Notion de risques : Chap.5.a : Sécurité des Infrastructures Réseau : Part.1

mardi 20 février 2024 15:01

Chap.5.b : Sécurité des Infrastructures Réseau

A. Composants Clés des Infrastructures Réseau
B. Menaces et Défis de Sécurité des Infrastructures
C. Stratégies et Solutions de Sécurisation
D. Études de Cas et Scénarios Pratiques (Protocole à filtre)
    a. TP : Mise en place d'une architecture de type 2 (séparation des fonctions applicatives)
E. Fondamentaux des Réseaux Informatiques (Architecture N3)
F. Menaces et Vulnérabilités Réseau (Architecture N3)
G. Stratégies de Sécurisation des Réseaux (Architecture N3)
H. Technologies et Outils de Sécurité Réseau
I. Technologies et Outils de Sécurité Réseau spécifiques à l'architecture N3
    a. TP : Mise en place d'une architecture de type 3 (ajout d'équipement réseau, filtrage)

A. Composants Clés des Infrastructures Réseau

Présentation de l'Architecture Réseau à Deux Niveaux

Dans une architecture à deux niveaux, les serveurs Front-End et Back-End jouent des rôles distincts, chacun ayant des interactions spécifiques avec les flux réseau et les protocoles utilisés.

Rôles et Interactions des Composants

1. Front-End (Serveurs Web avec Apache)

- Les serveurs Front-End, souvent basés sur Apache, jouent un rôle crucial dans la gestion du trafic entrant en provenance des utilisateurs finaux via les protocoles HTTP/HTTPS.

- Pour la gestion de contenu dynamique, les langages de script tels que PHP ou Python sont souvent utilisés. Les applications web sont déployées sur ces serveurs via des protocoles de transfert de fichiers tels que **FTP ou SSH**.

2. Back-End (Serveurs de Bases de Données avec MySQL)

- Les serveurs Back-End utilisent MySQL comme système de gestion de base de données pour stocker et gérer les données nécessaires aux applications hébergées sur les serveurs Front-End.

3. Administration du Front-End et du Back-End

- L'administration du Front-End implique généralement l'utilisation d'outils; par défaut sur des protocoles non sécurisés, il faudra bannir ces outils en les remplaçant par des outils s'appuyant sur des protocoles sécurisés.
- L'administration du Back-End implique généralement l'utilisation des mêmes outils que le Front-End ainsi que des outils spécifiques pour la gestion de la base de donnée, tels que PhpMyAdmin pour importer, exporter, modifier le schéma et gérer les tables de la base de données (MySQL ou MariaDB).

Flux Réseau et Protocoles Utilisés

1. Front-End/Back-End

- Les flux réseau entrants vers le Front-End sont gérés via les protocoles **HTTP/HTTPS**, permettant

aux utilisateurs d'accéder aux ressources web hébergées sur les serveurs Front-End.

- Les échanges entre le Front-End et le Back-End impliquent l'utilisation de protocoles de base de données tels que MySQL pour récupérer et transférer les données entre les deux niveaux.

- Pour assurer la sécurité des communications, **SSL/TLS** peut être utilisé pour chiffrer les données sensibles en transit entre les serveurs Front-End et Back-End.

2. Administration du Front-End et du Back-End

Pour l'administration, plusieurs outils/protocoles non sécurisés tel que **FTP (File Transfer Protocol) et KSH (Korn Shell)** sont utilisés pour le transfert de fichiers et la communication à distance, il conviendra de les bannir par les solutions suivantes :

1. ****SFTP (SSH File Transfer Protocol)****

- SFTP est une extension du protocole SSH (Secure Shell) utilisée pour le transfert de fichiers de manière sécurisée. Contrairement à FTP, SFTP chiffre à la fois les données et les commandes, protégeant contre les écoutes indiscrètes et les interceptions de données.

2. ****FTPS (FTP Secure)****

- FTPS est une version de FTP qui ajoute le support du chiffrement SSL/TLS. Il existe deux variantes principales : FTPS explicite (le client doit demander explicitement la sécurisation) et FTPS implicite (la connexion est sécurisée dès le début).

3. ****SCP (Secure Copy Protocol)****

- SCP est un protocole basé sur SSH utilisé pour le transfert sécurisé de fichiers entre des hôtes sur un réseau. Il utilise le même mécanisme d'authentification et de chiffrement que SSH.

4. ****WebDAV sur HTTPS****

- WebDAV (Web Distributed Authoring and Versioning) est un protocole pour la gestion de fichiers sur des serveurs web. Lorsqu'il est utilisé sur HTTPS, il offre un transfert de fichiers sécurisé.

5. ****SSH (Secure Shell) pour les connexions à distance****

- SSH remplace les protocoles de communication à distance moins sécurisés comme **Telnet et KSH**. Il fournit un canal sécurisé sur un réseau non sécurisé en utilisant un chiffrement fort.

6. ****rsync sur SSH****

- rsync est un outil de synchronisation de fichiers qui peut être utilisé sur SSH pour le transfert sécurisé de fichiers. Il est efficace pour la synchronisation de répertoires et la minimisation du transfert de données.

B. Menaces et Défis de Sécurité des Infrastructures

Identification des Menaces et Vulnérabilités

Dans une architecture à deux niveaux, plusieurs menaces et vulnérabilités spécifiques peuvent compromettre la sécurité du système :

1. Attaques sur les Serveurs Front-End :

- Les serveurs Front-End sont directement exposés à Internet et sont donc vulnérables à un large éventail d'attaques, telles que les injections SQL, les attaques par déni de service distribué (DDoS) et les attaques de script entre sites (XSS).

- Les applications web hébergées sur les serveurs Front-End sont souvent la cible d'attaques visant à exploiter des vulnérabilités de sécurité connues pour compromettre les données sensibles ou accéder illégalement au système.

2. Risques de Communication entre Front-End et Back-End

- Les échanges de données entre les serveurs Front-End et Back-End sont essentiels au bon fonctionnement de l'architecture à deux niveaux, mais ils présentent également des risques de sécurité.

- Les attaques de type interception de données peuvent compromettre la confidentialité des données en transit, tandis que les attaques de type altération de données peuvent modifier les informations échangées entre les serveurs Front-End et Back-End.

Analyse des Risques Potentiels et des Défis de Sécurité

La séparation des fonctions Front-End et Back-End dans une architecture à deux niveaux présente plusieurs défis de sécurité, notamment :

1. Gestion des Accès et des Permissions :

- Assurer un contrôle d'accès approprié pour limiter l'accès aux ressources sensibles et garantir que seuls les utilisateurs autorisés peuvent interagir avec les serveurs Front-End et Back-End.

- La gestion des identités et des permissions devient complexe dans un environnement où de multiples utilisateurs et applications doivent interagir avec les serveurs Front-End et Back-End de manière sécurisée.

2. Intégrité et Confidentialité des Données :

- Protéger l'intégrité et la confidentialité des données échangées entre les serveurs Front-End et Back-End en utilisant des protocoles de communication sécurisés tels que SSL/TLS pour chiffrer les données sensibles en transit.

- Veiller à ce que les données stockées sur les serveurs Back-End soient correctement sécurisées et protégées contre les accès non autorisés ou les manipulations malveillantes.

Techniques d'attaque et de défense

Les techniques d'attaques : les injections SQL, les attaques par déni de service distribué (DDoS) et les attaques de script entre sites (XSS).

Les solutions : Introduction aux solutions technologiques telles que les pare-feu, les IDS/IPS et les VPN pour renforcer la sécurité de l'infrastructure (voir ci-après)

C. Stratégies et Solutions de Sécurisation

Point d'étape sur notre architecture N1

Meilleures Pratiques pour Protéger les Architectures à Deux Niveaux

Dans une architecture à deux niveaux, plusieurs stratégies et solutions de sécurité peuvent être mises en œuvre pour renforcer la protection du système :

1. Sécurisation des Serveurs Front-End :

- Renforcer la sécurité des serveurs Front-End en appliquant les bonnes pratiques de configuration et de gestion des serveurs web, telles que la mise à jour régulière des logiciels, la configuration sécurisée des paramètres du serveur et la désactivation des fonctionnalités non utilisées.

- Mettre en œuvre des solutions de protection contre les attaques web, telles que les pare-feu d'application web (WAF) et les systèmes de détection et de prévention des intrusions (IDS/IPS), pour détecter et bloquer les tentatives d'exploitation de vulnérabilités connues.

2. Sécurisation des Communications Front-End et Back-End :

- Chiffrer les communications entre les serveurs Front-End et Back-End en utilisant des protocoles sécurisés tels que SSL/TLS pour garantir la confidentialité et l'intégrité des données échangées.

- Mettre en place des réseaux privés virtuels (VPN) pour établir des tunnels sécurisés entre les serveurs Front-End et Back-End, limitant ainsi l'accès aux communications uniquement aux parties autorisées.

Processus de Sécurisation pour les Informaticiens dans les 3 Architectures

1. Architecture N1 (Développement)

- **Accès au Réseau de Développement :**
  - Les développeurs doivent être autorisés à accéder au réseau de développement uniquement via des identifiants sécurisés.
  - L'accès au serveur de développement doit être limité aux seuls informaticiens autorisés, avec des connexions SSH sécurisées pour les transferts de fichiers et l'administration.

- **Sécurisation des Outils de Développement :**
  - Les outils de développement tels que Visual Studio doivent être régulièrement mis à jour avec les derniers correctifs de sécurité.
  - Les environnements de développement intégrés (IDE) doivent être configurés avec des paramètres de sécurité appropriés pour prévenir les vulnérabilités.

- **Processus de Développement Sécurisé :**
  - Les développeurs doivent suivre des pratiques de développement sécurisé, notamment en échappant les entrées utilisateur, en validant les données et en évitant les vulnérabilités courantes telles que les injections SQL et XSS.
  - Des analyses de sécurité du code doivent être effectuées régulièrement pour identifier et corriger les failles de sécurité potentielles.

2. Architecture N2 (Intégration)

- **Accès au Réseau d'Intégration :**
  - Les administrateurs système et les développeurs doivent être autorisés à accéder au réseau d'intégration via des identifiants sécurisés et des connexions VPN si nécessaire.
  - Les accès doivent être strictement contrôlés et limités aux personnes nécessaires à l'intégration et aux tests.

- **Sécurisation des Outils d'Intégration :**
  - Les outils d'intégration continue (CI) et de gestion de versions comme Git doivent être configurés avec des contrôles d'accès appropriés pour limiter l'accès au code source et aux artefacts d'intégration.
  - Des mécanismes de surveillance doivent être mis en place pour détecter toute activité suspecte ou non autorisée sur les outils d'intégration.

- **Processus d'Intégration Sécurisée :**
  - Les environnements d'intégration doivent être isolés du reste du réseau pour minimiser les risques de compromission.
  - Des tests de sécurité automatisés doivent être effectués à chaque intégration pour identifier les problèmes de sécurité potentiels dès que possible.

3. Architecture N3 (Production)

- **Accès au Réseau de Production :**
  - L'accès au réseau de production doit être strictement restreint aux administrateurs système autorisés et au personnel de support technique via des connexions sécurisées et des VLANs dédiés.
  - Les accès doivent être surveillés et audités régulièrement pour détecter toute activité suspecte.

- **Sécurisation des Outils de Production :**
  - Les outils de surveillance et de gestion des configurations doivent être configurés avec des niveaux appropriés d'accès et de contrôle pour protéger les systèmes de production contre les modifications non autorisées.
  - Des procédures de sauvegarde régulières doivent être mises en place pour assurer la disponibilité des données et des applications en cas de sinistre.

- **Processus de Gestion de la Production :**
  - Les mises à jour de sécurité doivent être appliquées régulièrement sur tous les systèmes de production pour corriger les vulnérabilités connues.
  - Des tests de pénétration et des audits de sécurité doivent être effectués périodiquement pour évaluer l'efficacité des mesures de sécurité en place et identifier les éventuelles lacunes.

D. Études de Cas et Scénarios Pratiques (Protocole à filtre)

Analyse d'études de cas réels

- TP : Mise en place d'une architecture de type 2 (séparation des fonctions applicatives)
  - Permet de mettre en place notre architecture d'intégration
  - Le TP sert à montrer la problématique du manque de filtrage au niveau des protocoles

E. Fondamentaux des Réseaux Informatiques (Architecture N3)

Introduction aux Solutions Technologiques (Architecture N3)

- Les pare-feu sont des composants essentiels de toute infrastructure réseau, filtrant le trafic entrant et sortant pour prévenir les attaques et limiter l'accès non autorisé aux ressources.

- Les systèmes de détection et de prévention des intrusions (IDS/IPS) surveillent le trafic réseau à la recherche de comportements suspects et peuvent prendre des mesures pour bloquer les activités malveillantes.

- Les réseaux privés virtuels (VPN) offrent un moyen sécurisé de connecter des réseaux distants via Internet, permettant aux données de circuler de manière confidentielle et sécurisée.

1. Fondamentaux de la Sécurité des Réseaux

Concepts de base

1. **Pare-feu (Firewall) :** Dispositif de sécurité filtrant le trafic réseau pour empêcher les accès non autorisés et protéger les ressources internes contre les attaques externes. Dans l'architecture proposée, un firewall sera ajouté en Front-End pour contrôler et sécuriser le trafic entrant vers les serveurs web.

2. **IDS/IPS (Intrusion Detection System / Intrusion Prevention System) :** Systèmes de détection et de prévention des intrusions surveillant le trafic pour détecter et bloquer les activités malveillantes sur le réseau. Ils seront essentiels pour surveiller et protéger les serveurs Front-End et

Back-End contre les intrusions et les attaques.

3. **VPN (Virtual Private Network) :** Réseau privé virtuel créant un tunnel sécurisé pour les communications sur des réseaux publics, assurant la confidentialité et l'intégrité des données. Un VPN peut être utilisé pour sécuriser les communications entre les différents VLANs, y compris pour l'administration des serveurs.

Protocoles de sécurité réseau

1. **SSL/TLS (Secure Sockets Layer / Transport Layer Security) :** Protocoles de cryptage sécurisant les communications sur Internet en fournissant confidentialité, intégrité et authentification des données échangées. Ils seront utilisés pour chiffrer les données sensibles lors des communications entre les serveurs Front-End et Back-End, ainsi qu'avec les clients.

2. **IPSec (Internet Protocol Security) :** Ensemble de protocoles sécurisant les communications IP en établissant des tunnels VPN et en garantissant la confidentialité et l'intégrité des données. IPSec peut être utilisé pour sécuriser les communications entre les différents VLANs de l'architecture proposée, renforçant ainsi la sécurité du réseau.

F. Menaces et Vulnérabilités Réseau (Architecture N3)

Identification des Menaces et Vulnérabilités

Dans le contexte de l'architecture proposée avec l'ajout d'un firewall en Front-End et la configuration de VLANs pour les communications et l'administration, plusieurs menaces et vulnérabilités doivent être prises en compte :

1. **Attaques sur le Front-End :**

Les serveurs Front-End, désormais exposés à Internet, sont susceptibles d'être la cible d'attaques telles que les injections SQL, les attaques par déni de service (DDoS) et les attaques de cross-site scripting (XSS), compromettant ainsi la sécurité et la disponibilité des services web.

L'administration devra se faire à partir d'un réseau Privé.

2. **Risques de Communication Non Sécurisée :**

Malgré la segmentation des VLANs, les communications entre le Front-End et le Back-End peuvent être exposées à des interceptions de données si elles ne sont pas sécurisées à l'aide de protocoles comme SSL/TLS. Cela peut entraîner la compromission des données sensibles échangées entre les deux niveaux d'infrastructure.

Les flux Front-End et le Back-End devront être segmentés, sécurisés ou les deux à la fois.

3. **Vulnérabilités du Firewall :**

Même si le firewall en Front-End est destiné à protéger le réseau contre les attaques externes, il peut lui-même présenter des vulnérabilités, telles que des failles de configuration ou des bugs logiciels, qui pourraient être exploitées par des attaquants pour contourner les mesures de sécurité mises en place.

L'administration devra se faire à partir d'un réseau Privé.

Analyse des Risques Potentiels

En raison de ces menaces et vulnérabilités, plusieurs risques potentiels doivent être pris en compte lors de la conception et de la gestion de l'architecture réseau :

- Risque de Perte de Données Sensibles : Les attaques réussies sur le Front-End ou les communications non sécurisées entre le Front-End et le Back-End peuvent entraîner la divulgation ou la compromission de données sensibles stockées ou transitant sur le réseau.

- Risque de Perturbation du Service : Les attaques DDoS sur le Front-End ou les failles de sécurité du firewall peuvent entraîner une interruption des services web, impactant ainsi la disponibilité des applications et des services pour les utilisateurs finaux.

- Risque de Perte de Contrôle de l'administration : Les vulnérabilités du firewall ou des équipements d'administration peuvent conduire à une compromission de l'accès administratif, permettant aux attaquants de prendre le contrôle du réseau ou d'effectuer des modifications non autorisées.

G. Stratégies de Sécurisation des Réseaux

Suggestions d'Améliorations pour Chaque Domaine de Gestion :

Front-End (Serveurs Web)

- Mettre en place une solution de protection des applications web (WAF) comme **ModSecurity** pour détecter et bloquer les attaques web.
- Utiliser des certificats SSL/TLS à jour et des algorithmes de chiffrement robustes pour sécuriser les communications avec les utilisateurs.

Front-End/Back-End Communication

- Implémenter un contrôle d'accès strict entre les VLANs Front-End et Back-End en utilisant des pare-feu nouvelle génération (NGFW) comme Fortinet **FortiGate**.
- Utiliser des solutions de détection des menaces avancées comme **Check Point SandBlast** pour analyser le trafic entre les deux niveaux et détecter les activités suspectes.

Administration des Serveurs et Équipements

- Renforcer l'accès administratif aux équipements réseau en utilisant des solutions de gestion des identités et des accès (IAM) comme **Okta**.
- Mettre en place une surveillance continue des VLANs d'administration à l'aide de solutions de gestion des événements et des informations de sécurité (SIEM) comme **Splunk**.

H. Technologies et Outils de Sécurité Réseau

Pare-feu Nouvelle Génération (NGFW)

- **Description :** Les pare-feu nouvelle génération (NGFW) combinent les fonctionnalités des pare-feu traditionnels avec des capacités avancées de filtrage de contenu, d'inspection SSL/TLS et de prévention des intrusions.

- **Solutions Recommandées :** Palo Alto Networks Next-Generation Firewall, Fortinet FortiGate, Cisco Firepower.

Systèmes de Détection et de Prévention des Intrusions (IDS/IPS)

- **Description :** Les IDS/IPS surveillent le trafic réseau à la recherche de comportements suspects

ou de signatures d'attaques connues, et peuvent soit détecter les incidents soit les bloquer automatiquement.

- **Solutions Recommandées :** Snort, Suricata, Cisco Intrusion Prevention System (IPS).

Protection des Applications Web (WAF)

- **Description :** Les WAF protègent les applications web contre les attaques telles que les injections SQL, les attaques XSS et les attaques de déni de service distribué (DDoS).

- **Solutions Recommandées :** ModSecurity, Imperva SecureSphere, F5 BIG-IP Application Security Manager.

Analyse des Menaces Avancées (ATP)

- **Description :** Les solutions d'analyse des menaces avancées (ATP) utilisent l'intelligence artificielle et l'apprentissage automatique pour détecter les menaces émergentes et les comportements anormaux.

- **Solutions Recommandées :** Cisco Advanced Malware Protection (AMP), FireEye Threat Intelligence Platform, CrowdStrike Falcon.

Gestion des Identités et des Accès (IAM)

- **Description :** Les solutions IAM permettent de gérer les identités et les droits d'accès des utilisateurs et des appareils sur le réseau, garantissant ainsi une authentification et une autorisation sécurisées.

- **Solutions Recommandées :** Okta Identity Cloud, Microsoft Azure Active Directory, Ping Identity.

Gestion des Événements et des Informations de Sécurité (SIEM)

- **Description :** Les solutions SIEM collectent, corrélation et analysent les données de sécurité à partir de diverses sources pour détecter les menaces et les incidents de sécurité.

- **Solutions Recommandées :** Splunk Enterprise Security, IBM QRadar, LogRhythm NextGen SIEM Platform.

I. Technologies et Outils de Sécurité Réseau spécifiques à l'architecture N3

Les dispositifs composant les architectures N1, N2, N3 :

Architecture N1 (Développement)

1. **Serveur Apache** : Le serveur web Apache est utilisé pour héberger et servir des applications web développées en PHP.

2. **PHP** : PHP est un langage de script côté serveur utilisé pour développer des applications web dynamiques. Il s'intègre étroitement avec le serveur Apache pour générer du contenu web dynamique.

3. **MySQL** : MySQL est un système de gestion de base de données relationnelle largement utilisé dans les applications web. Il permet de stocker et de gérer les données utilisées par les applications PHP.

4. **Visual Studio (ou tout autre environnement de développement intégré)** : Visual Studio est un environnement de développement intégré (IDE) utilisé pour écrire, déboguer et tester le code des applications web développées en PHP.

Architecture N2 (Intégration)

1. **Serveur Front-End** : Le serveur Front-End est le point d'entrée des requêtes utilisateur. Il peut s'agir d'un serveur web Apache ou tout autre serveur HTTP.

2. **Serveur Back-End** : Le serveur Back-End héberge les bases de données et les applications nécessaires au fonctionnement de l'application web. Il peut utiliser MySQL ou tout autre système de gestion de base de données approprié.

L'architecture N2 étend l'architecture N1 en ajoutant des environnements et des outils pour tester et intégrer les applications développées.

Architecture N3 (Production)

1. **Serveur Front-End** : Le serveur Front-End est le point d'entrée des requêtes utilisateur. Il peut s'agir d'un serveur web Apache ou tout autre serveur HTTP.

2. **Serveur Back-End** : Le serveur Back-End héberge les bases de données et les applications nécessaires au fonctionnement de l'application web. Il peut utiliser MySQL ou tout autre système de gestion de base de données approprié.

3. **Firewall (pare-feu)** : Le pare-feu est un dispositif de sécurité qui contrôle et filtre le trafic réseau entrant et sortant. Il peut être placé en amont du serveur Front-End pour protéger l'infrastructure contre les attaques et les intrusions.

4. **VLANs (Réseaux Locaux Virtuels)** : Les VLANs sont utilisés pour segmenter le trafic réseau et isoler différents types de données et de services. Ils peuvent être configurés pour séparer les flux de données utilisateur, d'administration et de gestion, améliorant ainsi la sécurité et les performances du réseau.

5. **VPN (Virtual Private Network)** : Un réseau privé virtuel est utilisé pour établir des connexions sécurisées entre différents réseaux ou entre des utilisateurs distants et le réseau interne.

<u>Les réseaux composant l'architecture N3 :</u>

Architecture de Production (N3)

1. Configuration du Pare-feu PfSense

- Le pare-feu PfSense dispose de deux interfaces réseau :
  - Une interface connectée au réseau public (Internet) pour gérer le trafic entrant et sortant.
  - Une interface connectée au VLAN privé VP1 pour contrôler le trafic interne.

2. Configuration des VLANs

- VLAN privé VP1 :
  - Connecté au pare-feu PfSense pour le trafic interne.
  - Connecté au serveur web pour l'hébergement des applications.

- VLAN privé VP2 :
  - Connecté au serveur MySQL pour les échanges avec le serveur web.
  - Séparé du VLAN privé VP1 pour une isolation des services.

- VLAN privé VP_Admin :
  - Connecté au pare-feu PfSense, au serveur web et au serveur MySQL pour l'administration.
  - Permet l'accès sécurisé aux équipements réseau et serveurs pour l'administration système et réseau.

3. Plan d'Adressage IP

- VLAN privé VP1 :
  - Plage d'adresses IP attribuée pour les équipements du VLAN privé VP1.
  - Adresses IP attribuées statiquement aux serveurs web et autres périphériques connectés à ce VLAN.

- VLAN privé VP2 :
  - Plage d'adresses IP réservée pour le VLAN privé VP2.
  - Adresses IP statiquement assignées , au serveur Web, au serveur MySQL et autres périphériques connectés à ce VLAN.

- VLAN privé VP_Admin :
  - Plage d'adresses IP réservée pour l'administration réseau.
  - Adresses IP statiquement assignées aux équipements administratifs connectés à ce VLAN.

<u>Les flux-réseaux composant l'architecture N3 :</u>

VLAN Public (liaison vers Internet)
- **Applications/Services** :
  - Serveur Web (Apache, Nginx)
  - Applications Web accessibles au public

- **Protocoles** :
  - HTTP/HTTPS pour la communication web
  - DNS pour la résolution des noms de domaine
  - SMTP/IMAP/POP pour la messagerie électronique sortante et entrante (le cas échéant)

VLAN Privé VP1 (liaison vers le Serveur Web)
- **Applications/Services** :
  - Serveur Web (Apache)
  - Applications Web internes

- **Protocoles** :
  - HTTP/HTTPS pour la communication web interne

VLAN Privé VP2 (liaison vers le Serveur MySQL)
- **Applications/Services** :
  - Serveur MySQL
  - Applications nécessitant un accès à la base de données

- **Protocoles** :
  - MySQL/MariaDB pour l'accès à la base de données
  - MySQL Workbench ou PhpMyAdmin pour l'administration de la base de données

VLAN Privé VP_Admin
- **Applications/Services** :
    - Outils d'administration réseau et système
    - Interface de gestion du Pare-feu PfSense
    - Outils de développement et d'administration du serveur Web
    - SSH pour l'administration sécurisée du serveur Web

- FTP/SFTP pour le transfert de fichiers vers le serveur Web (si nécessaire)
- Protocoles de sauvegarde et de restauration de base de données (par exemple, mysqldump)*

- **Protocoles** :
  - SSH pour l'administration sécurisée des équipements réseau et serveurs
  - Protocoles d'administration spécifiques au Pare-feu PfSense (par exemple, HTTPS pour l'interface web d'administration)

Les mesures de filtrage pour sécuriser l'architecture réseau avec PfSense :

1. **Filtrage des adresses IP** : Autorisez uniquement le trafic provenant des adresses IP autorisées et bloquez les adresses IP suspectes ou non autorisées.

2. **Filtrage des ports** : Autorisez uniquement les ports nécessaires pour les services spécifiques que vous utilisez (par exemple, le port 80 pour HTTP, le port 443 pour HTTPS, le port 22 pour SSH, etc.). Bloquez les ports non utilisés pour réduire la surface d'attaque.

3. **Filtrage par protocole** : Autorisez uniquement les protocoles nécessaires pour les applications spécifiques. Par exemple, autorisez le protocole TCP pour les connexions web, le protocole UDP pour les services de streaming, etc.

4. **Filtrage par état de connexion** : Utilisez le suivi d'état de connexion pour autoriser uniquement les connexions entrantes qui correspondent à des connexions sortantes établies précédemment. Cela empêche les connexions entrantes non sollicitées.

5. **Filtrage par VLAN** : Configurez des règles de pare-feu spécifiques pour chaque VLAN afin de contrôler précisément le flux de trafic entre les différents réseaux.

6. **Filtrage de contenu** : Utilisez des fonctionnalités de filtrage de contenu pour bloquer les URL malveillantes, les scripts dangereux, les attaques par injection SQL, etc.

7. **Filtrage géographique** : Bloquez le trafic provenant de certaines régions géographiques connues pour être des sources d'attaques en ligne.

8. **Surveillance des journaux** : Activez la journalisation du trafic et surveillez régulièrement les journaux pour détecter les activités suspectes et les tentatives d'intrusion.