est un utilisateur autorisé du système, remplace simplement la valeur d'un paramètre faisant directement référence à un objet du système, par une référence à un autre objet qui lui est interdit. Aura-t-il accès à cette ressource?	Les applications utilisent souvent le nom ou la clé même des objets lors de la génération de pages web. La vérification des autorisations de l'utilisateur avant accès aux objets n'est pas systématique. On parle dans ce cas de références directes non sécurisées. Il est facile de détecter cette vulnérabilité en modifiant la valeur des paramètres lors de tests. L'analyse du code permet aussi de démontrer rapidement si les autorisations sont bien vérifiées.	Toutes les données référencées par le paramètre considéré peuvent être compromises. A moins que des références non prédictibles soient utilisées, il est facile pour l'attaquant d'accéder à toutes les données disponibles pour le type en question.	Prendre en compte la valeur pour l'entreprise des données exposées. Prendre aussi en compte l'impact pour l'entreprise de la divulgation de la vulnérabilité au grand public.	CWE-22, CWE-99, CWE-639, CWE-932

Tableau V.1 – Les dix principaux risques pour la sécurité des applications, recensés par l'OWASP

Risque	Agent de menace	Vecteur d'attaque	Vulnérabilité	Impact sur le plan technique	Impact au niveau de l'entreprise	Références
A-5 –Mauvaise configuration de la sécurité	Prendre en compte les attaquants externes anonymes ainsi que les utilisateurs ayant leur propre compte qui peuvent tenter de compromettre le système. Prendre aussi en compte les utilisateurs internes voulant camoufler leurs actes.	L'attaquant accède à des comptes par défaut, des pages non utilisées, des vulnérabilités non corrigées, des fichiers et répertoires non protégés, etc., afin d'obtenir un accès au système ou des informations sur le système sans y être autorisé.	Une mauvaise configuration de la sécurité peut survenir dans n'importe laquelle des couches, la plate-forme, le serveur web, le serveur d'application, le cadre et le code spécifique. Développeurs et administrateurs système ont besoin de travailler ensemble pour s'assurer que l'ensemble des couches sont configurées correctement. Les scanners automatisés sont utiles pour détecter les correctifs manquants, les erreurs de configuration, l'utilisation de comptes par défaut, les services inutiles, etc.	Ces vulnérabilités donnent souvent aux attaquants un accès non autorisé à des systèmes ou des fonctionnalités. Occasionnellement, elles entraînent une compromission complète du système.	Le système peut être complètement compromis à votre insu. Toutes vos données peuvent être dérobées ou modifiées lentement dans le temps. Les coûts de récupération peuvent être élevés.	CWE-2, CWE-16, CWE-209, CWE-215, CWE-548, CWE-933

Tableau V.1 – Les dix principaux risques pour la sécurité des applications, recensés par l'OWASP

Risque	Agent de menace	Vecteur d'attaque	Vulnérabilité	Impact sur le plan technique	Impact au niveau de l'entreprise	Références
A-6 –Exposition de données sensibles	Prendre en compte tout utilisateur interne ou externe pouvant obtenir un accès à vos données sensibles ou à toute copie de sauvegarde de ces données, qu'il s'agisse de données en mémoire, en transit, voire dans les navigateurs de vos clients.	La cryptanalyse (cassage de l'algorithme ou de la clé) reste rare. On préfère dérober les clés, intercepter le trafic ou accéder aux données en clair sur le serveur, en transit ou dans le navigateur des utilisateurs.	La principale erreur est de ne pas chiffrer les données sensibles. Les autres erreurs fréquentes sont: génération et gestion de clés faibles, utilisation d'algorithmes faibles, en particulier protection insuffisante des mots de passe. Les faiblesses dans le navigateur sont répandues et simples à détecter, mais difficiles à exploiter à grande échelle. En général, les faiblesses côté serveur sont plus difficiles à identifier et à exploiter de l'extérieur, en raison d'un accès limité.	L'exploitation entraîne souvent la compromission de toutes les données qui auraient dû être protégées, à savoir des données sensibles telles que données personnelles, médicales, éléments de cartes de crédit ou d'authentification, etc.	Prendre en compte la valeur pour l'entreprise des données perdues, l'impact pour la réputation de l'entreprise ainsi que les implications légales pouvant résulter de l'exposition de ces données.	CWE-310, CWE-311, CWE-312, CWE-319, CWE-325, CWE-326, CWE-934

Tableau V.1 – Les dix principaux risques pour la sécurité des applications, recensés par l'OWASP

Risque	Agent de menace	Vecteur d'attaque	Vulnérabilité	Impact sur le plan technique	Impact au niveau de l'entreprise	Références
A-7 – Contrôle d'accès au niveau fonctionnel	Toute personne ayant accès au réseau peut envoyer une requête à votre application. Un utilisateur anonyme peut-il accéder à une fonctionnalité privée? Un simple utilisateur peut-il accéder à une fonctionnalité privilégiée?	L'attaquant, qui est un utilisateur autorisé du système, modifie simplement l'URL ou les paramètres d'une fonctionnalité privilégiée. Si l'accès est autorisé, cela signifie que des utilisateurs anonymes pourraient accéder à des fonctionnalités privées non protégées.	Les applications ne protègent pas toujours correctement certaines fonctionnalités. Parfois, les protections au niveau fonctionnel sont gérées par configuration et le système est mal configuré. Parfois, les développeurs oublient d'intégrer les vérifications logicielles adéquates. Détecter de telles vulnérabilités est aisé. La tâche la plus difficile consiste à identifier les pages (URL) ou fonctionnalités devant être testées.	Ces vulnérabilités permettent à un attaquant d'accéder à des fonctionnalités non autorisées. Les fonctionnalités d'administration sont les cibles privilégiées de ce type d'attaque.	Prendre en compte la valeur pour l'entreprise des fonctionnalités exposées et des données qu'elles traitent. Prendre aussi en compte l'impact sur la réputation de l'entreprise si la vulnérabilité devenait publique.	CWE-285, CWE-287, CWE-935

Tableau V.1 – Les dix principaux risques pour la sécurité des applications, recensés par l'OWASP

Risque	Agent de menace	Vecteur d'attaque	Vulnérabilité	Impact sur le plan technique	Impact au niveau de l'entreprise	Références
A-8 – Falsification de requête intersites (CSRF)	Prendre en compte toute personne qui peut charger du contenu dans le navigateur de vos utilisateurs et, ainsi, les forcer à soumettre une requête sur votre site web. N'importe quel site web ou flux HTML auquel vos utilisateurs ont accès peut être utilisé à cette fin.	L'attaquant falsifie une requête HTTP et, par le biais d'une balise image, d'une faille XSS ou d'une autre technique, force le navigateur à émettre la requête, à l'insu de l'utilisateur. Si ce dernier est authentifié, l'attaque va réussir.	CSRF exploite le fait que la plupart des applications web permettent aux attaquants de prévoir tous les détails de certaines actions. Et, comme les navigateurs envoient automatiquement les informations d'authentification, telles que les cookies de session, les attaquants peuvent concevoir des pages web malveillantes, qui génèrent des requêtes falsifiées paraissant légitimes. Une faille CSRF est assez facile à détecter par une analyse de code, ou par un test d'intrusion.	L'attaquant peut persuader la victime par la ruse de réaliser n'importe quelle opération de changement d'état que la victime est autorisée à réaliser. Ainsi, il peut la forcer à modifier son compte, à faire des achats, à se déconnecter ou même à se connecter.	Prendre en compte la valeur pour l'entreprise des données ou des fonctions affectées. Imaginer l'impact qu'il y aurait à ne pas savoir si les actions réalisées par vos utilisateurs ont été faites intentionnellement ou pas. Prendre en compte l'impact sur la réputation de l'entreprise.	CWE-346, CWE-352, CWE-441, CWE-642, CWE-935

Tableau V.1 – Les dix principaux risques pour la sécurité des applications, recensés par l'OWASP

Risque	Agent de menace	Vecteur d'attaque	Vulnérabilité	Impact sur le plan technique	Impact au niveau de l'entreprise	Références
A-9 – Utilisation de composants avec des vulnérabilités connues	Certains composants vulnérables (par exemple des bibliothèques) peuvent être identifiés et exploités à l'aide d'outils automatisés, augmentant la population des agents de menace au-delà des attaquants cibles pour inclure des hacktivistes.	L'attaquant utilise des outils manuels ou des scanners afin d'identifier un composant vulnérable. Il personnalise si besoin l'exploit et exécute l'attaque. Cela est d'autant plus difficile si le composant est imbriqué profondément dans l'application.	Potentiellement, toutes les applications peuvent être impactées par ces vulnérabilités, notamment parce que souvent les équipes de développement ne s'assurent pas que les composants/ bibliothèques sont à jour. Dans la plupart des cas, les développeurs ne connaissent même pas tous les composants sur lesquels repose leur application, sans même parler des numéros de version correspondants. Les dépendances entre composants aggravent d'autant plus la situation.	Toutes les faiblesses sont envisageables: injection, violation de contrôle d'accès, XSS, etc. L'impact peut aller d'une incidence minime à la prise de contrôle totale du système et à la compromission de toutes les données.	Prendre en compte ce qu'implique la vulnérabilité pour l'entreprise utilisant l'application touchée. Cela peut être bénin ou correspondre à une compromission totale.	CWE-937

Tableau V.1 – Les dix principaux risques pour la sécurité des applications, recensés par l'OWASP

Risque	Agent de menace	Vecteur d'attaque	Vulnérabilité	Impact sur le plan technique	Impact au niveau de l'entreprise	Références
A-10 –Redirections et renvois non validés	Prendre en compte toute personne pouvant persuader vos utilisateurs par la ruse de soumettre une requête sur votre site web. N'importe quel site web ou flux HTML auquel vos utilisateurs ont accès peut être utilisé à cette fin.	Un attaquant crée un lien vers une redirection non validée et persuade les victimes par la ruse de cliquer sur ce lien. Les victimes sont en confiance car le lien pointe vers un site connu. L'attaquant cible les renvois non sûrs afin de contourner les mécanismes de sécurité.	Les applications utilisent fréquemment les redirections et les renvois pour rediriger les utilisateurs vers d'autres pages. Parfois la page cible est spécifiée dans un paramètre non validé, permettant à un attaquant de choisir la page de destination. La détection de redirections non vérifiées est facile. Il suffit de rechercher les redirections où l'URL complet peut être modifié. La détection de renvois non vérifiés est plus compliquée puisqu'ils ciblent des pages internes.	Certaines redirections visent à installer des logiciels malveillants ou à persuader les victimes par la ruse de divulguer des mots de passe ou d'autres informations sensibles. Des renvois non sûrs peuvent amener le contournement de contrôles d'accès.	Prendre en compte la valeur pour l'entreprise du maintien de la confiance de vos utilisateurs. Quel serait l'impact en cas d'installation d'un logiciel malveillant? Quel serait l'impact si un attaquant accédait à des fonctions uniquement internes?	CWE-601, CWE-938

Bibliographie

[b-UIT-T M.3030] Recommandation UIT-T M.3030 (2002), *Langage de balisage pour les télécommunications (tML): cadre général*.

[b-UIT-T T.411] Recommandation UIT-T T.411 (1993)|ISO/CEI 8613-1:1994, *Technologies de l'information – Architecture ouverte des documents et format d'échange: introduction et principes généraux*.

[b-UIT-T X.800] Recommandation UIT-T X.800 (1991), *Architecture de sécurité pour l'interconnexion en systèmes ouverts d'applications du CCITT*.

[b-UIT-T X.810] Recommandation UIT-T X.810 (1995)|ISO/CEI 10181-1:1996, *Technologies de l'information – Interconnexion des systèmes ouverts – Cadres de sécurité pour les systèmes ouverts: aperçu général*.

[b-UIT-T X.1252] Recommandation UIT-T X.1252 (2010), *Termes et définitions de base relatifs à la gestion d'identité*.

[b-UIT-T X.1520] Recommandation UIT-T X.1520 (2014), *Vulnérabilités et expositions courantes.*

[b-UIT-T X.1524] Recommandation UIT-T X.1524 (2012), *Liste des failles courantes.*

[b-UIT-T X.1541] Recommandation UIT-T X.1541 (2012), *Format d'échange de descriptions d'objet concernant les incidents*.

[b-UIT-T X.1544] Recommandation UIT-T X.1544 (2013), *Liste et classification des schémas d'attaque courants*.

[b-UIT-T X.1546] Recommandation UIT-T X.1546 (2014), *Enumération et caractérisation des attributs de logiciels malveillants*.

[b-ISO/CEI 27000] ISO/CEI 27000:2012, ISO/CEI 27000:2012, *Technologies de l'information – Techniques de sécurité – Systèmes de management de la sécurité de l'information – Vue d'ensemble et vocabulaire*.

[b-ISO/CEI 27033-1] ISO/CEI 27033-1:2009, *Technologies de l'information – Sécurité de réseau – Partie 1: Vue d'ensemble et concepts*.

[b-CAPEC-62] CAPEC-62: Cross Site Request Forgery (ou "Session Riding").
https://capec.mitre.org/data/definitions/62.html

[b-CAPEC-66] CAPEC-66: SQL Injection.
https://capec.mitre.org/data/definitions/66.html

[b-CAPEC-103] CAPEC-103: Clickjacking.
https://capec.mitre.org/data/definitions/103.html

[b-CWE] CWE-928: Weaknesses in OWASP Top Ten (2013).
http://cwe.mitre.org/data/graphs/928.html

[b-iframe] W3C (2014), *HTML <iframe> Tag*.
http://www.w3schools.com/tags/tag_iframe.asp

[b-NA] Idika, Nwokedi, and Mathur, Aditya P. (2007), A Survey of Malware Detection Techniques, Department of Computer Science, Purdue University, 2 February. http://www.serc.net/system/files/SERC-TR-286.pdf

[b-NIST SP 800-83] NIST Special Publication 800-83 (2005), Guide to Malware Incident Prevention and Handling.

[b-NTobjectives] Kuykendall, Dan (2009), Is Your Website Already Infected? Analyzing and Detecting Malicious Content, 20 March. http://www.manvswebapp.com/is-your-website-already-infected.

[b-OWASP] OWASP (2013), OWASP Top 10 application security risks. https://www.owasp.org/index.php/Top_10.

SÉRIES DES RECOMMANDATIONS UIT-T

Série A	Organisation du travail de l'UIT-T
Série D	Principes généraux de tarification
Série E	Exploitation générale du réseau, service téléphonique, exploitation des services et facteurs humains
Série F	Services de télécommunication non téléphoniques
Série G	Systèmes et supports de transmission, systèmes et réseaux numériques
Série H	Systèmes audiovisuels et multimédias
Série I	Réseau numérique à intégration de services
Série J	Réseaux câblés et transmission des signaux radiophoniques, télévisuels et autres signaux multimédias
Série K	Protection contre les perturbations
Série L	Construction, installation et protection des câbles et autres éléments des installations extérieures
Série M	Gestion des télécommunications y compris le RGT et maintenance des réseaux
Série N	Maintenance: circuits internationaux de transmission radiophonique et télévisuelle
Série O	Spécifications des appareils de mesure
Série P	Terminaux et méthodes d'évaluation subjectives et objectives
Série Q	Commutation et signalisation
Série R	Transmission télégraphique
Série S	Equipements terminaux de télégraphie
Série T	Terminaux des services télématiques
Série U	Commutation télégraphique
Série V	Communications de données sur le réseau téléphonique
Série X	**Réseaux de données, communication entre systèmes ouverts et sécurité**
Série Y	Infrastructure mondiale de l'information, protocole Internet et réseaux de prochaine génération
Série Z	Langages et aspects généraux logiciels des systèmes de télécommunication

Imprimé en Suisse
Genève, 2015