



الوكالة الوطنية للسلامة المعلوماتية
Agence Nationale de la Sécurité Informatique

GUIDE DES BONNES PRATIQUES POUR LA SECURITE DES PLATEFORMES WEB

V2.0

Sécurité, Web, Serveur, Application, Hébergement

La sécurité des applications web est de plus en plus menacée. Ce document traite les bonnes pratiques recommandées pour aider les administrateurs et les développeurs à sécuriser leurs plateformes web qui englobe le système d'exploitation, le serveur web, le serveur base de données, le contenu web et l'infrastructure du réseau d'hébergement.

GESTION DOCUMENTAIRE

Auteur	Version	Date de la version	Modification apportée
ANSI/SET	1.0	13/04/2009	Premier draft
ANSI/SET	1.2	25/05/2009	Première révision
ANSI/SET	2.0	14/04/2010	Deuxième version

TABLE DES MATIERES

1. INTRODUCTION	4
2. SECURISATION DU SYSTEME D'EXPLOITATION DU SERVEUR WEB.....	5
2.1. Installation et configuration du système d'exploitation	5
2.1.1. Patch et mise à niveau du système d'exploitation	5
2.1.2. Supprimer ou désactiver les services et applications inutiles	5
2.1.3. Configurer l'authentification des utilisateurs du système d'exploitation	6
2.1.4. Installer et configurer des outils de sécurité supplémentaires	6
2.2. Evaluation de la sécurité du système d'exploitation	7
2.3. Check-list de sécurité du système d'exploitation du serveur web	7
3. SECURISATION DU SERVEUR WEB.....	8
3.1. Installation sécurisée du serveur web	8
3.2. Configuration des contrôles d'accès	9
3.3. Configuration sécurisée du répertoire de contenu web	9
3.4. Check-list de sécurité du serveur web.....	9
4. SECURISATION DU SERVEUR BASE DE DONNEES	11
4.1. Configuration sécurisée du serveur base de données	11
4.2. Check-list de sécurité du serveur base de données	12
5. SECURISATION DU CONTENU WEB	13
5.1. Publication d'informations sur les sites web publics	13
5.2. Considérations de sécurité côté client et serveur	13
5.2.1 Vulnérabilités côté client	14
5.2.2 Considérations côté serveur	14
5.3. Check-list pour la sécurisation du contenu web	16
6. SECURISATION DES COMMUNICATIONS.....	18
6.1. Check-list pour la sécurité des communications	18
7. IMPLEMENTATION D'UNE INFRASTRUCTURE RESEAU SECURISEE	20
7.1. Emplacements déconseillés pour l'hébergement d'un serveur web.....	20
7.2. Zone Démilitarisée (DMZ)	20
7.3. Configuration des éléments du réseau	23
7.3.1. Configuration Routeur / Firewall	24
7.3.2. Systèmes de détection et de prévention d'intrusions (IDS/IPS).....	25
7.3.3. Commutateurs réseau	26
7.3.4. Répartiteurs de charge (Load balancers)	26
7.3.5. Reverse Proxy	26
7.4. Check-list pour mettre en place une infrastructure réseau sécurisée	26

8. ADMINISTRATION SECURISEE DU SERVEUR WEB	28
8.1. Journalisation	28
8.2. Sauvegarde du serveur web	28
8.3. Audit périodique de la plateforme web	29
8.3.1. Scan de vulnérabilités	29
8.3.2. Test de pénétration	30
8.3.3. Audit de code source	30
8.4. Supervision	30
8.4.1. Supervision système	31
8.4.2. Supervision réseau	31
8.4.3. Supervision des applications	31
8.5. Gestion des incidents	31
8.6. Check-list pour la gestion du serveur web	32
9. BIBLIOGRAPHIE	35
10. ANNEXES	36

1. INTRODUCTION

Une plateforme web est l'ensemble de composants matériels et logiciels configuré et connecté à Internet permettant de servir des pages web sur demande. Les informations sur les serveurs web public peuvent être consultées par les internautes n'importe où sur Internet. De ce fait, ils peuvent être soumis à des tentatives d'attaques par les pirates.

Les pirates peuvent modifier le contenu des sites web et voler des données critiques du système. Cela peut se traduire par une perte importante de revenu si c'est une institution financière ou un site e-commerce et une perte ou vol de données pour d'autres entreprises.

Un incident de web defacement peut causer aussi bien des dommages importants à l'image de marque d'une entreprise. Les menaces communes à un serveur web public peuvent être classées comme suit :

- Accès non autorisé
 - Defacement
 - Vol de données
 - Manipulation de données
- Mauvaise utilisation
 - Lancement d'attaques
 - Hébergement de contenus malicieux
- Déni de Service
- Menaces physiques

Ce document traite les bonnes pratiques recommandées pour aider les administrateurs et les développeurs à sécuriser leurs plateformes web qui englobe le système d'exploitation, le serveur web, le serveur base de données, le contenu web et l'infrastructure du réseau d'hébergement.

2. SECURISATION DU SYSTEME D'EXPLOITATION DU SERVEUR WEB

La première étape de sécurisation d'un serveur web est le durcissement du système d'exploitation sous-jacent.

La majorité des systèmes d'exploitation sont configurés par défaut ; des configurations matérielles et logicielles qui sont généralement fixées par les fabricants qui mettent l'accent sur les caractéristiques, les fonctions et la facilité d'utilisation, au détriment de la sécurité. Ceci du fait que ces constructeurs ne sont pas conscients des besoins en sécurité de chaque entreprise. Pour cela, chaque administrateur doit configurer de nouveaux les systèmes d'exploitation afin de refléter les exigences de sécurité de sa plateforme.

2.1. Installation et configuration du système d'exploitation

2.1.1. Patch et mise à niveau du système d'exploitation

Une fois un système d'exploitation est installé, l'application des correctifs ou mises à niveau nécessaires pour corriger les vulnérabilités connues est essentielle.

Toute vulnérabilité connue sur l'OS utilisé pour l'hébergement devrait être corrigée avant la mise en exploitation du serveur web sinon il sera exposé à des utilisateurs malveillants.

Pour détecter correctement et corriger ces failles, les administrateurs de serveur web doivent faire ce qui suit:

- Créer, documenter et mettre en place une procédure de patch
- Identifier les vulnérabilités et les patches manquants
 - Pour vérifier les vulnérabilités des systèmes d'exploitation, des services et d'autres applications, vous pouvez consulter :
<http://www.securityfocus.com/vulnerabilities>, le NIST National Vulnerability Database (NVD) à l'adresse <http://nvd.nist.gov/>, etc. (voir annexe pour plus de ressources)
- Installer les correctifs et les mises à jour à partir du site web officiel de l'OS utilisé
- Si des correctifs ne sont pas encore disponibles, désactiver les services qui sont en relation avec la vulnérabilité si cela est possible

2.1.2. Supprimer ou désactiver les services et applications inutiles

Il est fortement recommandé qu'un serveur web soit sur un hôte dédié. Lors de la configuration du système d'exploitation :

- Désactiver ou supprimer tous les services et applications inutiles (réactiver seulement ceux requis par le serveur web)

Si possible, installer la configuration OS minimale, puis ajouter les services et les applications nécessaires.

Parmi les services et applications qui devraient normalement être désactivée si non nécessaire sont les suivants:

- les services de partage de fichiers et d'imprimantes (par exemple Windows Network Basic Input / Output System [NetBIOS], Network File System [NFS], File Transfer Protocol [FTP])
- les services sans fil
- les programmes d'accès à distance, en particulier ceux qui ne cryptent pas leurs communications (par exemple, Telnet)
- Lightweight Directory Access Protocol [LDAP], Kerberos, Network Information System [NIS])
- les services e-mail (par exemple, Simple Mail Transfer Protocol [SMTP])
- les compilateurs de langage et les bibliothèques
- Les outils de développement
- Les outils et utilitaires de gestion système et réseau, y compris Simple Network Management Protocol (SNMP)

2.1.3. Configurer l'authentification des utilisateurs du système d'exploitation

Pour s'assurer qu'une authentification appropriée des utilisateurs est en place, il fallait prendre les mesures suivantes :

- Supprimer ou désactiver les comptes par défaut et les groupes inutiles
- Vérifier le choix des mots de passe (Longueur, Complexité, Réutilisation, ...)
- Prévenir le devine mot de passe (par exemple, refuser la connexion après un nombre défini de tentatives non réussis)
- Installer et configurer d'autres mécanismes de sécurité pour renforcer l'authentification

2.1.4. Installer et configurer des outils de sécurité supplémentaires

Les systèmes d'exploitation, souvent, ne comprennent pas tous les outils nécessaires pour garantir la sécurité du système d'exploitation, des services et des applications de manière adéquate. Pour cela, les administrateurs ont besoin de choisir, installer et configurer des logiciels supplémentaires pour assurer une sécurité plus efficace tels que :

- Les logiciels anti-malware : tels que les logiciels antivirus, les logiciels anti spyware et les détecteurs de rootkit ceci afin de protéger le système d'exploitation

locale des logiciels malveillants et afin de détecter et éliminer toutes les infections qui peuvent se produire.

- Les logiciels de détection et de prévention d'intrusion hôte (HIDS)

2.2. Evaluation de la sécurité du système d'exploitation

Le test périodique de la sécurité de l'OS est un moyen essentiel pour identifier les vulnérabilités et veiller à ce que les mesures de sécurité existantes sont efficaces.

Les méthodes courantes pour tester l'OS sont notamment le scan de vulnérabilités et les tests de pénétration.

Le scan des vulnérabilités devraient être effectués périodiquement, au moins hebdomadaire à mensuel et les tests de pénétration doivent être effectuées au moins chaque année (Pour plus d'informations, consulter la partie 8.1.3 : Audit périodique de la plateforme web)

2.3. Check-list de sécurité du système d'exploitation du serveur web

Complété	Action
	Patch et mise à niveau du système d'exploitation
<input type="checkbox"/>	Créer, documenter et mettre en place une procédure de patch
<input type="checkbox"/>	Tester les patchs sur un serveur de test avant de les appliquer sur le serveur en exploitation
<input type="checkbox"/>	Identifier et installer tous les correctifs et mises à niveau du système d'exploitation
<input type="checkbox"/>	Identifier et installer tous les correctifs et mises à niveau des applications et des services inclus avec le système d'exploitation
<input type="checkbox"/>	Installer les correctifs et les mises à jour à partir du site web officiel de l'OS utilisé
<input type="checkbox"/>	Si des correctifs ne sont pas encore disponibles, désactiver les services qui sont en relation avec la vulnérabilité identifiée si cela est possible
	Supprimer ou désactiver les services et applications inutiles
<input type="checkbox"/>	Désactiver ou supprimer tous les services et applications inutiles
	Configurer l'authentification des utilisateurs du système d'exploitation
<input type="checkbox"/>	Supprimer ou désactiver les comptes et les groupes par défaut ou inutiles
<input type="checkbox"/>	Vérifier le choix des mots de passe (Longueur, Complexité, Réutilisation, ...)

<input type="checkbox"/>	Prévenir le devine de mot de passe (par exemple, refuser la connexion après un nombre défini de tentatives non réussis)
<input type="checkbox"/>	Installer et configurer d'autres mécanismes de sécurité pour renforcer l'authentification
	Installer et configurer des contrôles de sécurité supplémentaires
<input type="checkbox"/> <input type="checkbox"/>	Choisir, installer et configurer des logiciels supplémentaires pour assurer les contrôles nécessaires ne figurant pas dans le système d'exploitation, tels que : <ul style="list-style-type: none"> ▪ les logiciels anti malwares : (antivirus, logiciel anti-espion, les détecteurs de rootkit) ▪ des logiciels de détection et de prévention d'intrusion hôte (HIDS)
	Evaluation de la sécurité du système d'exploitation
<input type="checkbox"/>	Effectuer le scan de vulnérabilité de l'OS après l'installation initiale afin d'identifier les vulnérabilités
<input type="checkbox"/>	Tester l'OS périodiquement pour identifier de nouvelles vulnérabilités
<input type="checkbox"/>	Effectuer un test de pénétration au moins une fois par an

3. SECURISATION DU SERVEUR WEB

3.1. Installation sécurisée du serveur web

- Installer le logiciel serveur web sur un serveur dédié ou sur un système d'exploitation virtualisé.
- Appliquer les correctifs et les mises à jour pour neutraliser les vulnérabilités connues
- Créer un disque physique dédié ou une partition logique (séparé de l'OS et du serveur web) pour le contenu web
- Supprimer ou désactiver tous les services inutiles installés avec le serveur web (par exemple, Gopher, FTP, administration à distance)
- Supprimer ou désactiver tous les comptes par défaut ou inutiles créés lors de l'installation du serveur web
- Supprimer du serveur toutes les documentations du constructeur
- Supprimer tous les fichiers et répertoires inutiles à partir du serveur (les fichiers de test, les scripts, codes exécutables, etc.)
- Appliquer un modèle de sécurité approprié ou suivre un guide de hardening du serveur web

- Reconfigurer la bannière HTTP afin de ne pas signaler le type du serveur web et la version de l'OS

3.2. Configuration des contrôles d'accès

- Configurer le processus du serveur web à s'exécuter en tant qu'un simple utilisateur avec une limite des privilèges
- Configurer le serveur web en lecture seule sur les répertoires de l'application web
- Configurer le serveur web afin que seuls les processus autorisés pour l'administration de serveurs web puissent écrire des fichiers
- Configurer le système d'exploitation pour que le serveur web puisse écrire des fichiers journaux mais pas les lire
- Installer le contenu web sur un autre disque dur ou une partition logique autre que celle de l'OS et du serveur web
- Si l'écriture est autorisée sur le serveur web, limiter la taille des fichiers à uploader sur l'espace disque qui est dédié à cet effet. Les fichiers ajoutés doivent être placés sur une partition séparée
- S'assurer que les fichiers journaux sont stockés dans un emplacement qui est dimensionné de façon appropriée; les fichiers journaux doivent être placés sur une partition séparée
- Configurer le nombre maximal de processus de serveur web et / ou des connexions réseau que le serveur web doit permettre
- S'assurer que les utilisateurs et les administrateurs sont en mesure de changer leurs mots de passe périodiquement
- Désactiver les utilisateurs après une certaine période d'inactivité

3.3. Configuration sécurisée du répertoire de contenu web

- Dédier un seul disque dur ou une partition logique pour les contenus web
- Définir un répertoire exclusif pour tous les scripts externes ou les programmes exécutable (par exemple, CGI, ASP, PHP)
- Désactiver l'utilisation des liens physiques ou symboliques (par exemple, des raccourcis pour Windows)
- Définir une matrice d'accès au contenu web permettant d'identifier qui peut accéder aux dossiers et fichiers du contenu du serveur web
- Configurer la protection anti-spambot (par exemple, les CAPTCHA, mail [at] mail [point] com au lieu de mail@mail.com)

3.4. Check-list de sécurité du serveur web

Complété	Action
	Installation sécurisée du serveur web
<input type="checkbox"/>	Installer le logiciel serveur web sur un serveur dédié ou sur un système d'exploitation virtualisé

<input type="checkbox"/>	Appliquer les correctifs et les mises à jour pour neutraliser les vulnérabilités connues
<input type="checkbox"/>	Créer un disque physique dédié ou une partition logique (séparé de l'OS et du serveur web) pour le contenu Web
<input type="checkbox"/>	Supprimer ou désactiver tous les services inutiles installés avec le serveur web (par exemple, Gopher, FTP, administration à distance)
<input type="checkbox"/>	Supprimer ou désactiver tous les comptes de connexion par défaut ou inutiles créés lors de l'installation du serveur web
<input type="checkbox"/>	Enlever toute la documentation du fabricant du serveur web
<input type="checkbox"/>	Supprimer tous les fichiers et répertoires inutiles à partir du serveur (les fichiers de test, les scripts, codes exécutables, etc.)
<input type="checkbox"/>	Appliquer un modèle de sécurité approprié ou suivre un guide de hardening du serveur
<input type="checkbox"/>	Reconfigurer la bannière http afin de ne pas signaler le type du serveur web et la version de l'OS
	Configuration des contrôles d'accès
<input type="checkbox"/>	Configurer le processus du serveur web à exécuter en tant qu'un simple utilisateur avec une limite des privilèges
<input type="checkbox"/>	Configurer le serveur web en lecture seule sur les répertoires de l'application web
<input type="checkbox"/>	Configurer le serveur web afin que seuls les processus autorisés pour l'administration de serveurs web puissent écrire des fichiers
<input type="checkbox"/>	Configurer le système d'exploitation pour que le serveur web puisse écrire des fichiers journaux mais pas les lire
<input type="checkbox"/>	Si l'écriture est autorisée sur le serveur web, limiter la taille des fichiers à uploader sur l'espace disque qui est dédié à cet effet. Les fichiers ajoutés doivent être placés sur une partition séparée
<input type="checkbox"/>	S'assurer que les fichiers journaux sont stockés dans un emplacement qui est dimensionné de façon appropriée; les fichiers journaux doivent être placés sur une partition séparée
<input type="checkbox"/>	Configurer le nombre maximal de processus de serveur Web et / ou des connexions réseau que le serveur Web doit permettre
<input type="checkbox"/>	S'assurer que les utilisateurs et les administrateurs sont en mesure de changer leurs mots de passe périodiquement
<input type="checkbox"/>	Désactiver les utilisateurs après une certaine période d'inactivité

	Configuration sécurisée du répertoire de contenu web
<input type="checkbox"/>	Dédier un seul disque dur ou une partition logique pour les contenus web
<input type="checkbox"/>	Définir un répertoire exclusif pour tous les scripts externes ou les programmes exécutable (par exemple, CGI, ASP, PHP)
<input type="checkbox"/>	Désactiver l'utilisation des liens physiques ou symboliques (par exemple, des raccourcis pour Windows)
<input type="checkbox"/>	Définir une matrice d'accès au contenu web permettant d'identifier qui peut accéder aux dossiers et fichiers du contenu du serveur web
<input type="checkbox"/>	Configurer la protection anti-spam (par exemple, les CAPTCHA, mail [at] mail [point] com au lieu de mail@mail.com)

4. SECURISATION DU SERVEUR BASE DE DONNEES

Une base de données est installée comme étant un composant de serveur de back-end au service d'une application web grâce à l'utilisation du langage de requête, généralement SQL.

4.1. Configuration sécurisée du serveur base de données

Assurer la sécurité de la base de données est primordiale et doit être mis en place afin de protéger les données et limiter l'accès seulement aux utilisateurs autorisés.

Les points suivants devraient être pris en considération pour garantir la sécurité d'un système de gestion de base de données :

- Mettre à jour le SGBD avec les derniers correctifs stables
- Utiliser des algorithmes de hachage/cryptage pour stocker les données critiques
- Sécuriser le serveur de base de données derrière un firewall et utiliser un IDS pour détecter toute tentative d'intrusion
- Le processus serveur base de données devrait fonctionner comme étant un utilisateur avec des privilèges minimum et jamais en tant qu'administrateur
- Mettre en place une politique stricte de contrôle d'accès physique et logique
- Activer les logs sur les tables jugés critiques
- Certains serveurs de base de données comprennent des serveurs d'applications par défaut. Il est recommandé qu'ils soient supprimés s'ils sont inutiles
- Le serveur de base de données ne devrait pas avoir une adresse IP accessible au public
- L'accès à la base de données ne devrait être autorisé qu'à partir du serveur web sur un port bien particulier

Voici quelques références intéressantes :

Microsoft:

SQL Server Security Center

<http://www.microsoft.com/technet/security/prodtech/dbsql/default.msp>

Microsoft:

SQL Server Best Practices Analyzer

<http://www.microsoft.com/downloads/details.aspx?FamilyID=b352eb1f-d3ca-44ee-893E-9e07339c1f22&displaylang=fr>

CISecurity:

Oracle Security Testing tools and guide

www.cisecurity.com

Autres

<http://www.petefinnigan.com>

<http://www.appsecinc.com>

4.2. Check-list de sécurité du serveur base de données

Complété	Action
<input type="checkbox"/>	Mettre à jour le SGBD avec les derniers correctifs stables
<input type="checkbox"/>	Utiliser des algorithmes de hachage/cryptage pour stocker les données critiques
<input type="checkbox"/>	Sécuriser le serveur de base de données derrière un firewall et utiliser un IDS pour détecter toute tentative d'intrusion
<input type="checkbox"/>	Le processus serveur base de données devrait fonctionner comme étant un utilisateur avec des privilèges minimum et jamais en tant qu'administrateur
<input type="checkbox"/>	Mettre en place une politique stricte de contrôle d'accès physique et logique
<input type="checkbox"/>	Activer les logs sur les tables jugés critiques
<input type="checkbox"/>	Certains serveurs de base de données comprennent des serveurs d'applications par défaut. Il est recommandé qu'ils soient supprimés s'ils sont inutiles
<input type="checkbox"/>	Le serveur de base de données ne devrait pas avoir une adresse IP accessible au public
<input type="checkbox"/>	L'accès à la base de données ne devrait être autorisé qu'à partir du serveur web sur un port bien particulier

5. SECURISATION DU CONTENU WEB

Trop souvent, peu de réflexion est donnée à la sécurité du contenu du site web. Le choix des informations à publier sur le site web doit être bien étudié.

5.1. Publication d'informations sur les sites web publics

Un site web public ne doit pas contenir les informations suivantes:

- Documents classifiés (document privé, confidentiel, top secret)
- Procédures internes
- Informations sensibles ou propriétaires
- Renseignements sur le personnel de l'entreprise (tels que les adresses, les numéros de téléphone, les membres de famille des personnels, etc.)
- Politique et procédures de sécurité de l'information
- Information concernant le réseau et l'infrastructure de système d'information (par exemple, des plages d'adresses, les conventions de nommage)
- Données qui impliquent des informations sur la sécurité physique de l'entreprise (plans, cartes, schémas, photographies aériennes et de plans architecturaux du bâtiment de l'entreprise)
- Information sur le plan de continuité d'activité de l'entreprise (détails sur les procédures d'intervention d'urgence, les voies d'évacuation, le personnel responsable)

Prévoir une procédure pour décider des informations à publier sur le site web. Une telle procédure devrait comprendre les étapes suivantes:

- Identifier les informations qui doivent être publiés sur le web
- Identifier le public cible (pourquoi publier si aucune audience n'existe?)
- Identifier les conséquences négatives de la publication des informations
- Déterminer qui doit être responsable de la création, la publication et le maintien de ces informations
- Publier ces informations
- Vérifier les informations publiées
- Revoir périodiquement les informations publiées pour vérifier la conformité du contenu avec les lignes directrices de l'entreprise

5.2. Considérations de sécurité côté client et serveur

Aujourd'hui, divers types d'éléments interactifs ont été introduites offrant de nouvelles façons aux utilisateurs d'interagir de manière plus dynamique avec les sites web. Toutefois, ces éléments interactifs mettent en place de nombreuses vulnérabilités liées au web.

Une variété de technologies de contenu actif existe tels que ActiveX, VBScript, JavaScript, Asynchronous JavaScript and XML (AJAX). L'utilisation du contenu actif exige souvent les utilisateurs à réduire les paramètres de sécurité sur leur navigateur web afin de pouvoir s'exécuter. Si ce n'est pas correctement mis en œuvre, le contenu actif peut présenter une menace grave pour l'utilisateur final.

5.2.1 Vulnérabilités côté client

Chaque technologie de contenu actif a ses forces et ses faiblesses, aucune n'est parfaitement sécurisée. Tout administrateur web ou webmaster qui envisage déployer un site web avec des fonctions qui exigent une technologie de contenu actif côté client doit soigneusement évaluer les risques et les avantages de la technologie avant son utilisation.

Parce que ces technologies consistent à placer le code au niveau du client, un attaquant peut tenter de désassembler le code pour comprendre sa relation avec l'application web et exploiter cette relation. Parmi ces technologies, on cite : JavaScript, Adobe Flash, Adobe Shockwave, AJAX, Visual Basic Script (VBScript) et ActiveX.

Parmi les attaques côté client les plus connues, on cite le Cross Site Scripting **XSS** qui exploite les vulnérabilités des pages web dynamiques (écrites en PHP, ASP ou JSP). Un hacker peut alors récupérer les cookies (normalement utilisés pour identifier un utilisateur sur un site web) ce qui lui permet de se connecter sur le système cible sous l'identité de sa victime.

5.2.2 Considérations côté serveur

Les applications côté serveur peuvent être écrites en utilisant différents langages de programmation web. Si les scripts et les composants ne sont pas développés avec soin, les attaquants peuvent trouver et exploiter des failles dans le code afin de pénétrer dans le serveur web ou dans les composants backend. Par conséquent, les scripts doivent être développés en tenant compte des besoins en sécurité.

Les générateurs de contenu côté serveur peuvent créer les failles de sécurité suivantes sur le serveur:

- Ils peuvent créer des fuites d'informations sur la plateforme web ce qui peut servir à un attaquant à préparer une attaque ciblée
- Ils peuvent, intentionnellement ou non négliger, le contrôle de la valeur des inputs au niveau des formulaires remplis par les utilisateurs, des paramètres d'URL ou au niveau des requêtes de recherche facilitant ainsi à un attaquant l'exécution de commandes arbitraires (exemple : XSS, SQL injection)
- Ils peuvent permettre à des attaquants de modifier le contenu du site web (exemple : Web defacement, File include)

Lors de l'examen ou le développement du code, prendre ce qui suit en considération :

- Un contrôle sur les différents inputs est nécessaire afin de filtrer et valider la taille et les valeurs des paramètres d'entrée de sorte qu'un attaquant ne peut pas dépasser les limites de la mémoire ou exécuter des commandes arbitraires.
- Le code doit utiliser les noms de chemin d'accès explicite lors de l'invocation des programmes externes. Il n'est pas recommandé d'utiliser la variable d'environnement PATH pour résoudre les noms de chemin d'accès partiel
- Pour les formulaires contenant des champs de saisie des données, établir une liste des caractères attendus et filtrer les caractères inattendus avant de traiter un formulaire. Par exemple, sur la plupart des formulaires, les données attendues sont généralement dans ces catégories: les lettres **a-z, A-Z et 0-9**. Des précautions doivent être prises au moment d'accepter des caractères spéciaux tels que **&, ', ", @, et !**. Ces symboles peuvent être mal interprétés par le langage de programmation utilisé
- Les cookies doivent être examinés en filtrant tous les caractères spéciaux
- Un mécanisme de cryptage doit être utilisé pour crypter les mots de passe entrés à travers les formulaires d'authentification pour éviter leur passage en clair sur le réseau
- Pour les applications web qui sont restreints par nom d'utilisateur et mot de passe, aucune des pages web dans l'application ne devrait être accessible sans passer par la page d'authentification
- Eliminer tous les exemples de scripts, les exécutables ou toute documentation installés inutilement avec le serveur web.
- Limiter l'utilisation des fonctions qui lisent, écrivent ou exécutent des fichiers sur le serveur car ce type de code peut violer les restrictions d'accès et modifier ou endommager le système
- Vérifier l'interaction du code avec d'autres programmes ou applications afin d'identifier les failles de sécurité

L'emplacement et la configuration des permissions affectés aux répertoires du contenu web doit être bien étudié. En effet, il est recommandé ce qui suit :

- Les fichiers accessibles en écriture devraient être identifiés et placés dans des dossiers distincts. Aucun fichier script ne doit exister dans des dossiers accessibles en écriture
- Les fichiers exécutables (par exemple, CGI, .EXE, .CMD, et PL) doivent être placés dans des dossiers distincts. Aucun autre document lisible ou accessible en écriture ne devrait être placé dans ces dossiers
- Les fichiers de script (par exemple ASP, PHP et PL) doivent être placés dans des dossiers distincts. Il est également recommandé de stocker ces scripts dans un dossier avec un nom non évident (par exemple, pas "Scripts") pour rendre plus difficile pour un attaquant de trouver les scripts par navigation directe

- Les fichiers Include (par exemple, INC, PHP et ASP) créés pour la réutilisabilité du code doivent être placés dans des répertoires distincts. Notez qu'une grande partie du risque avec les fichiers include est dans leur capacité d'exécution. Si la capacité d'exécution est désactivée, ce risque est considérablement réduit.

5.3. Check-list pour la sécurisation du contenu web

Complété	Action
	Veiller à ce que le site web ne contient pas les informations suivantes :
<input type="checkbox"/>	Documents classifiés (document privé, confidentiel, top secret)
<input type="checkbox"/>	Procédures internes
<input type="checkbox"/>	Informations sensibles ou propriétaires
<input type="checkbox"/>	Renseignements sur le personnel de l'entreprise (tels que les adresses, les numéros de téléphone, les membres de famille des personnels, etc.)
<input type="checkbox"/>	Politique et procédures de sécurité de l'information
<input type="checkbox"/>	Information concernant le réseau et l'infrastructure du système d'information (par exemple, des plages d'adresses, les conventions de nommage)
<input type="checkbox"/>	Données qui impliquent des informations sur la sécurité physique de l'entreprise (plans, cartes, schémas, photographies aériennes et de plans architecturaux du bâtiment de l'entreprise)
<input type="checkbox"/>	Information sur le plan de continuité d'activité de l'entreprise (détails sur les procédures d'intervention d'urgence, les voies d'évacuation, le personnel responsable)
	Établir une politique organisationnelle formelle bien documentée et un processus d'approbation des contenus web publics:
<input type="checkbox"/>	Identifier les informations qui doivent être publiés sur le web
<input type="checkbox"/>	Identifier le public cible
<input type="checkbox"/>	Identifier les conséquences négatives de la publication des informations
<input type="checkbox"/>	Déterminer qui doit être responsable de la création, la publication et le maintien de ces informations
<input type="checkbox"/>	Publier ces informations

<input type="checkbox"/>	Vérifier les informations publiées
<input type="checkbox"/>	Revoir périodiquement les informations publiées pour vérifier la conformité de contenu avec les lignes directrices de l'entreprise
	Considérations côté client sur la sécurité du contenu actif
<input type="checkbox"/>	Estimer (peser) les risques et les avantages du contenu actif côté client
<input type="checkbox"/>	Ne prendre aucune action sans l'autorisation expresse de l'utilisateur
<input type="checkbox"/>	Si possible, utilisez uniquement le contenu actif largement adopté tel que JavaScript, PDF et Flash
<input type="checkbox"/>	Lorsque c'est possible, proposer des alternatives (par exemple, HTML fourni avec PDF)
	Considérations côté serveur
<input type="checkbox"/>	Un contrôle sur les différents inputs est nécessaire afin de filtrer et valider la taille et les valeurs des paramètres d'entrée de sorte qu'un attaquant ne peut pas dépasser les limites de la mémoire ou exécuter des commandes arbitraires
<input type="checkbox"/>	Le code doit utiliser les noms de chemin d'accès explicite lors de l'invocation des programmes externes. Il n'est pas recommandé d'utiliser la variable d'environnement PATH pour résoudre les noms de chemin d'accès partiel
<input type="checkbox"/>	Pour les formulaires contenant des champs de saisie des données, établir une liste des caractères attendus et filtrer les caractères inattendus avant de traiter un formulaire. Par exemple, sur la plupart des formulaires, les données attendues sont généralement dans ces catégories: les lettres a-z, A-Z et 0-9. Des précautions doivent être prises au moment d'accepter des caractères spéciaux tels que &, ', ", @, et ! Ces symboles peuvent être mal interprétés par le langage de programmation utilisé
<input type="checkbox"/>	Les cookies doivent être examinés en filtrant tous les caractères spéciaux
<input type="checkbox"/>	Un mécanisme de cryptage doit être utilisé pour crypter les mots de passe entrés à travers les formulaires d'authentification pour éviter leur passage en clair sur le réseau
<input type="checkbox"/>	Pour les applications web qui sont restreints par nom d'utilisateur et mot de passe, aucune des pages web dans l'application ne devrait être accessible sans passer par la page d'authentification
<input type="checkbox"/>	Éliminer tous les exemples de scripts, les exécutables ou toute documentation installés inutilement avec le serveur web

<input type="checkbox"/>	limiter l'utilisation des fonctions qui lisent, écrivent ou exécutent des fichiers sur le serveur car ce type de code peut violer les restrictions d'accès et modifier ou endommager le système
<input type="checkbox"/>	Vérifier l'interaction du code avec d'autres programmes ou applications afin d'identifier les failles de sécurité
<input type="checkbox"/>	Les fichiers accessibles en écriture devraient être identifiés et placés dans des dossiers distincts. Aucun fichier script ne doit exister dans des dossiers accessibles en écriture
<input type="checkbox"/>	Les fichiers exécutables (par exemple, CGI, .EXE, .CMD, et PL) doivent être placés dans des dossiers distincts. Aucun autre document lisible ou accessible en écriture ne devrait être placé dans ces dossiers
<input type="checkbox"/>	Les fichiers de script (par exemple, ASP, PHP et PL) doivent être placés dans des dossiers distincts. Il est également recommandé de stocker ces scripts dans un dossier avec un nom non évident (par exemple, pas "Scripts") pour rendre plus difficile pour un attaquant de trouver les scripts par navigation directe
<input type="checkbox"/>	Les fichiers Include (par exemple, INC, PHP et ASP) créés pour la réutilisabilité du code doivent être placés dans des répertoires distincts.

6. SECURISATION DES COMMUNICATIONS

Le chiffrement est utilisé pour protéger les données transmis entre un client web et un serveur web public. Sans cryptage, n'importe qui ayant accès au trafic réseau peut écouter, et, éventuellement, modifier, le contenu des informations sensibles.

Selon l'importance des ressources web, il est recommandé de choisir un mécanisme d'authentification approprié (authentification basique, authentification par adresse IP, authentification digest, authentification du client et du serveur et cryptage des communications basée sur SSL/TLS).

6.1. Check-list pour la sécurité des communications

Complété	Action
	Configurer l'authentification web et les technologies de chiffrement
<input type="checkbox"/>	Pour les ressources web qui nécessitent une protection minimale et pour lesquels il existe un petit public cible clairement défini, configurer l'authentification basique

<input type="checkbox"/>	Pour les ressources web qui nécessitent une protection supplémentaire, mais pour lesquels il existe un petit public cible clairement défini, configurer l'authentification basée sur l'adresse IP comme une seconde ligne de défense
<input type="checkbox"/>	Pour les ressources web qui nécessitent une protection minimale, mais pour lesquelles il n'existe pas de définition claire du public, configurer une authentification basique ou digest (meilleure)
<input type="checkbox"/>	Pour les ressources web qui nécessitent une protection contre les robots collecteurs ou les robots de bombardement, configurer l'authentification de base ou digest (mieux) ou appliquer d'autres techniques (tels que captcha, nofollow, etc.)
<input type="checkbox"/>	Pour les ressources web qui nécessitent une protection maximale, configurer SSL/TLS
	Configurer SSL/TLS
<input type="checkbox"/>	S'assurer que le SSL / TLS mis en œuvre est entièrement mis à jour
<input type="checkbox"/>	Utiliser un certificat émis par une tierce partie pour l'authentification du serveur
<input type="checkbox"/>	Pour les configurations qui nécessitent un niveau moyen d'authentification du client, configurer le serveur pour exiger un nom d'utilisateur et un mot de passe via SSL / TLS
<input type="checkbox"/>	Pour les configurations qui nécessitent un niveau élevé d'authentification de clients, configurer le serveur à exiger des certificats client via SSL / TLS
<input type="checkbox"/>	S'assurer que les algorithmes de chiffrement faibles sont désactivés
<input type="checkbox"/>	Configurer un contrôleur d'intégrité pour surveiller le certificat de serveur web
<input type="checkbox"/>	Si seulement SSL / TLS doit être utilisé dans le serveur web, s'assurer que l'accès via n'importe quel port TCP autre que le 443 est désactivé
	Protéger contre les attaques de brute force
<input type="checkbox"/>	Utiliser l'authentification forte, si possible (one time password, certificat numérique, etc.)
<input type="checkbox"/>	Verrouiller un compte après un nombre déterminé de tentatives de connexion a échoué
<input type="checkbox"/>	Appliquer une politique de mot de passe
<input type="checkbox"/>	Mettre une liste noire des adresses IP connus de tenter des attaques en brute force
<input type="checkbox"/>	Utiliser un logiciel de contrôle du journal (log) pour détecter les attaques en brute force

7. IMPLEMENTATION D'UNE INFRASTRUCTURE RESEAU SECURISEE

L'infrastructure réseau qui héberge le serveur web joue un rôle critique dans la sécurité du serveur web. Dans la plupart des configurations, l'infrastructure de réseau est la première ligne de défense entre l'Internet et un serveur web public.

Cette section décrit les composants de réseau qui peuvent soutenir et protéger les serveurs web afin de renforcer leur sécurité globale.

7.1. Emplacements déconseillés pour l'hébergement d'un serveur web

- Certaines entreprises choisissent d'implanter leurs serveurs web public sur leurs réseaux de production interne. La principale faiblesse de ce plan est qu'il expose les composants du réseau interne à des risques additionnels. Les serveurs web sont souvent la cible des personnes malveillantes. Si des attaquants réussissent à compromettre un serveur web, ils auront accès au réseau interne et pourront plus facilement compromettre les hôtes internes. Par conséquent, cette disposition n'est pas recommandée.
- Une autre configuration de réseau qui n'est pas recommandée est de placer le serveur web avant le pare-feu d'une entreprise ou un routeur qui permet le filtrage IP. Dans cette architecture, le réseau fournit peu de protection pour le serveur web. En effet, le serveur web doit lui-même maintenir la sécurité : l'OS du serveur web et les applications doivent être extrêmement 'hardened', tous les services inutiles et non sécurisés doivent être désactivés et tous les patches de sécurité nécessaires doivent être appliqués.

7.2. Zone Démilitarisée (DMZ)

Une zone démilitarisée (DMZ) décrit un hôte ou un segment de réseau insérée comme une zone à accès publique entre le réseau privé d'une entreprise et l'Internet.

Il existe une grande variété de configurations DMZ, chacune ayant ses forces et ses faiblesses.

La création d'une DMZ consiste à placer un pare-feu entre le routeur frontal d'une entreprise et son réseau interne, et ceci en créant un nouveau segment de réseau qui ne peut être atteint qu'à travers le dispositif zone démilitarisée. Le serveur web est mis sur le nouveau segment, avec d'autres composants d'infrastructure de réseau et des serveurs qui doivent être accessibles de l'extérieur. Dans certaines configurations, le routeur frontière lui-même peut agir comme un pare-feu de base. La Figure ci-dessous illustre un exemple de cette DMZ simple à l'aide d'un routeur avec des listes de contrôle d'accès (ACL) pour limiter certains types de trafic réseau depuis et vers la zone démilitarisée.

Un seul pare-feu DMZ est une approche à faible coût, car l'entreprise a besoin seulement d'ajouter un pare-feu simple et d'utiliser son routeur frontal existant pour assurer une protection à la zone démilitarisée. Il est généralement opportun que pour les petites entreprises qui font face à un risque minime.

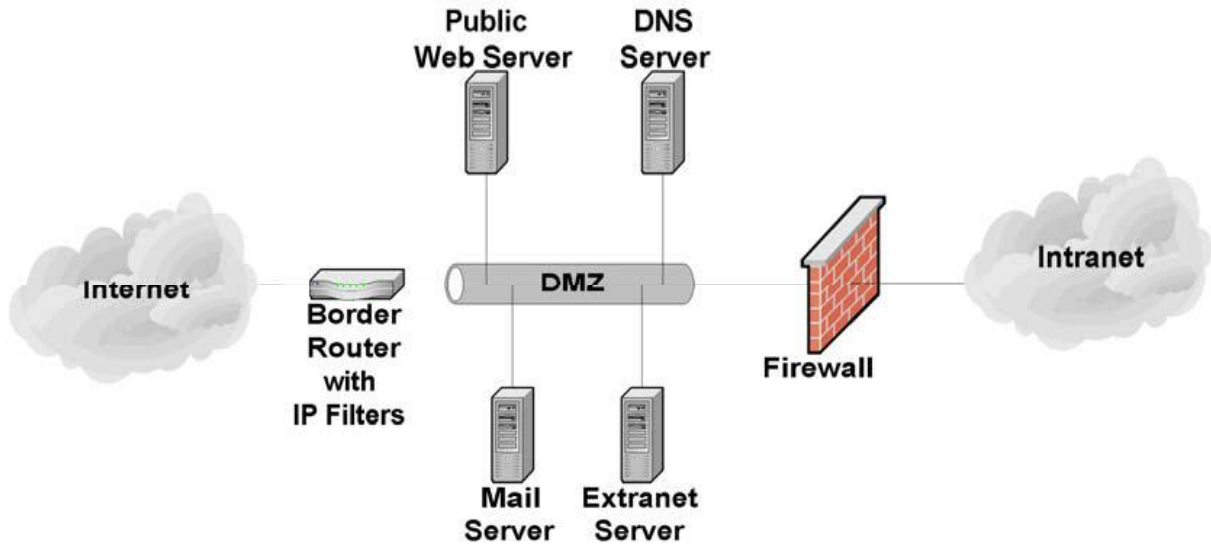


Figure 1 : Simple et unique-Firewall DMZ

La faiblesse fondamentale dans cette approche est que, bien que le routeur est capable de protéger contre la plupart des attaques de réseau, il n'est pas «au courant» des protocoles de la couche d'application de serveur web (par exemple, HTTP) et ne peut donc pas protéger contre les attaques de la couche application visant le serveur web.

Une meilleure approche consiste à ajouter un second pare-feu entre l'Internet et la zone démilitarisée, comme le montre la figure suivante :

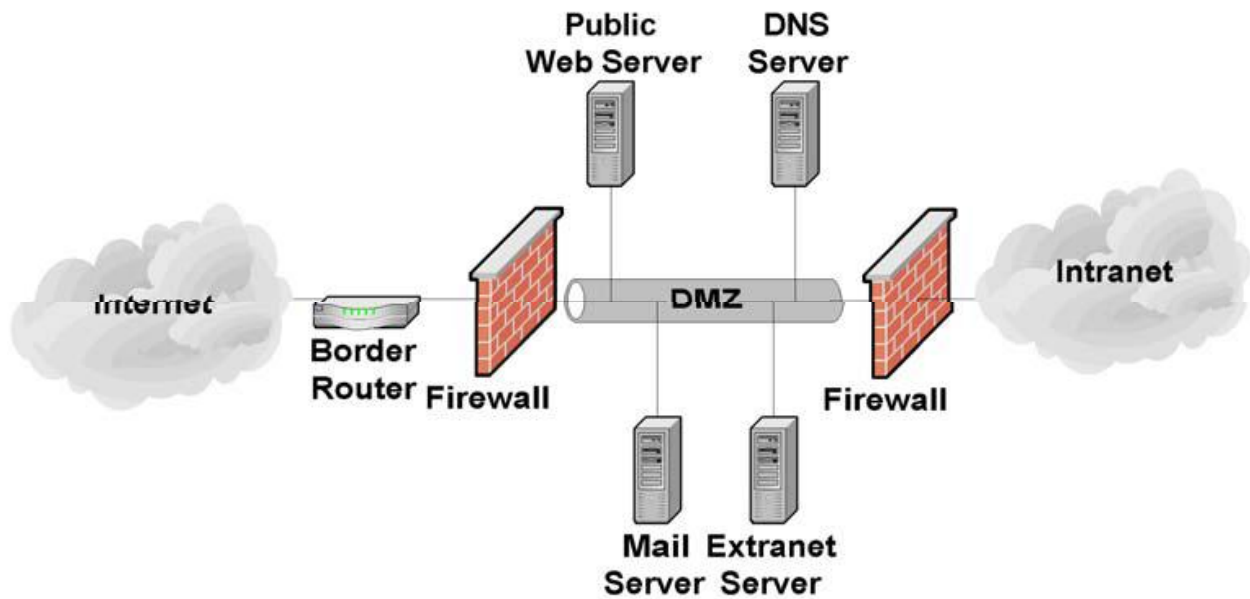


Figure 2 : Deux-Firewall DMZ

Une configuration à deux-firewall DMZ améliore la protection par rapport à un routeur- firewall DMZ car les pare-feu dédiés peuvent avoir plus de règles de sécurité. En outre, comme un pare-feu dédié est souvent en mesure d'analyser le trafic HTTP entrant et sortant, il peut détecter et se défendre contre les attaques de la couche d'application visant le serveur web. Selon l'ensemble des règles définies au niveau des pare-feu et le niveau du trafic de la zone démilitarisée reçu, ce type de zone démilitarisée peut entraîner une certaine dégradation de performances.

Pour les entreprises qui cherchent la sécurité assurée par deux-firewall DMZ mais qui n'ont pas les moyens d'acheter deux pare-feu, une autre option existe ; appelé "service leg" DMZ. Dans cette configuration, un pare-feu est construit avec trois (ou plus) interfaces réseau. Une interface réseau attachée au routeur frontal, une autre attachée au réseau interne et une troisième interface réseau se connectant à la zone démilitarisée (voir Figure 1.3).

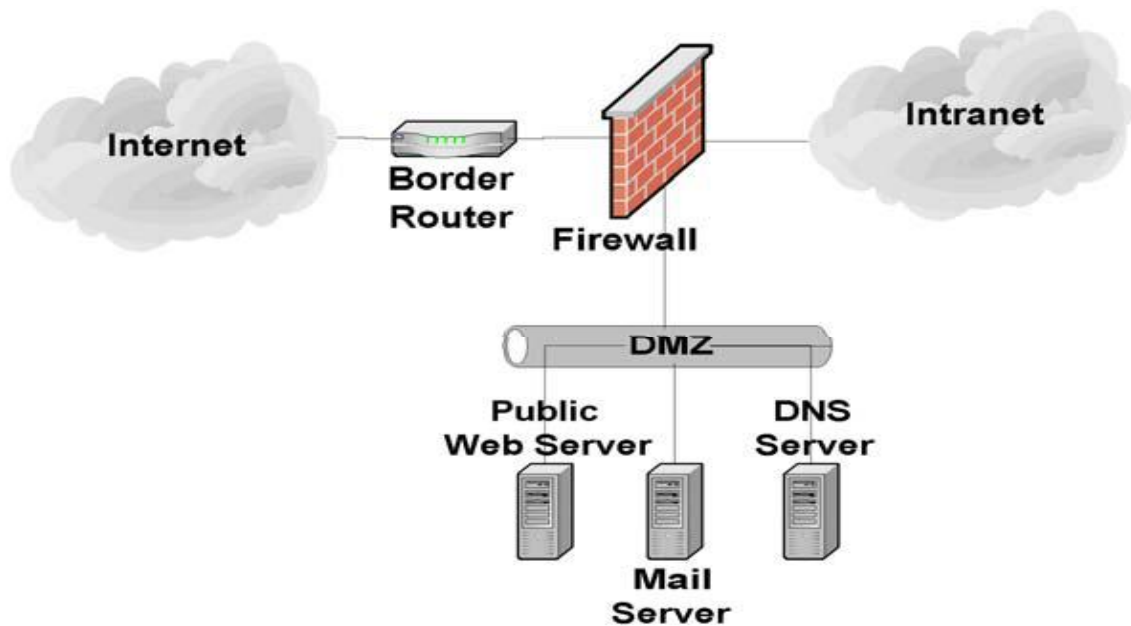


Figure 3 : Service Leg DMZ

Cette configuration du pare-feu est sujette à un risque accru de dégradation du service lors d'une attaque DoS visant la zone démilitarisée. En effet, dans la configuration standard simple et unique Firewall DMZ discuté ci-dessus, une attaque de déni de service contre le serveur web n'affecte généralement que le serveur web. Par contre, dans la configuration du service leg DMZ, le pare-feu est la principale victime d'une attaque DoS puisqu'il fallait examiner tout le trafic réseau avant qu'il atteigne le serveur web (ou toute autre DMZ ou ressource du réseau interne).

Les avantages d'une zone démilitarisée d'un point de vue sécurité sont les suivantes:

- Le serveur web peut être mieux protégé et le trafic réseau depuis et vers le serveur web peut être suivi
- L'attaque visant le serveur web ne menace pas directement le réseau de production interne
- Un meilleur contrôle peut être proposé sur la sécurité du serveur web, car le trafic vers et depuis le serveur web peut être contrôlé
- La configuration de la zone DMZ peut être optimisée pour soutenir et protéger les serveurs web

7.3. Configuration des éléments du réseau

Une fois le serveur web est mis en place dans le réseau, les éléments de l'infrastructure réseau doivent être configurés de façon à le supporter et le protéger. Les principaux éléments d'une infrastructure réseau qui affectent la sécurité du

serveur web sont les pare-feu, les routeurs, les IDS/IPS, les commutateurs, les répartiteurs de charge et les reverse proxy.

Chacun a un rôle important à jouer et qui est essentiel à la stratégie globale de protection du serveur web. En effet, il n'existe pas de "solution miracle". Un pare-feu ou IPS seule ne peut pas protéger adéquatement un serveur web publique de toutes les menaces ou attaques.

7.3.1. Configuration Routeur / Firewall

Il existe plusieurs types de firewalls. Les pare-feu les plus basiques sont ceux qui fonctionnent sur le principe du filtrage simple de paquets. Ensuite, nous avons les firewalls Stateful qui peuvent offrir en plus un contrôle d'accès basé sur TCP et UDP. Les pare-feu les plus puissants sont firewalls applicatifs qui sont en mesure de comprendre et de filtrer le contenu web.

Une perception erronée commune sur les firewalls (et les routeurs agissant comme pare-feu) est qu'ils éliminent tous les risques et peuvent protéger contre une mauvaise configuration du serveur web ou une mauvaise conception de réseau. Malheureusement, ce n'est pas le cas. Les pare-feu et les routeurs sont eux même vulnérables à une mauvaise configuration et à des vulnérabilités logiciels. En outre, beaucoup de firewalls ont des limites pour la couche application où de nombreuses attaques se produisent. Toutefois, il est à noter que les serveurs web peuvent être vulnérables à de nombreuses attaques même lorsqu'ils sont situés derrière un pare-feu sécurisé et bien configuré.

Un firewall qui protège un serveur web doit être configuré pour bloquer tous les accès au serveur web de l'Internet, sauf les ports nécessaires, tels que les ports TCP 80 (HTTP) et 443 (HTTPS).

Afin de protéger avec succès un serveur web en utilisant un pare-feu, assurez-vous que le firewall est mis à jour avec les derniers correctifs stables et est configuré pour effectuer les opérations suivantes :

- Contrôler tout le trafic entre l'Internet et le serveur web
- Bloquer tout trafic entrant au serveur web à l'exception du trafic qui est requis, tels que les ports TCP 80 (HTTP) et / ou 443 (HTTPS)
- Bloquer tout le trafic entrant avec une adresse IP interne (pour éviter les attaques de type IP spoofing)
- Bloquer les connexions client à partir du serveur web pour l'Internet et le réseau interne de l'entreprise
- Bloquer les adresses IP que l'IDS/IPS reporte en tant que des adresses d'attaques

- Avertissez l'administrateur du serveur web des activités suspectes par un moyen approprié (par exemple, e-mail, sms, etc.)
- Fournir le filtrage de contenu
- Protéger contre les attaques DoS
- Détecter les requêtes malformées ou les URL d'attaques connus
- Enregistrer les événements critiques

7.3.2. Systèmes de détection et de prévention d'intrusions (IDS/IPS)

Afin de protéger avec succès un serveur web en utilisant un IDS/IPS, il fallait s'assurer qu'il est configuré pour :

- Surveiller le trafic réseau depuis et vers le serveur web
- Surveiller les changements apportés aux fichiers importants sur le serveur web
- Surveiller les ressources systèmes disponibles sur le serveur web
- Bloquer (en conjonction avec le pare-feu) les adresses IP ou les sous-réseaux qui attaquent le réseau de l'entreprise
- Aviser les parties appropriées (par exemple, administrateur IDS/IPS, administrateur de serveur web, l'équipe de réponse aux incidents) des attaques soupçonnées par des moyens appropriés conformément à la politique et aux procédures de réponse aux incidents de l'entreprise
- Détecter la plus large variété des scans et des attaques avec un niveau acceptable de faux positifs
- Enregistrer les événements du journal, y compris les détails suivants:
 - Heure / date
 - Adresse IP du sensor
 - Nom de l'événement
 - type d'attaque (s'il en existe un)
 - adresses IP source et destination
 - numéro de port source et destination
 - protocole réseau
- Pour les événements de réseau, capturer l'information d'en-tête de paquet pour aider à l'analyse et au processus d'investigation
- Mise à jour fréquente avec de nouvelles signatures d'attaque (par exemple, sur une base quotidienne à hebdomadaire, généralement après avoir testé les mises à jour)

7.3.3. Commutateurs réseau

Les commutateurs doivent être configurés afin de protéger contre les écoutes réseau et vaincre l'usurpation ARP.

7.3.4. Répartiteurs de charge (Load balancers)

Les répartiteurs de charge sont utilisés pour augmenter la disponibilité du serveur web et sont complétés par les caches web (s'il ya lieu)

7.3.5. Reverse Proxy

Un reverse proxy est utilisé comme une passerelle de sécurité pour accroître la disponibilité du serveur web. Il inclue l'ensemble ou certaines des fonctionnalités suivantes:

- Accélérateur de chiffrement qui déchargent le traitement des calculs coûteux requis pour initier des connexions SSL / TLS
- Passerelle de sécurité qui surveille le trafic HTTP depuis et vers le serveur web pour les attaques potentielles et prendre les mesures nécessaires, agissant ainsi comme pare-feu niveau applicatif
- Le filtre de contenu qui peut contrôler le trafic depuis et vers le serveur web pour les données potentiellement sensibles ou inappropriées et prendre les mesures nécessaires
- Passerelle d'authentification qui authentifie les utilisateurs via une variété de mécanismes et contrôle l'accès aux URL hébergées sur le serveur web lui-même

7.4. Check-list pour mettre en place une infrastructure réseau sécurisée

Complété	Action
	Identifier l'emplacement dans le réseau
<input type="checkbox"/>	Serveur web est situé dans une DMZ
	Évaluer la configuration du firewall
<input type="checkbox"/>	Serveur web est protégé par un pare-feu de couche d'application
<input type="checkbox"/>	Firewall contrôle tout le trafic entre l'Internet et le serveur web
<input type="checkbox"/>	Pare-feu bloque tout le trafic entrant vers le serveur web, sauf les ports TCP 80 (HTTP) et / ou 443 (HTTPS)

<input type="checkbox"/>	Pare-feu bloque les adresses IP que l'IDS/IPS reporte en tant que des adresses d'attaque
<input type="checkbox"/>	Pare-feu réseau notifie l'administrateur du serveur web des activités suspectes par un moyen approprié
<input type="checkbox"/>	Pare-feu offre le filtrage de contenu (pare-feu de couche d'application)
<input type="checkbox"/>	Pare-feu est configuré pour se protéger contre les attaques DoS
<input type="checkbox"/>	Firewall détecte les requêtes mal formés ou les URL d'attaque connus
<input type="checkbox"/>	Firewall journalise (logue) les événements critiques
<input type="checkbox"/>	Le firewall est mis à jour avec les derniers correctifs stables
	Évaluer les systèmes de détection et de prévention d'intrusion
<input type="checkbox"/>	IDS/IPS est configuré pour surveiller le trafic réseau depuis et vers le serveur web
<input type="checkbox"/>	IDS/IPS est configuré pour surveiller les changements apportés aux fichiers importants sur le serveur web (IDS/IPS hôte ou contrôleur d'intégrité de fichiers)
<input type="checkbox"/>	IDS/IPS bloque (en conjonction avec le firewall) les adresses IP ou les sous-réseaux qui attaquent le réseau de l'entreprise
<input type="checkbox"/>	IDS/IPS avise l'administrateur du serveur web des attaques soupçonnées par des moyens appropriés
<input type="checkbox"/>	IDS/IPS est configuré de manière à maximiser la détection avec un niveau acceptable de faux positifs
<input type="checkbox"/>	IDS/IPS est configuré pour enregistrer les événements du journal
<input type="checkbox"/>	IDS/IPS est mis à jour fréquemment avec de nouvelles signatures d'attaque (par exemple, sur une base quotidienne)
<input type="checkbox"/>	IDS/IPS hôte est configuré pour surveiller les ressources système disponibles au niveau du serveur web
	Évaluer les commutateurs réseau
<input type="checkbox"/>	Les commutateurs sont utilisés pour protéger contre les écoutes réseau
<input type="checkbox"/>	Les commutateurs sont configurés en mode haute sécurité afin de vaincre les attaques ARP poisoning
<input type="checkbox"/>	Les commutateurs sont configurés pour envoyer tout le trafic sur le segment de réseau vers l'IDS/IPS réseau
	Évaluer les répartiteurs de charge (Load balancers)

<input type="checkbox"/>	Les répartiteurs de charge sont utilisés pour augmenter la disponibilité du serveur web
<input type="checkbox"/>	Les équilibreurs de charge sont complétés par les caches web
	Evaluer le reverse proxy
<input type="checkbox"/>	Le reverse proxy est utilisé comme une passerelle de sécurité pour accroître la disponibilité du serveur web
<input type="checkbox"/>	Le reverse proxy est complété par une accélération de chiffrement, une authentification des utilisateurs et des fonctionnalités de filtrage de contenu

8. ADMINISTRATION SECURISEE DU SERVEUR WEB

Cette section fournit des recommandations générales pour l'administration sécurisée des serveurs web. Les activités essentielles incluent l'analyse des fichiers journaux, les sauvegardes régulières, le contrôle régulier de la sécurité du serveur web, l'administration à distance sécurisée, la supervision et la gestion des incidents.

8.1. Journalisation

Pour une gestion efficace du serveur web, il est nécessaire de disposer d'un retour d'informations (feedback) à propos de l'activité et des performances du serveur, ainsi que de tout problème qui pourrait survenir. Pour cela, une journalisation souple et complète s'avère primordiale. Par conséquent, le suivi et l'analyse des logs sont des activités essentielles pour se rendre compte des comportements suspects.

Dans ce cadre, les points suivants doivent être considérés :

- Utiliser un serveur Syslog centralisé
- Avoir des mécanismes d'alerte (mail, SMS) pour avertir l'administrateur en cas d'actes malveillants détectés dans les logs
- Utiliser un format de journal bien approprié (tel que Combined Log format)
- Mettre en place des noms différents des fichiers journaux pour les différents sites web virtuels qui peuvent être déployés sur un seul serveur web physique
- S'assurer que des procédures sont en place pour que les fichiers log ne remplissent pas le disque dur
- S'assurer que les fichiers journaux sont régulièrement archivés, sécurisés et analysés

8.2. Sauvegarde du serveur web

- Une politique de sauvegarde du contenu de la plateforme web (code source, fichiers de configuration, base de données) devrait être appliquée et une sauvegarde régulière des fichiers devrait être assurée

- Maintenir la copie la plus récente de contenu de site web sur un hôte sécurisé ou sur des médias
- Maintenir la vérification de l'intégrité de tous les fichiers importants dans le système. Cela peut être fait par génération des tables de hachage MD5 des fichiers importants ou en utilisant des logiciels de vérification d'intégrité (tel que Tripwire)

8.3. Audit périodique de la plateforme web

L'audit périodique de la sécurité de la plateforme web est fortement recommandé. Sans ces tests périodiques, il n'y a aucune assurance que les mesures de sécurité sont mises en place de façon approprié. Plusieurs techniques de test de la sécurité existent (scan de vulnérabilités, test de pénétration, audit de code source)

8.3.1. Scan de vulnérabilités

Le scan de vulnérabilités permet d'identifier les vulnérabilités et de vérifier si les mesures de sécurité déjà mises en place sont efficaces.

Plusieurs scanners de vulnérabilités (commerciales et open sources) existent. La plupart de ces scanners donnent une classification de la criticité et de l'impact des vulnérabilités découvertes ainsi que les mesures nécessaires à mettre en place.

Voici les points relevés par un scanner de vulnérabilités :

- Identification des machines/serveurs actifs dans un réseau
- Identification des services (ports) actifs sur les machines/serveurs
- Identification des applications installées.
- Identification des systèmes d'exploitation
- Identification des vulnérabilités associées aux systèmes d'exploitation, applications installées et services actifs
- Test de conformité par rapport aux bonnes pratiques de configuration et d'utilisation de la plateforme web et par rapport à la politique de sécurité.

Il est recommandé de mener un scan de vulnérabilités afin de s'assurer que le système d'exploitation et le serveur web sont mis à jour.

Les résultats de scan de vulnérabilité devraient être documentés et analysés et les défaillances constatées doivent être corrigés.

L'utilisation de plusieurs scanners de vulnérabilités est recommandée puisqu'aucun scanner n'est capable de détecter toutes les vulnérabilités connues.

8.3.2. Test de pénétration

Les tests de pénétration sont les tests de sécurité où les évaluateurs tentent de contourner les dispositifs de sécurité d'un système en se basant sur la compréhension de la conception du système et des applications.

Le but des tests de pénétration est d'essayer de contourner les mécanismes de protection mis en place en utilisant des outils et des techniques propriétaires ou développés par des pirates. Ces tests sont fortement recommandés pour les systèmes complexes ou critiques.

Il est à noter que suite aux différents tests de pénétration, le temps de réponse des serveurs se ralentisse et le système peut être endommagé. Ce risque peut être minimisé en faisant appel à des pentesteurs de haut niveau d'expérience.

Les avantages du test de pénétration sont les suivants :

- Tester le réseau en utilisant les méthodologies et les outils employés par les hackers.
- Vérifier si des vulnérabilités existent
- Montrer comment les vulnérabilités identifiées peuvent être exploitées itérativement pour obtenir un accès
- Tester les procédures et sensibiliser le personnel de l'entreprise face aux attaques de type « ingénierie sociale »

8.3.3. Audit de code source

L'audit de code source est l'analyse détaillée du code source des différents modules de l'application. En effet, par méconnaissance des risques ou par malveillance, les développeurs peuvent introduire des vulnérabilités dans les applications qu'ils développent. En outre, sous contraintes des délais à respecter et suite à la complexité de l'application, plusieurs failles peuvent être induites lors du développement. Ces vulnérabilités peuvent avoir des conséquences néfastes sur la plateforme.

Il est alors fortement recommandé de mener un audit de code source afin d'identifier les vulnérabilités de niveau applicatif qui n'ont pas pu être détectées au niveau des scans de vulnérabilités et des tests de pénétration.

8.4. Supervision

La supervision du serveur (ou monitoring serveur) permet de surveiller les différents composants propres de la machine et le fonctionnement des applicatifs qu'elle héberge.

Principalement, trois types de supervision existent.

8.4.1. Supervision système

La supervision système porte principalement sur les trois ressources système suivants:

- i. Processeur
- ii. Mémoire
- iii. Stockage

8.4.2. Supervision réseau

La supervision réseau assure les principales fonctionnalités suivantes :

- Vue d'ensemble du réseau
- Surveillance des services réseaux (SMTP, POP3, HTTP, NNTP, PING, etc.)
- Surveillance des ressources des équipements réseaux (charge processeur, mémoires, etc.)
- Surveillance de la disponibilité des services en ligne
- Surveillance des débits
- Suivi les flux en temps-réel
- Suivi et remontée d'alertes

Quelques outils permettent de réaliser ces fonctionnalités tels que : NetCrunch 5, Checklan Monitor, Nagios, Centreon, ACGVision, MRTG, Cacti, Zabbix, Visual I/O, Iperf, Nagios2Cacti, Eyesofnetwork et Opennms.

8.4.3. Supervision des applications

La supervision des applications (ou supervision applicative) permet de connaître la disponibilité des machines en terme de services rendus et ceci en testant les applications hébergées par les serveurs.

A titre d'exemple, un serveur web peut avoir une supervision système et réseau avec des signaux au vert et pourtant la machine n'est pas disponible au sens du service web si apache n'est pas démarré.

8.5. Gestion des incidents

Un incident de sécurité informatique peut être un acte de violation de la politique de sécurité qui peut être traduit par un accès non autorisé, un déni de service ou une perturbation du système, etc.

Une équipe de réponse aux incidents doit être créée au sein de l'entreprise et à laquelle doivent être signalés les incidents. Si ce n'est pas possible par manque de ressources (compétences techniques, matériels et logiciels pour le traitement des incidents), il est recommandé de remonter l'information à l'Agence Nationale de la Sécurité Informatique ANSI (incident@ansi.tn)

Voici comment réagir en cas d'incident :

- **Identification/Qualification de l'incident**
 - Analyser les anomalies signalées
 - Rechercher d'autres traces suspectes
 - Confirmer l'incident

- **Limitation de l'éventuelle extension de l'incident**
 - Isoler les machines infectées ou protéger les machines saines
 - Retirer ou sauvegarder certaines données critiques
 - Passer en mode de crise si la gravité est élevée

- **Investigation**
 - Identifier les scénarios de l'incident
 - Analyser les fichiers logs pertinents
 - Comprendre la nature du problème

- **Retour à la Normale**
 - Renforcer le niveau de sécurité (niveau de patch, fermeture de ports inutiles, etc.)
 - Restaurer les données et les applications affectées à partir des sauvegardes

- **Tirer les leçons**
 - Identifier les améliorations à apporter au système d'information
 - Maintenir une surveillance sur le système qui a été affecté
 - Rédiger un rapport global décrivant l'incident

8.6. Check-list pour la gestion du serveur web

Complété	Action
	Effectuer la journalisation (logging)
<input type="checkbox"/>	Utiliser un serveur Syslog centralisé
<input type="checkbox"/>	Avoir des mécanismes d'alerte (mail, SMS) pour avertir l'administrateur en cas d'actes malveillants détectés dans les logs
<input type="checkbox"/>	Utiliser un format de journal bien approprié (tel que Combined Log format)

<input type="checkbox"/>	Mettre en place des noms différents des fichiers journaux pour les différents sites web virtuels qui peuvent être déployés sur un seul serveur web physique
<input type="checkbox"/>	S'assurer que des procédures sont en place pour que les fichiers log ne remplissent pas le disque dur
<input type="checkbox"/>	S'assurer que les fichiers journaux sont régulièrement archivés, sécurisés et analysés
	Effectuer des sauvegardes du serveur web
<input type="checkbox"/>	Une politique de sauvegarde du contenu de la plateforme web (code source, fichiers de configuration, base de données) devrait être appliquée et une sauvegarde régulière des fichiers devrait être assurée
<input type="checkbox"/>	Maintenir la copie la plus récente de contenu de site web sur un hôte sécurisé ou sur des médias
<input type="checkbox"/>	Maintenir la vérification de l'intégrité de tous les fichiers importants dans le système. Cela peut être fait par génération des tables de hachage MD5 des fichiers importants ou en utilisant des logiciels de vérification d'intégrité tel que Tripwire
	Audit périodique de la plateforme web
<input type="checkbox"/>	Mener périodiquement des scans de vulnérabilités sur le système d'exploitation, le serveur web, le contenu dynamiquement généré et sur le réseau supporté
<input type="checkbox"/>	Mise à jour du scanner de vulnérabilité avant utilisation
<input type="checkbox"/>	Corriger les failles identifiées par le scanner de vulnérabilité
<input type="checkbox"/>	Effectuer des tests de pénétration sur le serveur web et l'infrastructure du réseau supporté
<input type="checkbox"/>	Corriger les vulnérabilités identifiées par les tests de pénétration
	Conduite d'administration à distance et mises à jour du contenu
<input type="checkbox"/>	Utiliser un mécanisme d'authentification forte
<input type="checkbox"/>	Restreindre les hôtes, qui peuvent administrer à distance ou qui mettent à jour le contenu du serveur web, par adresse IP et au réseau interne
<input type="checkbox"/>	Ne pas autoriser l'administration à distance à partir d'Internet à moins que des mécanismes tels que les VPN sont utilisés
<input type="checkbox"/>	Utiliser des protocoles sécurisés (par exemple, SSH, HTTPS)
<input type="checkbox"/>	Faire appliquer le concept du moindre privilège pour l'administration à distance et pour la mise à jour de contenu (par exemple, tenter de minimiser les droits d'accès pour l'administration à distance / mise à jour des

	comptes)
<input type="checkbox"/>	Changer tous les comptes et les mots de passe par défaut à partir de l'utilitaire d'administration à distance ou par l'application elle-même
<input type="checkbox"/>	Ne monter pas de partages de fichiers sur le réseau interne à partir du serveur web ou vice-versa
	Supervision
<input type="checkbox"/>	Supervision système
<input type="checkbox"/>	Supervision réseau
<input type="checkbox"/>	Supervision des applications
	Gestion d'incident
<input type="checkbox"/>	Une équipe de réponse aux incidents crée au sein de l'entreprise
<input type="checkbox"/>	Identification/Qualification de l'incident <ul style="list-style-type: none"> • Analyser les anomalies signalées • Rechercher d'autres traces suspectes • Confirmer l'incident
<input type="checkbox"/>	Limitation de l'éventuelle extension de l'incident <ul style="list-style-type: none"> • Isoler les machines infectées ou protéger les machines saines • Retirer ou sauvegarder certaines données critiques • Passer en mode de crise si la gravité est élevée
<input type="checkbox"/>	Investigation <ul style="list-style-type: none"> • Identifier les scénarios de l'incident • Analyser les fichiers logs pertinents • Comprendre la nature du problème
<input type="checkbox"/>	Retour à la Normale <ul style="list-style-type: none"> • Renforcer le niveau de sécurité (niveau de patch, fermeture de ports inutiles, etc.) • Restaurer les données et les applications affectées à partir des sauvegardes
<input type="checkbox"/>	Tirer les leçons <ul style="list-style-type: none"> • Identifier les améliorations à apporter au système d'information • Maintenir une surveillance sur le système qui a été affecté • Rédiger un rapport global décrivant l'incident

9. BIBLIOGRAPHIE

- <http://csrc.nist.gov/publications/nistpubs/800-44-ver2/SP800-44v2.pdf>
- <http://www.cert-in.org.in/knowledgebase/guidelines/cisg-2004-04.htm>

10. ANNEXES

* Liste de quelques bases de données de vulnérabilités

Secunia	http://secunia.com/
Vupen	http://www.vupen.com/?fr
CVE-MITRE	http://cve.mitre.org/ http://cme.mitre.org/
Cert-IST	http://www.cert-ist.com/
OSVDB	http://osvdb.org/
NIST	http://csrc.nist.gov/
FIRST	http://www.first.org/
Securityfocus	http://www.securityfocus.com/
Securitytracker	http://www.securitytracker.com/
SecurityVulns	http://securityvulns.com/
SecurityTeam	http://www.securiteam.com/
Milw0rm	http://www.milw0rm.com/
Sebug	http://www.sebug.net/
US-CERT	http://www.us-cert.gov/
CERTA	http://www.certa.ssi.gouv.fr/

** OWASP Top 10 Application Security Risks –2010

http://www.owasp.org/index.php/File:OWASP_T10_-_2010_rc1.pdf

*** CWE/SANS TOP 25 Most Dangerous Programming Errors

<http://www.sans.org/top25-programming-errors/>

<http://cwe.mitre.org/top25/>