Architecture de sécurité

Architecture de sécurité

Introduction

Les éléments de sécurité de chaque composant de Pandora FMS sont décrits, conformément aux réglementations telles que PCI/DSS, ISO 27001, ENS, LOPD et autres.

En outre, une description spécifique des mécanismes de sécurité de chaque élément de Pandora FMS est incluse, ainsi que les risques possibles et la manière de les atténuer, en utilisant les outils disponibles dans Pandora FMS ou d'autres mécanismes possibles.

Mise en œuvre de la sécurité générale

- Les composants de Pandora FMS ont documenté leurs ports d'entrée et de sortie, de sorte qu'il est possible de sécuriser au moyen de *firewalls* tous les accès vers et depuis ses composants.
- Sécuriser le trafic grâce au cryptage et aux certificats : Pandora FMS, à tous les niveaux (fonctionnement de l'utilisateur, communication entre les composants), prend en charge le cryptage SSL/TLS et les certificats aux deux extrémités.
- Système d'accès à double authentification : Un système de double authentification peut être mis en place. Le premier, au niveau de l'accès (HTTPS) intégré avec n'importe quel système de jetons Opensource ou commercial.
- Système d'authentification des tiers : Au niveau de l'application, il est géré par Pandora FMS, qui peut être authentifié par rapport aux éléments suivants LDAP ou Active Directory.
- SSO (Single Sign-On), avec SAML.
- Politiques de sécurité dans la gestion des utilisateurs : La gestion des utilisateurs est limitée par des politiques au niveau du profil de l'utilisateur et au niveau du profil de visibilité des opérations, défini comme le système ACL Extended.
- Possibilité d'auditer les actions des éléments contrôlés : Pandora FMS vérifie toutes les actions des utilisateurs, y compris les informations sur les champs modifiés ou supprimés. Il comprend également un système de validation des signatures pour ces enregistrements.
- Transfert des données d'audit vers des gestionnaires de journaux externes : Les journaux d'audit *logs* sont disponibles pour exportation via SQL et peuvent être intégrés dans une source tierce pour une plus grande sécurité, en temps quasi réel.
- Séparation physique des composants : Ils fournissent une interface à l'utilisateur et aux conteneurs d'informations (*filesystem*). Les données stockées dans la base de données et le *système de fichiers* qui stocke les informations de configuration de la supervision peuvent se trouver sur des machines physiques distinctes, sur des réseaux différents, protégés par des systèmes périmétriques.
- Politique active en matière de mots de passe : Elles permettent d'appliquer une politique stricte de gestion des mots de passe pour l'accès aux utilisateurs de l'application (console web).
- Chiffrement des données sensibles : Le système permet de stocker les données les plus sensibles, telles que les identifiants d'accès, de manière cryptée et sécurisée.

Sécurité des composants de l'architecture

Server

- Le serveur nécessite des permissions root et peut être installé (avec des limitations) avec des permissions non root (uniquement sur les systèmes GNU/Linux).
- Le serveur a besoin d'un accès direct (lecture et écriture) aux fichiers de configuration à distance des agents, qui sont propagés périodiquement lorsque les agents contactent le serveur. Ces fichiers sont protégés par *filesystem*, avec les permissions standard.
- Le serveur en tant que tel n'écoute sur aucun port. C'est le serveur Tentacle qui écoute sur un port, le serveur n'accède qu'aux fichiers qu'il laisse sur le disque.
- Le serveur possède son propre journal d'événements détaillé (*log*).
- Le serveur se connecte à la base de données principale via une connexion MySQL/TCP standard.
- Une partie du code peut être demandée dans des conditions contractuelles spécifiques (pour les clients uniquement).

Vulnérabilités et garanties potentielles

- Accès non autorisé aux fichiers de configuration de l'agent.

Solution : Implémenter un conteneur externe sécurisé pour les fichiers de configuration externes, via NFS.

- Insertion de commandes dans les agents distants par la manipulation des fichiers de configuration stockés dans le conteneur de configuration.

Solution : Désactiver la configuration à distance sur les agents particulièrement sensibles après la configuration et les laisser fonctionner sans pouvoir modifier quoi que ce soit à distance, pour une sécurité absolue. Supervision à distance - sans agents - des appareils les plus sensibles.

- Vulnérabilité aux attaques par falsification d'informations, en simulant des agents qui ne sont pas enregistrés dans le système ou en supplantant leur identité. Pour éviter cela, *plusieurs mécanismes peuvent être utilisés* :

1. Mécanisme de protection par mot de passe (qui fonctionne par groupe).
2. Limiter l'auto création des agents et les créer manuellement.
3. Limiter la capacité d'auto-détection des changements dans l'agent et ne pas intégrer les nouvelles informations du XML dans l'existant.

- Capture malveillante de la communication entre le serveur et la console (capture du trafic réseau).

Solution : Activez la communication TLS entre le serveur et la base de données MySQL.

Tentacle

- Tentacle est un service Internet officiel, documenté comme tel par l'IANA. Cela signifie qu'il peut être facilement protégé par n'importe quel outil de sécurité périmétrique.
- Aucun utilisateur root ou privilège spécial n'est requis.
- Il comporte quatre niveaux de sécurité : aucun cryptage (par défaut), SSL/TLS de base, SSL/TLS avec certificat aux deux extrémités, et SSL/TLS avec validation du certificat et de l'autorité de certification (recommandé).
- Spécifiquement conçu pour ne donner aucun indice aux intrus potentiels dans les messages d'erreur et avec des délais d'attente spécifiques (*timeouts*) pour décourager les attaques par force brute.
- Il possède son propre audit *log*.
- 100% du code est accessible (sous licence Opensource GPL2).

Vulnérabilités et garanties potentielles

- Attaques sur *filesystem*. Vous devez accéder au conteneur de configuration.

Solution : Il est protégé de la même manière que le serveur, par un système NFS externe sécurisé.

- Les attaques par saturation (DoS).

Solutions : Monter une solution HA sur le service TCP qu'il offre pour l'équilibrage, ou un cluster actif/actif. Toute solution matérielle ou logicielle disponible est acceptable puisqu'il s'agit d'un service TCP standard.

Web console

- Pas de root requis, s'installe avec un utilisateur non privilégié.
- Doit avoir accès au référentiel de configuration de l'agent (*filesystem*).
- Écouter sur les ports HTTP ou HTTPS standard.
- Enregistre toutes les demandes via l'enregistrement des demandes HTTP.
- Fournit une API publique via HTTP/HTTPS, secured avec des informations d'identification et une liste d'adresses IP autorisées à l'avance.
- Il existe un audit spécifique à l'application, qui enregistre l'activité de chaque utilisateur sur chaque objet du système.
- L'accès de chaque utilisateur à n'importe quelle section de l'application peut être restreint, et il est même possible de créer des administrateurs avec des autorisations restreintes.
- L'application intègre un double système d'authentification.
- L'application intègre un système d'authentification déléguée (LDAP, AD).
- Un système en lecture seule peut être monté, sans accès aux configurations des périphériques.
- Les informations sensibles (mots de passe) peuvent être stockées sous forme cryptée dans la base de données.
- L'application se connecte à la base de données principale via une connexion MySQL/TCP standard.
- Une partie du code peut être demandée dans des conditions contractuelles spécifiques (pour les clients uniquement).
- Il y a une forte implémentation des politiques de sécurité concernant les mots de passe

(longueur, changement forcé, historique, type de caractères valides, etc.).

Vulnérabilités et garanties potentielles

- Attaques sur *filesystem*. Le conteneur de configuration doit être accessible.

Solution : Il est protégé de la même manière que le serveur, par un système NFS externe sécurisé.

- Attaques par force brute ou par dictionnaire contre l'authentification de l'utilisateur.

Solutions :

1. Mettre en œuvre une politique de mots de passe complexes.
2. Mettre en œuvre un mécanisme de double authentification.

- Capture du trafic (*eavesdropping*) du trafic vers la console.

Solution : Implémentez SSL/TLS.

- Capture du trafic (*eavesdropping*) du trafic vers la base de données.

Solution : Implémentez SSL/TLS.

- Les attaques par injection SQL pour obtenir des informations confidentielles à partir de la base de données de l'application.

Solution : Mettre en place un stockage crypté des données.

- Mauvaise utilisation (intentionnelle ou non) par les utilisateurs de l'application.

Solutions :

1. Activer l'audit *log* et montrer aux utilisateurs qu'il existe et qu'il est exact.
2. Activez le système ACL étendu pour restreindre autant que possible les rôles de chaque utilisateur.
3. Exporter régulièrement l'audit *log* vers un système externe.

- Exécution d'un code malveillant dans les outils de la console locale, en remplaçant des fichiers binaires.

Solution : Renforcement (*hardening*) du serveur contenant l'application.

Agents

- Peut être exécuté sans les autorisations du superutilisateur (avec des fonctionnalités limitées).
- La gestion à distance des agents peut être désactivée (localement et à distance), afin de minimiser l'impact d'une intrusion sur le système hôte.
- L'agent n'écoute pas les ports du réseau, il se connecte uniquement au serveur Pandora FMS.
- Chaque exécution fait l'objet d'un procès-verbal.

- Par défaut, les fichiers de configuration sont protégés par les autorisations *filesystem*. Seul un utilisateur disposant des droits de super administrateur peut les modifier.
- 100 % du code est accessible (sous licence Opensource GPL2).

Vulnérabilités et garanties potentielles

- Intrusion dans le système hôte permettant de distribuer des commandes malveillantes aux agents.

Solutions :

1. Limiter les utilisateurs qui peuvent effectuer ces modifications de politique ou de configuration (via l'ACL ordinaire de la console ou l'ACL étendue).
2. Activer le mode lecture seule (*readonly*) pour les agents (ne pas autoriser les modifications de configuration à distance), pour les systèmes particulièrement sensibles.

- Faiblesse du *filesystem* permettant la modification des fichiers.

Solution : Corriger les paramètres de permission.

- Exécution de *plugins* ou de commandes malveillantes.

Solutions :

1. Limiter les utilisateurs qui peuvent télécharger des exécutables (via l'ACL ordinaire de la console ou l'ACL étendue).
2. Effectuer un audit de *plugins*new.

Base de données

- Il s'agit d'un produit standard (MySQL).

Vulnérabilités et garanties potentielles

- *Eavesdropping* (capture du trafic réseau).

Solution : Implémentation d'une connexion TLS sécurisée. MySQL le supporte.

- Autorisations incorrectes.

Solution : Configuration correcte des autorisations d'accès.

- Vulnérabilités connues de MySQL : Un plan de mise à jour du serveur MySQL doit être établi afin de le maintenir aussi à jour que possible et d'éviter ainsi les éventuelles vulnérabilités dues à d'anciennes versions.

Sécurisation du système de base

Le *hardening* des systèmes est un point clé de la stratégie de sécurité globale d'une entreprise.

En tant que fabricants, une série de recommandations sont émises pour effectuer une installation sécurisée de tous les composants Pandora FMS, sur la base d'une plateforme de serveur standard RHEL 8 ou Ubuntu.

Les mêmes recommandations sont valables pour tout autre système de supervision basé sur GNU/Linux.

Références d'accès

Pour accéder au système, des utilisateurs à accès nominatif seront créés, sans privilèges et avec un accès limité aux besoins qu'ils ont.

Idéalement, l'authentification de chaque utilisateur devrait être intégrée à un système d'authentification à deux facteurs basé sur un jeton. Il existe des alternatives gratuites et sécurisées telles que Google Authenticator® qui peuvent être intégrées dans GNU/Linux et qui sortent du cadre de ce guide. Pensez sérieusement à les utiliser.

S'il est nécessaire de créer d'autres utilisateurs pour les applications, il doit s'agir d'utilisateurs sans accès à distance (pour ce faire, désactivez leur *Shell* ou une méthode équivalente).

Accès au super-utilisateur

Si certains utilisateurs doivent avoir des droits d'administrateur, la commande sudo sera utilisée.

Système d'exploitation mis à jour

Il suffit d'être connecté à l'internet ou de configurer dnf ou apt pour utiliser un serveur *proxy*.

Cette commande peut entraîner des problèmes potentiels liés à la modification des bibliothèques, des configurations, etc. Il est important de mettre à jour le système d'exploitation avant de mettre le système en production. Si vous révisez un système de production déjà en service, il se peut que vous ne deviez mettre à jour que les composants critiques, par exemple ceux qui présentent une vulnérabilité.

Par exemple, pour mettre à jour uniquement MySQL sur un système RHEL : `dnf update mysql-`

`server`.

La mise à jour du système d'exploitation est un processus qui doit être régulier. L'inventaire des paquets du système permet d'interroger les versions vulnérables et d'exécuter les mises à jour d'urgence.

Audit d'accès

Vous devez avoir le journal de sécurité `/var/log/secure` actif et superviser ces *logs* avec la supervision.

Par défaut, cette fonction est activée. Si ce n'est pas le cas, vérifiez le fichier `/etc/rsyslog.conf` ou `/etc/syslog.conf`.

Il est recommandé que les *logs* du système d'audit soient portés et collectés par un système externe de gestion des *logs*. Pandora FMS peut le faire et il sera utile d'établir des alertes ou de les examiner de manière centralisée si nécessaire.

Serveur SSH

Le serveur SSH permet de se connecter à distance aux systèmes GNU/Linux pour l'exécution de commandes, il s'agit donc d'un point critique et il faut s'en assurer en faisant attention aux points suivants (pour cela, il faut éditer le fichier `/etc/ssh/sshd_config` et ensuite redémarrer le service).

- Modifier le port par défaut :

```
#Port 22     ->     Port 31122
```

- Désactiver la connexion du superutilisateur root login :

```
#PermitRootLogin yes        ->    PermitRootLogin no
```

- Désactiver le transfert de port port forwarding :

```
#AllowTcpForwarding yes        ->    AllowTcpForwarding no
```

- Désactiver tunneling :

```
#PermitTunnel no        ->    PermitTunnel no
```

- Supprimez les clés SSH pour l'accès à distance à partir de root. En supposant qu'il n'y ait qu'un seul utilisateur valide pour l'accès à distance (par exemple pfms), s'il y en a d'autres, vérifiez-les également. Pour ce faire, vérifiez le contenu du fichier `/home/pfms/.ssh/authorized_keys` et vérifiez à quelles machines ils appartiennent, *supprimez-le si vous pensez qu'il ne devrait pas y en*

avoir.

- Établissez un avis d'accès à distance standard expliquant que le serveur est accessible en privé et que toute personne ne disposant pas d'informations d'identification doit se déconnecter :

```
Banner /etc/issue.net
```

Serveur MySQL

Si MySQL ne gère qu'un élément interne, vérifiez avec netstat qu'il n'écoute que sur *localhost* :

```
netstat -an | grep 3306 | grep LIST
tcp        0      0 0.0.0.0:3306            0.0.0.0:*               LISTEN
```

Dans l'exemple ci-dessus vous écoutez sans restrictions, vous devez éditer le fichier `/etc/my.cnf`, section `[mysqld]`, en ajoutant la ligne suivante :

```
bind-address = 127.0.0.1
```

Après avoir redémarré le service, vérifiez à nouveau le port d'écoute.

Mot de passe MySQL

Connectez-vous à la console MySQL avec un utilisateur privilégié :

```
mysql -h host -u root -p
```

Vérifiez que le mot de passe est complexe et que vous avez demandé un mot de passe. Si ce n'est pas le cas, il est défini à l'aide de la commande :

```
mysqladmin password
```

Cette mesure de sécurité est essentielle pour protéger les bases de données non seulement contre les attaques externes, mais aussi contre les abus des utilisateurs internes.

Serveur web Apache

```

ServerTokens Prod
```

Ajoutez la ligne ci-dessus pour cacher la version du serveur web (Apache, Nginx) dans les en-têtes d'information du serveur :

- `/etc/httpd/conf/httpd.conf` (RHEL).
- `/etc/apache2/conf-enabled/pandora_security.conf` (Ubuntu server)

Moteur d'application PHP

Afin de sécuriser le moteur d'application sur lequel fonctionne Pandora FMS, il peut être nécessaire, dans certains environnements particulièrement sensibles sur le plan de la sécurité, de sécuriser l'accès à l'application de sorte que les *cookies* de session ne soient transmis qu'avec SSL.

L'application ne fonctionnera pas si elle est utilisée sur HTTP (sans cryptage).

Pour ce faire, la configuration suivante *tokens* doit être incluse dans le fichier `php.ini` :

```
session.cookie_httponly = 1
session.cookie_secure = 1
```

Minimiser les services dans le système

Cette technique, qui peut être très complète, consiste à supprimer tout ce qui est inutile sur le système. Cela permet d'éviter d'éventuels problèmes à l'avenir avec des applications mal configurées et dont on n'a pas vraiment besoin. Pour simplifier l'approche de cette pratique, ne considérez que les applications qui ont un port ouvert sur la machine : `netstat -tulpn`.

Chaque port doit être examiné et l'application qui se cache derrière doit être connue. Cela peut être fait en utilisant la commande lsof, qui doit être installée avec dnf ou apt.

Les services qui écoutent localhost (`127.0.0.1`) sont plus sûrs que ceux qui écoutent toutes les adresses IP (`0.0.0.0`) et certains d'entre eux, s'ils écoutent sur un port ouvert, devraient être corrigés pour n'écouter que localhost.

En utilisant le système d'inventaire des processus de Pandora FMS, il est nécessaire de vérifier qu'aucun nouveau processus n'est lancé au fil du temps.

Configuration supplémentaire

Synchronisation du temps NTP

Il est recommandé de configurer la synchronisation de l'heure du système sur un système RHEL :

```
dnf install ntpdate
echo "ntpdate 0.us.pool.ntp.org"> /etc/cron.daily/ntp
chmod 755 /etc/cron.daily/ntp
```

Suivi local

Le système doit avoir un Pandora FMS Software Agent installé et fonctionnant sur le serveur PFMS. Pour le système d'exploitation MS Windows®, à partir de la version 761, les exécutables d'installation sont signés numériquement.

Les contrôles actifs suivants sont recommandés en plus des contrôles standard :

- Plug-in de sécurité active (*plugin*).
- Inventaire complet du système (en particulier des utilisateurs et des paquets installés).
- Collecte des *logs* du système et de la sécurité :

```
module_plugin grep_log_module /var/log/messages Syslog \.\*
module_plugin grep_log_module /var/log/secure Secure \.\*
```

Une fois l'agent logiciel installé, les informations suivantes doivent au moins être définies manuellement dans l'onglet de l'agent :

- Description.
- Adresse IP (si vous en avez plusieurs, indiquez-les toutes).
- Groupe.
- Département, responsable et lieu d'implantation (*custom fields*).

Supervision de la sécurité sous GNU/Linux

Le plugin officiel official plugin permet de superviser de manière proactive la sécurité de l'agent, à chaque exécution, presque en temps réel, en proposant des vérifications qui peuvent alerter de certains événements pertinents.

Ce *plugin* est destiné à fonctionner uniquement sur des machines GNU/Linux modernes. Il contient une version personnalisée de John the ripper 1.8 + des correctifs Contrib avec des binaires statiques 32-bit et 64-bit. Le concept principal de *plugin* est d'être monolithique, de détecter ce qui peut être renforcé et d'essayer de résoudre les différences entre les distributions sans rien demander à l'administrateur, de sorte que le déploiement puisse être le même pour n'importe quel système, en ignorant les versions, *distro* ou l'architecture.

Ce *plugin* vérifiera :

- Auditer les mots de passe des utilisateurs, en utilisant le dictionnaire (fourni) avec les 500 mots de passe les plus courants. Si vous avez des centaines d'utilisateurs, vous devrez probablement personnaliser le *plugin* pour qu'il ne s'exécute que toutes les 2 à 6 heures. Vous pouvez personnaliser le dictionnaire de mots de passe en ajoutant le mot de passe type de votre organisation dans le champ « basic_security/password-list ».
- SSH ne fonctionne pas sur le port par défaut.
- SSH ne permet pas l'accès à partir de root.
- Que FTP ne fonctionne pas sur le port par défaut.
- Vérifier si un serveur MySQL fonctionne sans que le mot de passe root soit défini.
- Autres chèques.

Retour à l'index de la documentation de Pandora FMS