# CIBLE DE SECURITE

## UBIKA WAAP Gateway 6.11.1

## Historique

| Version | Date | Auteur | Notes |
|---|---|---|---|
| **v4.0** | **25/07/2022** | Matthieu Vaidie | Mise à jour pour la version 6.10.0 |
| **v5.1** | **14/03/2023** | Matthieu Vaidie | Mise à jour pour la version 6.11.1 |
| **v5.2** | **13/11/2023** | Matthieu Vaidie | Ajout d'un historique du document |

## Sommaire


CIBLE DE SECURITE ..... 1
UBIKA WAAP Gateway 6.11.1 ..... 1
Sommaire ..... 3
Identification ..... 5
Identification du document ..... 5
Identification du produit ..... 5
Description du produit ..... 6
Description générale ..... 6
Principe de fonctionnement ..... 6
Description des dépendances ..... 8
Description de l'environnement technique de fonctionnement ..... 8
Périmètre de l'évaluation ..... 8
Problématique de sécurité ..... 9
Description des utilisateurs typiques ..... 9
Description des hypothèses sur l'environnement ..... 9
H1. Environnement physique ..... 9
H2. Environnement technique ..... 9
H3. Utilisateurs ..... 10
H4. Poste de l'administrateur ..... 10
Description des biens sensibles ..... 10
B1. Application web protégée par la TOE ..... 10
B2. Application cliente accédant à l'application web protégée par la TOE ..... 10
B3. Configuration de la TOE ..... 10
B4. Fichiers de journalisations de la TOE ..... 11
B5. Clés de chiffrement pour la terminaison TLS/SSH ..... 11
Description des menaces ..... 11
M1. Exécution de code arbitraire par l'application ou le serveur web ..... 11
M2. Exécution de code arbitraire sur le navigateur web des utilisateurs ..... 11
M3. Usurpation de l'identité d'un utilisateur de l'application ..... 12
M4. Déni de service non distribué sur les moteurs de filtrage de la TOE ..... 12

M5. Contournement des moteurs de filtrage de la TOE ..... 12
M6. Exécution de code arbitraire sur la TOE ..... 12
M7. Accès illégitime à la configuration de la TOE ..... 12
M8. Accès illégitime aux fichiers de journalisation de la TOE ..... 12
M9. Mise à jour illégitime de la TOE ..... 12
Description des fonctions de sécurité ..... 13
F1. Filtrage des données HTTP ..... 13
F2. Sécurité des cookies ..... 14
F3. Sécurité des données XML ..... 14
F4. Sécurité des données JSON ..... 14
F5. Terminaison TLS/SSH ..... 15
F6. Validation de certificats clients ..... 15
F7. Authentification des administrateurs ..... 15
F8. Durcissement du socle de l'appliance ..... 16
F9. Politique et sécurisation des mises à jour ..... 17
F10. Stockage sécurisé des clés de chiffrement ..... 18
Matrices de couvertures ..... 18
Menaces et biens sensibles ..... 18
Menaces et fonctions de sécurité ..... 19

# Identification

## Identification du document

Ce document décrit la cible de sécurité relative à la certification de sécurité de premier niveau des technologies de l'information (CSPN).

## Identification du produit

Les informations d'identification du produit sont détaillées ci-dessous.

| | |
|---|---|
| **Editeur** | **UBIKA** |
| **Site web de l'éditeur** | https://www.ubikasec.com/ |
| **Nom commercial** | UBIKA WAAP Gateway |
| **Version évaluée** | 6.11.1 |
| **Catégorie de produit** | Pare-feu Applicatif Web (WAF) |

UBIKA WAAP Gateway est la nouvelle version du produit R&S®Web Application Firewall et plus anciennement iSuite 5.5.5 certifié le 16/09/2014 (Certificat ANSSI-CSPN-2014/05).

Ce nouveau nom est le résultat de l'acquisition de Rohde & Schwarz Cybersecurity SAS par la société Total Specific Solutions (TSS).

# Description du produit

## Description générale

UBIKA WAAP Gateway est un pare-feu applicatif web, également appelé WAAP pour Web Application and API Protection.

UBIKA WAAP Gateway permet de protéger les services et applications web des menaces que ce soit dans un contexte d'utilisation interne ou externe.

La protection des services et applications web est assurée par des règles de filtrage appliquées aux requêtes HTTP reçues. Ces règles de filtrage peuvent porter sur des adresses IP, des URL ou des données transmises dans les requêtes HTTP, en-têtes incluses (headers, cookies).

Les actions et les éléments inspectés sont définis par les administrateurs. Les éléments inspectés peuvent être confrontés à des motifs choisis par les administrateurs ou définis par l'éditeur de logiciel.

Afin de vérifier l'innocuité des requêtes web UBIKA WAAP Gateway doit être placé entre l'utilisateur et les serveurs web. De la sorte toutes les communications transitent par le logiciel qui peut alors inspecter les flux selon les règles définies pour le service protégé.

Le produit permet également la gestion des chiffrements des données échangées avec l'utilisateur.

## Principe de fonctionnement

Le produit UBIKA WAAP Gateway doit être mis en œuvre en amont des serveurs et services web à protéger vis-à-vis des zones à risque. Le produit s'utilise en coupure entre les utilisateurs et les serveurs web. Il permet ainsi d'analyser les requêtes HTTP, de les bloquer ou de les modifier et de gérer les tunnels SSL/TLS (voir Figure 1).

Aucun prérequis n'est demandé côté utilisateur concernant le périphérique. Le navigateur web utilisé pour accéder aux services et applications web doit être suffisamment récent pour supporter les suites cryptographiques définis pour les connexions TLS.

Les serveurs web sont redondés et configurés en tant que Load Balancer members dans la TOE.

Les administrateurs sont connectés à l'équipement via une interface réseau dédiée qui leur permet l'accès aux fonctions d'administration (voir Figure 2).

L'évaluation porte sur le workflow « WAAP Default ».

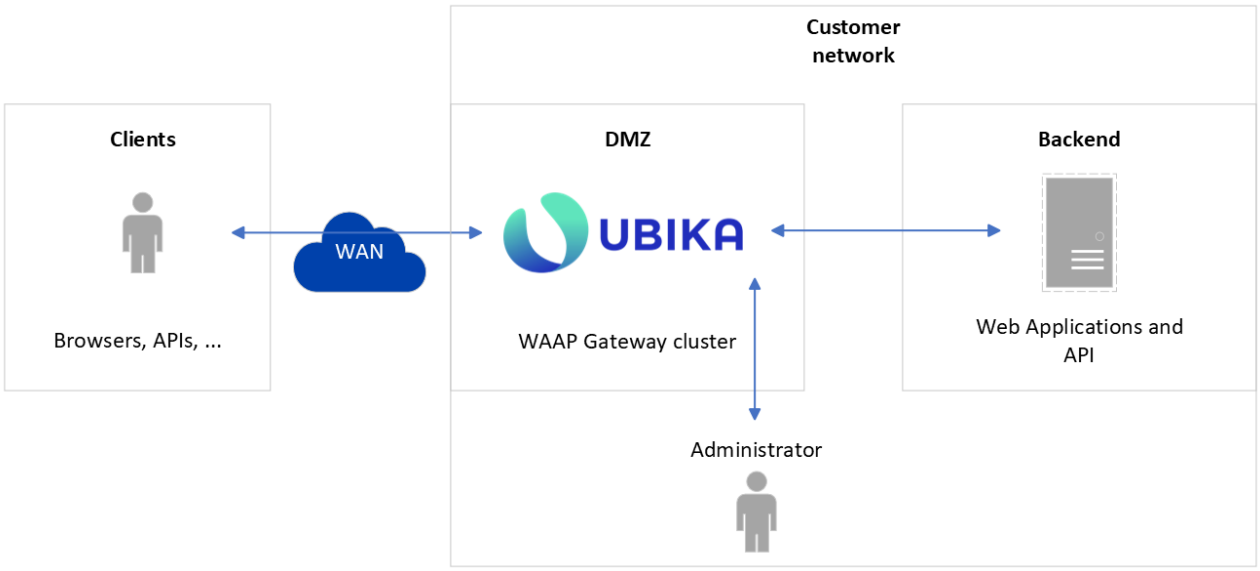


Figure 1 - Architecture cible d'utilisation

## Description des dépendances

Le produit évalué est une appliance autonome. Aucune application ou module supplémentaire n'est à installer sur l'appliance pour son bon fonctionnement. L'appliance se configure à l'aide d'une application de configuration dédiée à installer sur le poste de l'administrateur.

Le produit évalué s'appuie sur le composant OpenSSL pour la gestion des terminaisons TLS. Ce composant est intégré dans l'appliance. Il est mis à jour très régulièrement, notamment dès la publication de nouvelle vulnérabilité. La mise à jour de ce composant utilise la procédure standard de mise à jour (voir F9. Politique et sécurisation des mises à jour).

## Description de l'environnement technique de fonctionnement

Pour l'évaluation, les utilisateurs utiliseront différentes technologies de navigateur web sur différents supports. L'utilisateur peut se servir d'un micro-ordinateur, d'un smartphone ou d'une tablette et ainsi utiliser les différents navigateurs web associés : Internet Explorer, Firefox, Chrome, Safari sur les systèmes Windows, Linux, iOS, Android, etc...

L'application UBIKA WAAP Gateway est interconnectée à la fois à la partie « externe » qui peut être un intranet, un extranet ou internet et à une partie « interne » sur laquelle se trouve au moins une application web. L'administration de l'équipement se fait sur une interface dédiée (voir Figure 2).

Dans le cadre de l'évaluation, différentes applications web seront positionnées sur la partie « interne » de façon à représenter le plus exhaustivement possible les technologies qui peuvent être présente sur les réseaux internet et d'entreprises.
Nous nous attacherons également à évaluer le filtrage applicatif de UBIKA WAAP Gateway avec les web services.


Figure 2 - Configuration des interfaces réseaux

## Périmètre de l'évaluation

La cible d'évaluation sont les appliances 1450 et 4450 de UBIKA WAAP Gateway 6.11.1.

Le périmètre de l'évaluation porte sur les fonctionnalités :

- d'analyse des requêtes HTTP(S),
- de redondance Actif/Passif derrière un Load Balancer (HA Proxy sera utilisé dans le laboratoire de test).

# Problématique de sécurité

## Description des utilisateurs typiques

Les deux profils d'utilisateurs de UBIKA WAAP Gateway sont :

- Les utilisateurs en transit sur le produit qui souhaite accéder à un service ou une application web ;
- L'administrateur ayant un accès privilégié au produit lui permettant de le configurer.

L'utilisateur n'a pas conscience qu'il utilise le produit UBIKA WAAP Gateway car il agit comme un intermédiaire entre le client et l'application web. Ce fonctionnement est typique des « reverse-proxies ».

L'administrateur utilise le produit afin de diffuser les services et applications web et de gérer les politiques de sécurité et de filtrage vers ces services et applications.

Il est considéré pour l'évaluation que les administrateurs de UBIKA WAAP Gateway sont formés et de confiance. Leurs postes informatiques sont considérés comme sûrs.

## Description des hypothèses sur l'environnement

### H1. Environnement physique

La TOE est positionnée dans un local dont l'accès est restreint à des personnes de confiance.

### H2. Environnement technique

Il est considéré pour l'évaluation de la TOE que les recommandations en termes d'architecture réseau sont respectées. Notamment la séparation des zones selon la confiance qui leur sont accordées ainsi que le positionnement d'un équipement de filtrage réseau en amont du produit UBIKA WAAP Gateway. Cette séparation doit être mise en œuvre par la définition de règles pare feux réseaux spécifiques, cette segmentation ne doit permettre uniquement aux administrateurs d'atteindre l'interface ETH0.

La configuration réseau doit donc respecter les éléments suivants (voir Figure 2) :

- Une interface « d'administration », ETH0 sur la figure 2, est dédiée pour les flux d'administration. (Canal qui reçoit les actions et les instructions de configuration à réaliser, seuls les administrateurs y ont accès).

- Une interface « publique », ETH1 sur la figure 2, est dédiée pour les flux d'entrée. (Canal qui reçoit les requêtes des clients provenant de l'extérieur)

- Une interface « interne », ETH2 sur la figure 2, est dédiée pour les flux de sortie. (Canal qui permet de communiquer avec les serveurs applicatifs protégés)

### H3. Utilisateurs

Les administrateurs, dont un des rôles est la définition et la mise en place des règles de sécurité, ont les formations nécessaires à l'utilisation du produit et sont de confiance. Entre autres, nous estimons que les règles définies n'introduisent pas de problèmes de sécurité pour le produit, les services ou applications web. Ainsi les scénarii de règles récursives non terminales, de motifs ou d'encodage mal définis pour garantir la sécurisation des sites, les « evil regex » et tous autres cas similaires de règles dangereuses sont écartés.

Les utilisateurs respectent les règles de bonne pratique concernant la gestion de leurs mots de passe. Il est considéré que ces mots de passe n'ont pas été compromis ou usurpé par un tiers.

### H4. Poste de l'administrateur

Il est considéré que le poste de l'administrateur n'a pas été compromis.

## Description des biens sensibles

Les biens sensibles que la TOE doit protéger sont :

### B1. Application web protégée par la TOE

Les pages web, les données, les ressources ainsi que les services délivrés par l'application. La confidentialité des données, matérialisé par des zones d'accès restreints au sein de l'appliance. Il s'agit de protéger l'accès à des données confidentielles stockées sur les serveurs web protégés par l'appliance.

### B2. Application cliente accédant à l'application web protégée par la TOE

- Navigateur ou application cliente conçus pour se connecter aux applications et services web et interpréter les données renvoyées comprenant : chemin (path), paramètres (query et body), entêtes (headers), cookies, document XML, document JSON.

### B3. Configuration de la TOE

Fichiers et bases de données stockant la configuration du produit relatifs, entre autres, aux applications protégées, aux règles de filtrages appliquées et à l'authentification des administrateurs du produit.

### B4. Fichiers de journalisations de la TOE

Traces des événements survenus sur le produit.

### B5. Clés de chiffrement pour la terminaison TLS/SSH

La rupture de chiffrement nécessaire à l'analyse du flux HTTPS nécessite le stockage des clés de chiffrement. Ces clés peuvent être importées et exportées par un administrateur habilité. (Ayant les autorisations nécessaires).
Les besoins de sécurité de chacun de ces biens sont donnés ci-dessous :

| Bien | Intégrité | Disponibilité | Confidentialité | Authenticité |
|---|---|---|---|---|
| **B1. Application web** | ✓ | ✓ | ✓ | ✓ |
| **B2. Application client** | ✓ | ✓ | ✓ | ✓ |
| **B3. Configuration de la TOE** | ✓ | ✓ | ✓ | ✓ |
| **B4. Fichiers de journalisations de la TOE** | ✓ | ✓ | ✓ | |
| **B5. Clés de chiffrement** | | | ✓ | |

Tableau 1 – Besoins de sécurité des biens sensibles

## Description des menaces

Les menaces qui portent sur les biens sensibles de la TOE sont les suivantes :

### M1. Exécution de code arbitraire par l'application ou le serveur web

L'attaquant parvient à faire exécuter du code par l'application ou une commande sur le serveur web hébergeant l'application.
Il pourra alors porter atteinte à l'intégrité et à la disponibilité des services web. En contournant les règles de filtrage, l'attaquant pourra également porter atteinte à la confidentialité des données hébergées par les applications web.

### M2. Exécution de code arbitraire sur le navigateur web des utilisateurs

L'attaquant parvient à injecter des données malveillantes sur un service web pourra porter atteinte aux navigateurs web des clients des services web. Par exemple des attaques XSS qui injecte du code stocké par l'application ou le service web et ensuite envoyé et interprété par les navigateurs des autres utilisateurs.
Les menaces qui pèsent sur les navigateurs web et indirectement sur la totalité du périphérique utilisé pour accéder aux services web sont la perte d'intégrité, de confidentialité, de disponibilité. La compromission totale des navigateurs et des périphériques est envisageable.

### M3. Usurpation de l'identité d'un utilisateur de l'application

L'attaquant parvient à se faire passer par un autre utilisateur dans l'application. Il parvient ainsi à accéder à des ressources protégées, il pourra ainsi porter atteinte à la confidentialité des données.

### M4. Déni de service non distribué sur les moteurs de filtrage de la TOE

L'attaquant parvient à rendre indisponible ou inactif les moteurs de sécurités de la TOE. La TOE ne permet pas aux requêtes de l'attaquant d'atteindre l'application web. Dans ce cas l'attaquant aura rendu indisponible l'application web protégée par la TOE.

### M5. Contournement des moteurs de filtrage de la TOE

L'attaquant parvient à contourner les moteurs de sécurités de la TOE et à envoyer des requêtes à l'application web sans action de filtrage. L'attaquant pourra alors porter atteinte à l'intégrité des services web et à la confidentialité des données.

### M6. Exécution de code arbitraire sur la TOE

L'attaquant envoie une requête malicieuse lui permettant d'exécuter des commandes sur la TOE, via l'interface publique.
Selon les droits attribués lors de l'exécution de commandes et les commandes disponibles l'attaquant pourra altérer l'intégrité, la disponibilité et la confidentialité de la TOE.

### M7. Accès illégitime à la configuration de la TOE

L'attaquant parvient à accéder accès aux données de configuration de la TOE en envoyant des requêtes via l'interface publique, by-passant ainsi la restriction de ces données à l'interface d'administration. L'attaquant pourra ainsi compromettre la disponibilité, l'intégrité et la confidentialité des données de configurations.

### M8. Accès illégitime aux fichiers de journalisation de la TOE

L'attaquant parvient à accéder accès aux fichiers de journalisation de la TOE en envoyant des requêtes via l'interface publique, by-passant ainsi la restriction de ces données à l'interface d'administration. L'attaquant pourra ainsi compromettre la disponibilité, l'intégrité et la confidentialité des fichiers de journalisation.

### M9. Mise à jour illégitime de la TOE

L'attaquant parvient à charger sur la TOE une mise à jour illicite et non officielle en utilisant la console d'administration. L'attaquant pourra ainsi copier et/ou remplacer des fichiers sur le système de la TOE et compromettre la disponibilité, l'intégrité et la confidentialité du système.

# Description des fonctions de sécurité

## F1. Filtrage des données HTTP

Les différents éléments de la requête http sont analysés : Path, Query, Headers, Cookies, Get Vars, Post Vars.

Identification des attaques via des règles de filtrage et des moteurs heuristiques. Les catégories d'attaques ainsi filtrées sont les suivantes :

- **Cross Site Scripting (XSS)** : la requête contient du contenu (donnée ou code exécutable) qui sera injecté dans les pages de l'application et envoyé aux autres clients.

- **Cross Site Request Forgery (CSRF)** : cas particulier des attaques XSS qui forcera les autres clients à effectuer des requêtes à leur insu.

- **Débordement de tampon (buffer overflow)** : l'attaque contient des données surnuméraires qui déborderont les espaces mémoires alloués par l'application et pourront mener à des corruptions mémoires ou des exécutions de code arbitraire.

- **Déni de Service** : la requête est une attaque dont l'objectif est de rendre l'application indisponible.

- **Inclusion de fichier** : la requête contient une référence vers une ressource qui sera chargée par l'application et éventuellement interprétée. Cette ressource peut être locale ou distante.

- **Injection de commande** : la requête contient une commande système qui pourrait être exécutée par le serveur d'application.

- **Injection LDAP** : la requête contient des fragments de commande LDAP qui pourraient être incluses dans des commandes légitimes de l'application et donc exécutées par l'application.

- **Injections SQL** : la requête contient des fragments de commande SQL qui pourraient être incluses dans des commandes légitimes de l'application et donc exécutées par l'application.

- **Injection XPath** : la requête contient un fragment de requête XPATH qui sera injecté aux requêtes légitimes de l'application.

- **Injections diverses** : la requête contient des injections dans d'autres langages (HTML, PHP, Java, Javascript).

- **Mauvaise configuration** : la requête contient une attaque liée à une mauvaise configuration de l'application.

- **Path Traversal** : la requête contient des références vers des fichiers en dehors de l'arborescence servie par l'application.

- **Redirection** : la requête contient une adresse de redirection contrôlée par l'attaquant et qui sera suivie par le client à son insu.

- **Scanneurs de vulnérabilités** : les requêtes proviennent de logiciels dédiés à l'audit d'applications.

Ces règles de filtrage sont exécutées par le moteur de sécurité ICX. En complément des moteurs avancés sont disponibles :

- Le moteur de sécurité *Adv. Detection Engine – SQLi* permet de détecter spécifiquement la catégorie d'attaque **Injections SQL**.
- Le moteur de sécurité *Adv. Detection Engine – XSS* permet de détecter spécifiquement la catégorie d'attaque **« Cross Site Scripting » (XSS)**.
- Le moteur de sécurité *Adv. Detection Engine – CMDi* permet de détecter spécifiquement la catégorie d'attaque **Injection de commande**.

Le moteur de sécurité *Normalization Engine* permet de détecter spécifiquement la catégorie d'attaque **Evasion** : la requête contient des attaques encodées dans le but de contourner les moteurs de sécurités.

## F2. Sécurité des cookies

- **Cookie ciphering** : chiffrement des cookies (pour éviter leur altération)
- **Cookie tracking** : détection d'altérations frauduleuses des cookies
- **Cookie virtualization** : remplacement des cookies applicatifs de manière transparente (empêche toute altération des cookies)

## F3. Sécurité des données XML

- **XML Parsing** : validation du format XML d'entrée (protection des parseurs XML du serveur), protection contre l'inclusion d'entités XML (protège contre l'injection de données externe, protection contre le téléchargement de ressources non sollicitées), protection contre la récursivité XML (récursivité des d'entités XML, profondeur du XML, pour protéger les parseurs XML du serveur)
- **XML Schema Validation** : validation par WSDL ou par XSD
- **XML Sign / XML Signature Verify** : signature / vérification de signature XML
- **XML Encrypt / XML Decrypt** : chiffrement / déchiffrement XML

## F4. Sécurité des données JSON

- **JSON Attribute Set** : validation du format JSON d'entrée (protection des parseurs JSON du serveur)
- **JSON Schema Validation** : validation par schéma JSON
- **Symmetric Encryption / Decryption** : chiffrement et déchiffrement du document JSON ou d'une partie du document.
- **Asymmetric Encryption / Decryption** : chiffrement et déchiffrement du document JSON ou d'une partie du document.
- **Data sign / Data Signature Verify** : signature / vérification de signature d'un document JSON
- **JWT Generate / JWT Parsing** : génère et signe un JWT / parse et vérifie la signature d'un JWT

## F5. Terminaison TLS/SSH

Pour accéder à l'application protégé le client établie une connexion TLSv1.2 ou TLSv1.3 avec la TOE pour assurer la confidentialité et l'intégrité des données.

L'administrateur qui souhaite configurer la TOE établie une connexion TLSv1.2 ou TLSv1.3. La configuration est différente des applications protégées.
La TOE permet aussi d'établir une connexion SSHv2 avec un poste administrateur de la TOE. Le mot de passe root de la TOE n'est pas disponible par défaut. Celui-ci est fourni uniquement par l'équipe Support UBIKA en cas de problème critique.

## F6. Validation de certificats clients

Validation des certificats clients (SSL PKI) (seuls les clients qui ont un certificat valide et déclaré de confiance peuvent accéder au site)

## F7. Authentification des administrateurs

La connexion à la GUI de configuration nécessite une authentification de l'utilisateur. Les identifiants des utilisateurs sont stockés dans l'appliance, seul un hash bcrypt des mots de passe est stocké pour assurer sa confidentialité. Cette authentification assure la confidentialité et l'intégrité des données de configuration. La connexion entre l'interface d'administration et l'appliance utilise un chiffrement TLS pour assurer la confidentialité des données échangées.

La recommandation est de dédier une interface physique de l'appliance à l'administration pour en limiter l'accès uniquement à une zone réseau d'administration telle que décrit dans l'hypothèse H2.

Les autorisations des administrateurs sont découpées en plusieurs rôles :

- **Backup operator** : Autorisation de créer, restaurer et supprimer des fichiers de sauvegarde.
- **Network administrator** : Autorisation de créer, lire, modifier et supprimer les objets réseaux (Load Balancer, VRRP, routes, interfaces, devices réseau).
- **Default** : Autorisation de lire les informations des objets appliance, routes, interfaces, devices réseau, licence, trace d'évènements (event log).
- **Log Files operator** : Autorisation de lire les fichiers de trace.
- **Security operator** : Autorisation de créer, lire, modifier et supprimer les politiques de sécurité incluant la gestion des exceptions (faux positifs).
- **Workflow operator** : Autorisation de créer, lire, modifier et supprimer les workflow et objets associés + autorisation de lire les politiques de sécurité.

- **Identity manager** : Autorisation de créer, lire, modifier et supprimer les configurations d'authentifications.

- **Reverse Proxy operator** : Autorisation de créer, lire, modifier et supprimer les configurations de Reverse proxy + Autorisation de lire les objets associés incluant la configuration ssl.

- **Tunnel operator** : Autorisation de créer, lire, modifier et supprimer les configurations de tunnel incluant les key stores SSL.

Ces rôles peuvent être combinés pour élargir les autorisations d'un administrateur.

### F8. Durcissement du socle de l'appliance

La TOE s'appuie sur une distribution linux. Des mesures de durcissement ont été mises en œuvre sur ce système pour améliorer sa sécurité et diminuer sa surface d'attaque. Le document « Recommandations de configuration d'un système GNU/Linux - v1.2 » (https://www.ssi.gouv.fr/uploads/2016/01/linux_configuration-fr-v1.2.pdf) a été utilisé.

Les recommandations suivantes ont été suivies :

- R1. Minimisation des services installés

- R2. Minimisation de la configuration

- R3. Principe de moindre privilège

- R5. Principe de défense en profondeur

- R6. Cloisonnement des services réseau

- R7. Journalisation de l'activité des services : ce point est activable par le client en exportant les journaux en Syslog

- R8. Mises à jour régulières. Les mises à jour régulières de la TOE intègrent les mises à jour système.

- R10. Architecture 64 bits

- R14. Installation de paquets réduite au strict nécessaire

- R18. Robustesse du mot de passe administrateur. Le mot de passe administrateur n'est pas disponible pour le client. Ainsi, il n'est pas possible d'effectuer des modifications du système non prévue par le produit ou d'installer des paquets directement sur le système.

- R19. Imputabilité des opérations d'administration

- R20. Installation d'éléments secrets ou de confiance. Remplacement de tous les secrets par défaut.

- R21. Durcissement et surveillance des services soumis à des flux arbitraires
- R22. Paramétrage des sysctl réseau
- R23. Paramétrage des sysctl système
- R26. Désactivation des comptes utilisateurs inutilisés
- R27. Désactivation des comptes de services
- R28. Unicité et exclusivité des comptes de services
- R29. Délai d'expiration de sessions utilisateurs
- R30. Nombre d'applications utilisant PAM réduit au nécessaire
- R36. Droits d'accès aux fichiers de contenu sensible limités aux utilisateurs nécessaires
- R37. Seuls les programmes spécifiquement conçus pour être utilisés avec les bits setuid (ou setgid) peuvent avoir ces bits de privilèges positionnés.
- R40. Sticky bit et droits d'accès en écriture
- R41. Sécurisation des accès pour les sockets et pipes nommées
- R42. Services et démons résidents en mémoire
- R43. Durcissement et configuration du service syslog
- R60. Privilèges des utilisateurs cibles pour une commande sudo
- R62. Spécifications de commande sans règles de négation
- R63. Arguments explicites dans les spécifications sudo, limitation de l'usage de wild card au strict nécessaire
- R64. Du bon usage de sudoedit

### F9. Politique et sécurisation des mises à jour

Des mises à jour régulières sont mises à disposition. Elles intègrent des mises à jour de composants système et logiciel ainsi que des mises à jour des politiques de sécurités. Ces mises à jour sont à la disposition de tous les clients sur le portail utilisateur (Section Tech Support > Download). Les mises à jour peuvent être installées et désinstallées à partir de la console d'administration. Les mises à jour sont signées par une clé privée, la signature est contrôlée par la TOE lors de son chargement pour détecter toute mise à jour illégitime.

### F10. Stockage sécurisé des clés de chiffrement

Lors de l'ajout de clés privés pour la terminaison TLS, une option est disponible pour activer le chiffrement de la clé privé sur la TOE. La passphrase autogénérée pour le chiffrement est stockée dans un fichier également chiffré.
Les administrateurs disposant des autorisations nécessaires ont la possibilité d'exporter ces clés dans des fichiers de sauvegarde, l'export des clés privées est clairement indiqué dans la console lors de l'export (cette option est désactivée par défaut). Les fichiers de sauvegarde sont chiffrés par une clé symétrique, le mot de passe est modifiable par l'administrateur.

## Matrices de couvertures

### Menaces et biens sensibles

Le tableau suivant résume la couverture des biens sensibles par les menaces. Chaque cellule contient la première lettre de la propriété de sécurité mise en péril par la menace (Intégrité, Disponibilité et Confidentialité).

B1. Application web protégée par la TOE
B2. Postes des utilisateurs de l'application web protégée par la TOE
B3. Configuration de la TOE
B4. Fichiers de journalisations de la TOE
B5. Clés de chiffrement pour la terminaison TLS

| | **B1** | **B2** | **B3** | **B4** | **B5** |
|---|---|---|---|---|---|
| M1. Exécution de code arbitraire par l'application ou le serveur web | IDC | | | | |
| M2. Exécution de code arbitraire sur le navigateur web des utilisateurs | | IDC | | | |
| M3. Usurpation de l'identité d'un utilisateur de l'application | C | | | | |
| M4. Déni de service sur les moteurs de filtrage de la TOE | D | | | | |
| M5. Contournement des moteurs de filtrage de la TOE | IC | | | | |
| M6. Exécution de code arbitraire sur la TOE | | | IDC | IDC | |
| M7. Accès illégitime à la configuration de la TOE | | | IC | | C |
| M8. Accès illégitime aux fichiers de journalisation de la TOE | | | | IC | |
| M9. Mise à jour illégitime de la TOE | D | | IDC | IDC | ID |

Tableau 2 - Couverture des biens sensibles par les menaces

## Menaces et fonctions de sécurité

Les tableaux suivants résument la couverture des menaces par les fonctions de sécurités.

F1. Filtrage des données http
F2. Sécurité des cookies
F3. Sécurité des données XML
F4. Sécurité des données JSON
F5. Terminaison TLS
F6. Validation de certificats clients
F7. Authentification des administrateurs
F8. Durcissement du socle de l'appliance
F9. Politique et sécurisation des mises à jour
F10. Stockage sécurisé des clés de chiffrement

| | F1 | F2 | F3 | F4 | F5 | F6 | F7 | F8 | F9 | F10 |
|---|---|---|---|---|---|---|---|---|---|---|
| M1. Exécution de code arbitraire par l'application ou le serveur web | ✓ | | ✓ | ✓ | | | | | ✓ | |
| M2. Exécution de code arbitraire sur le navigateur web des utilisateurs | ✓ | | ✓ | ✓ | | | | | ✓ | |
| M3. Usurpation de l'identité d'un utilisateur de l'application | | ✓ | ✓ | ✓ | ✓ | ✓ | | | ✓ | |
| M4. Déni de service sur les moteurs de filtrage de la TOE | ✓ | | | | | | | | ✓ | |
| M5. Contournement des moteurs de filtrage de la TOE | ✓ | | | | | | | ✓ | ✓ | |
| M6. Exécution de code arbitraire sur la TOE | ✓ | | ✓ | ✓ | | | | ✓ | ✓ | |
| M7. Accès illégitime à la configuration de la TOE | ✓ | | ✓ | ✓ | | | ✓ | ✓ | | ✓ |
| M8. Accès illégitime aux fichiers de journalisation de la TOE | ✓ | | ✓ | ✓ | | | ✓ | ✓ | | |
| M9. Mise à jour illégitime de la TOE | | | | | | | | | ✓ | |

Tableau 3 - Couverture des menaces par les fonctions de sécurités