Démarqué en NON PROTÉGÉ
par décision n°15699/ANSSI/SDE/ST/LAM
du 18 juillet 2018

Documentation CLIP

0001

Description générale du système CLIP

Version	Date	Auteur	Commentaires
1.0	13/04/2010	Vincent Strubel	Version initiale, à jour pour CLIP version 04.01.01.

Table des matières

Introduction

CLIP est un système d'exploitation sécurisé développé principalement par l'Agence Nationale de la Sécurité des Systèmes d'Information (ANSSI), adapté aussi bien à des cadres d'emploi mononiveau que multiniveau. Le présent document constitue une synthèse générale du fonctionnement de ce système et des possibilités de mise en oeuvre de celui-ci. Il a également vocation à servir d'introduction à l'ensemble de la documentation technique CLIP, et renvoie à ce titre à des documents plus détaillés, généralement classifiés, tels que listés dans l'Annexe A.

1 Présentation générale

1.1 Historique

Le projet CLIP-RM (CLIent Privilégié en Réseaux de Messagerie) a été lancé initialement à l'été 2005 comme une étude interne à la Sous-Direction Scientifique de la Direction Centrale de la Sécurité des Systèmes d'Information (DCSSI). Cette étude interne s'est concrétisée par la réalisation d'une maquette de poste client biniveau, permettant l'accès simultané, sur un même poste de travail et dans deux compartiments logiciels isolés, à deux réseaux de messagerie, l'un sensible (Réseau de Messagerie Haut, RM_H) et l'autre non (Réseau de Messagerie Basse, RM_B). Cette maquette initiale était réalisée sur la base d'un système d'exploitation FreeBSD 4.8. Décidée début 2006, la réalisation d'un prototype plus avancé reprenant les principes de conception de cette maquette – toujours dans la configuration biniveau RM_B et RM_H – s'est traduite par un nouveau développement du socle CLIP sur la base d'un système Linux 2.6, puis par l'ajout progressif des fonctionnalités de mise à jour, d'administration, d'audit et de gestion des supports amovibles, aussi bien par des développements internes à la DCSSI que sous la forme de développements complémentaires au titre d'un contrat de 2 ans notifié à EADS-DS en Décembre 2006. Ce dernier contrat à également conduit à la mise en place d'un réseau de développement CLIP, homologué Confidentiel Défense – Spécial France, au développement d'une configuration passerelle du système CLIP (*CLIP-GTW,* cf. 3.3), ainsi qu'à la conduite d'une évaluation de sécurité (selon les Critères Communs version 3.1, au niveau EAL2+, cf. [CC]) de la version 3 du système CLIP-RM[1] pour **poste client biniveau**. Au terme de cette évaluation, le système CLIP-RM, en version 03.01.03, a **obtenu une certification Critères Communs** le 28 avril 2009 ([CERTIF]), et une **qualification standard** de la DCSSI le 11 mai 2009 ([QUALIF]), le déclarant apte notamment à protéger des informations de niveau *Diffusion Restreinte*, *Diffusion Restreinte OTAN* ou *Restreint UE* (au niveau RM_H) sur Internet (au niveau RM_B). La version majeure 3 a par ailleurs été déployée à partir de fin 2008 dans cette même configuration biniveau au sein d'un **réseau de test**, permettant l'accès simultané à Internet et à un réseau DR-SF. Ce réseau de test a vu sa taille augmenter progressivement jusqu'à dépasser fin 2009 les 100 utilisateurs, répartis entre le SGDSN et plusieurs ministères et administrations. Parallèlement à ces déploiement, une distribution CLIP mononiveau, *CLIP-single*, avait également été mise au point par la DCSSI à titre de prototype, mais cette version n'a pas été pérennisée et a finalement été remplacée par l'ajout de configurations mononiveau dans la distribution CLIP-RM (voir plus bas et section suivante).

Les retours utilisateurs issus de ce réseau de test, ainsi que la prise en compte de nouveau besoins, a conduit courant 2009 au développement par la DCSSI d'une nouvelle version majeure 4 du système CLIP, apportant de nombreuses évolutions, dont principalement :

- la réécriture complète des mécanismes de mise à jour, afin d'optimiser les traitements, de corriger des problèmes de fiabilité récurrents, et d'apporter une souplesse de gestion accrue (avec notamment le support de paquetages optionnels, dont l'installation est réalisée à la

[1] Les versions majeures 1 et 2 du système CLIP correspondent à des versions intermédiaires de développement, qui n'ont jamais été déployées. Plus précisément, la version 1 correspond à celle remise à l'industriel en entrée du contrat (Décembre 2006), et la version 2 à celle livrée par l'industriel titulaire au terme de la phase de développement. La version majeure 3, soumise à l'évaluation, correspond à la version 2 augmentée de développements complémentaires réalisés par la DCSSI, notamment en ce qui concerne la mise en oeuvre d'IPsec. La maquette initiale sur base FreeBSD n'est pas comptabilisée dans ce système de numérotation.

demande de l'administrateur de chaque poste) ;

- de nouveaux moyens d'accéder au réseau (filaire DHCP, *Wifi*, et ultérieurement liens téléphoniques mobiles 3G), une plus grande souplesse dans la gestion de la configuration réseau (gestion de plusieurs profils de connexion, changement de profil à la volée), et une gestion plus robuste des tunnels IPsec ;
- de nouvelle possibilités pour la gestion des supports amovibles : supports USB non initialisés ou signés mais non chiffrés, CD-ROM et DVD-ROM, utilisation de supports amovibles pour la mise à disposition de mises à jour, l'import de fichiers de configuration ou l'export des journaux ;
- des mécanismes de diodes chiffrante et montante pour transférer des fichiers entre les compartiments RM_B et RM_H dans une configuration biniveau (cf. 2.2.1) ;
- l'utilisation d'une configuration noyau modulaire, de sorte qu'une même distribution binaire puisse être installée sur tous les types de matériels supportés (cf. 5.3) ;
- la mise en oeuvre d'interfaces graphiques pour toutes les tâches d'administration et d'audit du poste (tâches qui devaient être réalisées en ligne de commande sur CLIP version 3) ;
- le support de nouveaux périphériques matériels : cartes réseau *Wifi* et 3G, cartes son, impressions locales (USB) et distantes (imprimantes réseau, serveurs CUPS).

Cette version 4 du système CLIP, disponible depuis l'été 2009, est désormais la seule déployée au sein du réseau de test, et des autres déploiements en cours ou à venir. Le développement se poursuit courant 2010 sur cette version majeure 4, avec de nombreuses révisions mineures (4.0.N), et une révision intermédiaire 4.1 apportant principalement :

- une mise à jour majeure de l'environnement logiciel utilisateur (*KDE* version 4 en remplacement de la version 3.5, *Mozilla Thunderbird* en version 3) ;
- une gestion améliorée de l'affichage, avec notamment une accélération matérielle sur certains matériels (cf. 5.1) ;
- le support par la même distribution binaire CLIP-RM (en fonction d'un paramètre choisi lors de l'installation) de configurations mononiveau, en plus de la configuration biniveau déjà supportée. Ces configurations permettent d'avoir une seule cage RM (RM_H ou RM_B) installée sur un poste, dont le bureau est affiché en plein écran dans les sessions utilisateur (cf. 3.2). Cette évolution permet en particulier la suppression définitive de la distribution CLIP-single évoquée plus haut.

Suite à la réorganisation de la DCSSI, le développement du système CLIP se poursuit sous la conduite du Laboratoire Architectures Matérielles et Logicielles de la sous-direction Assistance, Conseil et Expertise de l'Agence Nationale de la Sécurité des Systèmes d'Information (ANSSI/ACE/LAM). Il n'est pas prévu à ce stade de nouvelle version majeure (5), mais uniquement des versions mineures (4.1.N) ou intermédiaires (4.2, etc.).

Remarque 1 : signification des versions CLIP

Les versions du système CLIP sont exprimées sur trois champs : **<majeur>.<intermédiaire>.<mineur>**, *par exemple 4.1.2 (version majeure 4, intermédiaire 1, et mineure 2). La signification attribuée à ces différents champs est la suivante :*

- *le numéro de version **majeure** est incrémenté lors d'évolutions structurelles lourdes, qui ne peuvent être apportées à travers le mécanisme de mise à jour CLIP et **nécessitent donc une réinstallation** des postes ; par exemple, le passage de CLIP 3 à CLIP 4 a nécessité une telle réinstallation, du fait du redéveloppement complet des outils de mise à jour ;*
- *le numéro de version **intermédiaire** est incrémenté lors d'**évolutions fonctionnelles majeures**, pouvant néanmoins être déployées par le mécanisme de mise à jour ; par exemple, le passage de la version 4.0 à 4.1 apporte de nouvelles options d'installation, mais ne nécessite pas la réinstallation des postes CLIP-RM biniveau initialement installés en version 4.0 ;*
- *le numéro de version **mineure** est incrémenté pour toute autre mise à jour, par exemple évolution fonctionnelle mineure, mise à jour de sécurité ou correction de bogues.*

On notera par ailleurs qu'un poste CLIP correspond en général à plusieurs numéros de version décorrélés, correspondant à autant de « configurations » (cf. 4.1) installées sur le poste. On comptera au moins deux versions par poste, une pour les paquetages essentiels du socle CLIP et une pour les paquetages secondaires de ce dernier. S'y ajoutent, sur un poste de type CLIP-RM, deux versions pour les paquetages essentiels et secondaires, respectivement, au sein de chaque compartiment logiciel (cage) RM (RM_H et/ou RM_B). Dans tous les cas, le terme de « version CLIP », sans précision supplémentaire, désigne dans l'ensemble de la documentation CLIP le numéro de version des paquetages essentiels du socle CLIP.

1.2 Distributions CLIP

Le système CLIP est une distribution Linux durcie, développée par l'ANSSI à partir de la distribution publique *Gentoo* (cf. [Gentoo]). Il hérite de cette dernière le concept de *méta-distribution* : une unique distribution source, classifiée Confidentiel Défense – Spécial France, et disponible uniquement au sein du réseau de développement CLIP, permet la génération, en fonction de profils de compilation différents, de plusieurs distributions binaires plus classiques, qui sont celles effectivement déployées sur les postes CLIP. Une distribution source correspond à un ensemble de paquetages « sources », eux-mêmes combinaisons d'une ou plusieurs archives de code source et d'un script d'installation. L'installation d'un paquetage source consiste à décompresser les archives sources, à les compiler avec des options définies dans le script d'installation, et à installer les binaires résultant de cette compilation. A contrario, une distribution binaire est directement constituée de paquetages binaires déjà compilés, qui peuvent être installés et utilisés sur un poste sans compilation locale. Il s'agit du principe de fonctionnement normal de la plupart des distributions Linux grand public. Pour le développement de CLIP, les outils de compilation et d'installation de paquetages sources issus de *Gentoo* ont été adaptés, de manière à permettre non seulement l'installation locale de paquetages sources, mais également la production, à partir de ces mêmes paquetages, de paquetages binaires au format *Debian* (paquetages *.deb*). Ces paquetages constituent ensuite une distribution binaire CLIP, apte à être déployée sur un type de poste. Les distributions binaires CLIP, à la différence de la distribution source Confidentiel Défense, sont normalement non classifiées, et manipulées comme des données Diffusion Restreinte. Une exception possible à cette règle est liée à l'inclusion dans certaines distributions spécifiques de bibliothèques cryptographiques classifiées, même sous leur forme binaire.

Une des spécificités de la distribution *Gentoo* est la gestion fine des options de compilation, qui permet, à partir d'un profil de compilation général (drapeaux *USE* essentiellement), d'activer ou non certaines options de compilation ou d'installation pour chaque paquetage source. On peut ainsi par exemple activer ou désactiver le support de la localisation linguistique dans les paquetages compilés (drapeau *nls* pour *national language support*), ou encore activer ou désactiver la génération d'interfaces graphiques dans les paquetages pour lesquels ces interfaces sont optionnelles (drapeau *X*). Dans CLIP, des profils de compilation différents sont utilisés pour **générer plusieurs distributions binaires** distinctes, qui diffèrent par les options de compilation proprement dites, mais également par les types de fichiers installés (fichiers de configuration par exemple), ainsi que par les dépendances effectives entre paquetages binaires. Ces distributions sont elles-mêmes adaptées à des déploiements sur des types de postes différents, par exemple passerelles ou clients. A ce stade, le système CLIP est distribué sous la forme de deux distributions binaires principales :

- **CLIP-RM** : distribution pour postes clients, mono- ou biniveau (cf. section 3.2),
- **CLIP-GTW** : distribution pour passerelles de chiffrement IPsec (cf. section 3.3).

Historiquement, une troisième distribution binaire a également existé : **CLIP-single**, distribution pour poste client mononiveau uniquement. Cette distribution, qui trouvait sa raison d'être dans le fait que la distribution CLIP-RM ne supportait pas à l'époque de configurations mononiveau, n'est désormais plus maintenue. Une configuration mononiveau de la distribution CLIP-RM permet de remplir le même objectif fonctionnel, avec une sécurité améliorée, et la suppression d'une troisième distribution binaire permet de réduire les charges de maintien en condition des deux autres.

Outre cette déclinaison principale en deux distributions binaires, chacune de celles-ci peut faire, là encore à travers le profil de compilation, l'objet d'adaptations mineures pour un déploiement spécifique, consistant par exemple à modifier les fichiers d'aide à la prise en main du poste en fonction de son environnement de déploiement, à ajouter ou supprimer des paquetages à l'installation par défaut, ou à afficher des bandeaux d'avertissement spécifiques au niveau de sensibilité de l'information manipulée. Cependant, la politique de développement de CLIP vise à maintenir chaque distribution binaire aussi générique que possible, et à limiter les déclinaisons spécifiques à chaque déploiement à quelques paquetages spécifiques, afin de limiter la charge de maintien en condition opérationnelle et les risques d'erreurs ou de bogues spécifiques à une déclinaison donnée. En particulier, chaque distribution binaire CLIP peut normalement être installée indifféremment sur tout type de matériel supporté (cf. 5.1), et peut être au moins partiellement adaptée à un déploiement donné en fonction d'un certain nombre de paramètres d'installation (cf. 5.3).

2 Principaux mécanismes de sécurité

La sécurité d'un poste CLIP repose sur la combinaison de plusieurs mécanismes de sécurité complémentaires. Certains de ceux-ci résultent de l'intégration ou de la configuration de mécanismes publics (primitives déjà présentes dans le noyau Linux, ou patchs tierce partie), tandis que d'autres ont fait l'objet d'un développement spécifique dans le cadre du projet CLIP. Ces différents mécanismes se complètent mutuellement dans une optique de défense en profondeur. L'architecture de sécurité CLIP est décrite de manière détaillée dans le document de référence [CLIP_1002].

2.1 Choix architecturaux

2.1.1 Mécanismes de cloisonnement logiciel

Le système CLIP intègre une primitive de cloisonnement logiciel permettant de découper le système en plusieurs compartiments isolés. Les processus exécutés dans un compartiment logiciel donné n'ont par défaut aucun moyen d'interagir avec ceux des autres compartiments (ni même de voir l'existence de ces autres compartiments). Ces processus n'ont par ailleurs en aucun cas la possibilité de mettre en oeuvre des privilèges de niveau système (permettant de modifier la configuration du cloisonnement, ou des ressources partagées par tous les compartiments), et peuvent être traités différemment des processus des autres compartiments pour tout ce qui concerne l'accès au réseau. De ce fait, un attaquant qui aurait, suite à l'exploitation d'une ou plusieurs vulnérabilités logicielles, pris le contrôle complet d'un compartiment du système ne pourrait pas pour autant porter atteinte aux autres compartiments.

Cette primitive de cloisonnement est utilisée à plusieurs fins au sein du système CLIP. Elle assure en premier lieu le cloisonnement multiniveau au sein d'un système CLIP-RM biniveau, en interdisant la fuite d'information du niveau haut RM_H vers le niveau bas RM_B (et vice-versa le cas échéant), et les attaques en intégrité depuis le niveau bas vers le niveau haut ou vers le socle de confiance du système. Dans ce but, l'accès à chacun des deux niveaux d'information est circonscrit à un compartiment logiciel dédié, dit **cage RM**. Un système CLIP-RM biniveau intègre donc deux cages RM : une cage RM_H et une cage RM_B. Par extension, une configuration mononiveau CLIP-RM intègre une seule cage RM, RM_H ou RM_B, qui se trouve isolée du reste du système, de manière à protéger ce dernier en cas de compromission de la cage.

D'autre part, et indépendamment de ces considérations multiniveau, le système CLIP met également en oeuvre le cloisonnement logiciel à des fins de cloisonnement des rôles au sein du système, à des fins de défense en profondeur. Ainsi, le socle CLIP (c'est-à-dire le système hors éventuelles cages RM) est également partitionné en plusieurs compartiments logiciels, correspondant aux différents rôles du système (par exemple administration, mise à jour, affichage). Ce partitionnement assure la non propagation au reste du système d'une éventuelle compromission d'un de ces rôles, et réalise par ailleurs une réduction systématique des privilèges accordés aux différents rôles. Les différents compartiments logiciels du socle CLIP sont désignés sous le terme générique de **cages CLIP**. La composante (de taille réduite) du système qui reste hors de toute cage CLIP est désignée sous le nom de **coeur CLIP.**

Par ailleurs, une deuxième primitive de cloisonnement, plus légère, est employée au sein des cages RM

pour y réaliser une séparation similaires entre rôles (séparation qui ne peut être réalisée avec la première primitive de cloisonnement « lourd », laquelle ne permet pas de sous-divisions d'un compartiment logiciel), et le cas échéant d'affiner le cloisonnement entre rôles au sein des cages CLIP. Ces compartiments logiciels légers sont appelés **vues** dans l'ensemble de la documentation CLIP.

On notera que le coeur CLIP peut paramétrer des mécanismes spécifiques d'échange d'information, sous son contrôle, entre deux compartiments logiciels. C'est sur ce principe que se fonde la mise en oeuvre des diodes cryptographique et montante (cf. 2.2.1).

Le mécanisme de cloisonnement mis en oeuvre au sein de CLIP est décrit de manière plus détaillée dans les documents de référence [CLIP_1202] (mécanisme de cloisonnement lourd) et [CLIP_1203] (sécurité du mécanisme de cloisonnement léger). La construction des différents compartiments fait quant à elle l'objet des documents [CLIP_1304] et [CLIP_1401].

2.1.2 Durcissement générique de la gestion mémoire

Plusieurs mécanismes de durcissement générique de la gestion de la mémoire sont mis en oeuvre de manière quasi-systématique au sein de CLIP : interdiction de l'exécution des zones mémoire de données, mise en oeuvre de « canaris » pour détecter les débordements mémoire, randomisation des adresses, contrôle de certains déréférencement de pointeurs, protection en lecture seule de certaines zones mémoire, etc. Ces mesures visent à contrer ou du moins complexifier l'exploitation d'une vaste catégorie de vulnérabilités logicielles, notamment les débordements de tampon et les déréférencements de pointeurs nuls. Bien qu'elles ne soient pas totalement incontournables, elles offrent une **première ligne de défense** contre de telles attaques, et contribuent à ralentir l'exploitation d'éventuelles vulnérabilités le temps que le mécanisme de mise à jour les corrige.

Ces différents mécanismes de durcissement sont détaillés dans les documents de référence [CLIP_1203] (mesures de durcissement mises en oeuvre par le noyau) et [CLIP_1101] (durcissement à la compilation).

2.1.3 Minimalisation et réduction de privilèges

De même que l'isolation et le durcissement de la gestion mémoire présentés ci-dessus, un principe de minimalisation est mis en oeuvre systématiquement dans le socle de sécurité (coeur et cages CLIP) d'un système CLIP. Il s'agit en premier lieu de **réduire le nombre et les fonctionnalités des composants logiciels installés** au minimum nécessaire au bon fonctionnement du système, au besoin en remplaçant des composants standards trop complexes par des implémentations spécifiques plus simples. En effet, tout ce qui sort de ce cadre ne peut servir qu'à un attaquant, soit en tant qu'interface supplémentaire exposée à une attaque potentielle, soit comme outils à mettre en oeuvre après une attaque réussie. Cette approche de restriction du système n'est en revanche pas mise en oeuvre de manière aussi poussée dans les compartiments d'utilisation finale (cages RM typiquement), dans la mesure où, d'une part, il n'est pas possible de définir précisément a priori l'ensemble des fonctionnalités nécessaires aux utilisateurs, et d'autre part, le faible niveau de privilège de ces compartiments réduit dans tous les cas la portée d'une attaque éventuelle (ces compartiments sont essentiellement considérés comme « perdables » dans le modèle de sécurité global du système).

Par ailleurs, un même principe de minimalisation est recherché dans l'**attribution des privilèges** au sein du système. Ainsi, les démarches de réduction ou de séparation de privilège sont largement mises

en oeuvre au sein du coeur CLIP, ou pour la réalisation d'opérations privilégiées par ce même coeur CLIP à la demande des cages CLIP. De plus, un mécanisme spécifique à CLIP permet de **réduire systématiquement les privilèges attribués même au « superutilisateur »** (*root*) au sein du coeur CLIP, et de n'attribuer certains de ces privilèges que dans le cadre de chemins d'exécution de confiance. Là encore, l'effet recherché consiste à limiter la portée d'une attaque réussie, même au sein des composants parmi les plus critiques du système. De plus, certains privilèges complémentaires sont introduits par rapport au modèle standard de Linux, par exemple concernant l'accès au réseau, et sont systématiquement contrôlés, y compris dans les compartiments utilisateur.

La même approche de limitation des privilèges se retrouve à un plus haut niveau dans la **définition des rôles interactifs au sein du système**. Les rôles d'utilisation, d'administration et d'audit sont clairement identifiés et séparés (même si dans la pratique ils peuvent sans risque être tous conférés simultanément à un seul et même utilisateur physique), et surtout ne disposent que de privilèges très restreints sur le système. Ainsi, ni l'administrateur ni l'auditeur ne peuvent, par maladresse ou malveillance, porter atteinte aux propriétés de confidentialité (confidentialité des données de chaque utilisateur, cloisonnement multiniveau de l'information) ou d'intégrité du système. Seule la disponibilité du poste peut être remise en cause par un administrateur malveillant (par exemple en définissant des paramètres réseau incorrects).

Les mécanismes spécifiques de gestion des privilèges au sein de CLIP sont décrits dans le document de référence [CLIP_1201]. La répartition des prérogatives par rôles est liée à la construction des cages CLIP, détaillée dans le document [CLIP_1304].

2.1.4 Principe « W^X »

Le principe dit « W^X » consiste à rendre **exclusifs les accès en écriture et en exécution**. Il est généralement appliqué à la **mémoire**, en rendant simultanément non inscriptibles les zones de code en mémoire, et non exécutables les zones de données accessibles en écriture. Une telle approche est adoptée par plusieurs systèmes d'exploitation sur étagère, comme par exemple *OpenBSD* ou des variantes durcies de Linux. Elle est également mise en oeuvre aussi systématiquement que possible dans CLIP, comme décrit en section 2.1.2.

La spécificité du système CLIP consiste à **étendre ce principe d'exclusivité au système de fichiers**. Ainsi, la gestion des droits d'accès aux fichiers garantit sous CLIP[2], qu'aucun fichier ne peut être créé ou modifié puis exécuté, et ce quelque soit le niveau de privilège de l'utilisateur tentant de réaliser ces opérations. Cette mesure **interdit la pérennisation d'une attaque** réussie : un attaquant ayant réussi à prendre le contrôle d'une application en mémoire, suite à l'exploitation d'une vulnérabilité, cherchera généralement à installer sur le disque des exécutables malveillants lui permettant de renouveler plus facilement son attaque, et ce même si la vulnérabilité lui ayant permis la prise de contrôle initiale est par la suite corrigée par une mise à jour. Une telle démarche ne pourra pas fonctionner sous CLIP, car les exécutables malveillants que l'attaquant aura été mesure d'installer sur le disque (dans des zones où il dispose d'un accès en écriture) ne pourront en aucun cas être exécutés.

[2] En dehors des compartiments dédiés aux mises à jour : coeur CLIP, cage et vues de mise à jour, qui ne peuvent pas, par construction, respecter ce principe.

2.1.5 Mises à jour sécurisées

La gestion des mises à jour est un aspect critique de la sécurité d'un système d'exploitation. CLIP dispose à ce titre d'un système de mise à jour avancé, qui s'appuie largement sur les autres mécanismes de sécurité du poste. Cette gestion des mises à jour est décrite plus en détail en section 4, ainsi que dans le document de référence [CLIP_1305]. On en rappellera simplement ici les principaux avantages en termes de sécurité :

- un **fonctionnement entièrement automatique**, sans interaction avec l'utilisateur ou un administrateur : les mises à jour sont appliquées automatiquement dans un délai très court après leur mise à disposition, sans risque d'oubli ou d'erreur de la part de l'utilisateur ;
- une **authentification robuste des mises à jour** : chaque paquetage de mise à jour est doublement signé (cf. 4.2), en utilisant les mécanismes cryptographiques gouvernementaux offerts par la bibliothèque CCSD (cf. [CCSD]) ; ces signatures sont systématiquement vérifiées avant l'installation de mises à jour ;
- une **protection des flux réseau de mise à jour** : lorsque les mises à jour sont téléchargées à travers le réseau, ces flux sont protégés par un tunnel IPsec spécifique (cf. 2.2.2), dédié aux mises à jour ; les flux font par ailleurs l'objet d'une seconde protection au niveau applicatif (TLS) ;
- le téléchargement et l'application des mises à jour se font dans un compartiment logiciel dédié (cf. 2.1.1) et sont ainsi **isolés du reste du système et protégés en intégrité** ;
- les l**ogiciels critiques** utilisés pour la vérification et l'application des mises à jour ne sont eux-mêmes mis à jour qu'au **redémarrage**, dans une configuration particulièrement intègre du système.

2.1.6 Cloisonnement graphique

Le **serveur d'affichage** (serveur X11) utilisé dans CLIP intègre des modifications spécifiques, lui permettant de gérer **plusieurs domaines cloisonnés**. Plus précisément, outre le domaine par défaut (domaine privilégié, correspondant au domaine habituel d'un serveur X11), les clients graphiques peuvent être cloisonnés individuellement dans plusieurs domaines non privilégiés. Un client appartenant à un domaine d'affichage non privilégié n'a pas accès aux ressources et extensions sensibles du serveur d'affichage, et ne peut pas interagir au sens du protocole d'affichage (c'est-à-dire par exemple par des copier-coller, des captures d'écran, ou l'envoi d'événements) avec les clients d'autres domaines d'affichage, privilégiés ou non. Cette division en domaines d'affichage permet de **prolonger sur le plan du protocole d'affichage le cloisonnement système** décrit en section 2.1.1.

Parmi les domaines non privilégiés, les deux premiers sont désignés domaine haut et domaine bas, et affectés respectivement à l'affichage du bureau RM_H et RM_B, dans une configuration CLIP-RM multiniveau (cf. 3.2.1). Le domaine privilégié est quant à lui réservé aux clients exécutés dans le socle CLIP (cages USER, ADMIN et AUDIT, cf. 3.1). Le gestionnaire de fenêtre et la barre de tâches de la session graphique CLIP sont également adaptés, de manière à afficher clairement le domaine d'appartenance de chaque fenêtre, par une signalétique de couleur (bleu pour le domaine privilégié, rouge pour le domaine haut, vert pour le domaine bas, jaune pour les autres domaines) et en toutes lettres, dans le préfixe du nom de fenêtre (« CLIP », « RM_H », « RM_B » et « ???? »).

Le fonctionnement du cloisonnement graphique dans CLIP est détaillé dans le document de référence [CLIP_1303].

2.1.7 Journalisation intègre des actions

Le système CLIP assure une collecte centralisée et continue des journaux d'événements générés par les différents compartiments logiciels (coeur, cages CLIP et cages RM éventuelles). Ces journaux sont ensuite consultables par un utilisateur du rôle auditeur, ou d'un autre rôle disposant de prérogatives similaires (cf. 6.1). CLIP assure une protection améliorée de l'intégrité de ces journaux par rapport à un système d'exploitation classique. En particulier :

- La **collecte des journaux est isolée dans un compartiment spécifique**, et ainsi protégée contre des interactions non maîtrisées avec d'autres applications. Cette isolation assure par ailleurs la protection en confidentialité des journaux par rapport aux autres applications du système.
- Le démon de collecte des journaux dispose de privilèges spécifiques, qui **interdisent sa désactivation avant l'arrêt du système** par tout autre processus du système (même très privilégié).
- Les journaux une fois écrits sur le disque sont **protégés contre toute modification** (autre que l'ajout de nouveaux éléments) par n'importe quel processus du système. Même le rôle d'auditeur n'est pas en mesure de modifier ou supprimer les journaux. L'archivage, et au bout d'un certain temps la suppression, des anciens fichiers de journaux est réalisé de manière totalement automatique, uniquement au démarrage du système.

Par ailleurs, le système CLIP **journalise un certain nombre d'événements significatifs** au plan de la sécurité, qui ne sont pas normalement journalisés sur un poste Linux, en particulier la terminaison anormale des processus, et toutes les alertes relevées par les différents mécanismes de sécurité spécifiques à CLIP.

2.2 Mécanismes cryptographiques

2.2.1 Diode cryptographique et diode montante

Les diodes sont des fonctionnalités spécifiques aux postes CLIP multiniveaux, dans des configurations CLIP-RM. Il s'agit de services fournis par le socle CLIP, permettant des **interactions contrôlées entre les deux cages RM, RM_B et RM_H**, de manière à autoriser certains transferts de fichiers de l'une vers l'autre sans violer les **contraintes de confidentialité de la politique de sécurité multiniveau**, c'est à dire sans autoriser la descente d'information sensible du niveau haut vers le niveau bas. Plus concrètement, un unique service du socle CLIP, disposant d'interfaces spécifiques au sein de chacune des deux cages RM, assure les deux fonctionnalités, de diode montante d'une part et cryptographique d'autre part.

Le fonctionnement détaillé de ce service de diode est décrit dans le document de référence [CLIP_1307].

Diode montante

La diode montante permet le **transfert de fichiers en clair** (sans traitement cryptographique) du niveau bas (RM_B) vers le niveau haut (RM_H), et **dans ce sens uniquement**. Elle permet ainsi l'import de documents dans le niveau haut depuis une source extérieure de niveau moindre (par exemple import dans le compartiment haut Diffusion Restreinte de documents téléchargés sur Internet au niveau bas, dans le cas d'un poste du réseau de test CLIP), sans pour autant permettre la compromission de documents du niveau haut. Elle réalise par ailleurs une **journalisation de confiance**[3] **de chaque transfert**, en décrivant notamment l'utilisateur ayant réalisé le transfert et le nom, la taille et l'empreinte cryptographique du fichier transféré.

Remarque 2 : diode montante et protection en intégrité

La diode montante impose par construction le respect des contraintes multiniveau en confidentialité. En revanche, elle n'assure aucun traitement visant à protéger l'intégrité du niveau haut vis-à-vis du niveau bas, en s'assurant de l'innocuité des fichiers transférés. Le durcissement générique d'un poste CLIP mitige cette menace en limitant très fortement les possibilités d'atteinte à l'intégrité locale de la cage RM_H. En revanche, il est important de la garder à l'esprit dès lors que le poste CLIP peut être amené à transférer, au niveau haut, des fichiers à des postes moins sécurisés. Dans ce cas, une analyse (vérification de format, antivirus, dépollution éventuelle) des fichiers transférés pourra être souhaitable avant ou après leur transfert.

Diode cryptographique

La diode cryptographique permet des **transferts dans les deux sens**, aussi bien depuis RM_H vers RM_B que depuis RM_B vers RM_H, mais en réalisant une **opération de chiffrement et de signature en coupure** (chiffrement et signature du niveau haut vers le niveau bas, vérification de signature et déchiffrement du niveau bas vers le niveau haut) d'un niveau suffisant pour éviter la compromission des informations de niveau haut. Ainsi des informations de niveau de sensibilité haut, manipulées en clair (« rouge ») dans RM_H, peuvent être transmises sous forme « noircie », de niveau de sensibilité bas, dans RM_B. Les formats de fichiers et traitements cryptographiques réalisés par cette diode sont identiques à ceux mis en oeuvre pour la **création d'archives chiffrées signées par le logiciel de chiffrement ACID CRYPTOFILER** (version 7) en environnement *Microsoft Windows*. Ainsi, une archive chiffrée-signée produite par la diode cryptographique CLIP pourra être déchiffrée – sous réserve de compatibilité des logiques cryptographiques et des clés employées – par le logiciel ACID CRYPTOFILER, et vice-versa. On gardera cependant à l'esprit que la diode cryptographique n'utilise pas ACID CRYPTOFILER lui-même, mais consiste bien en un développement natif pour CLIP, exploitant au mieux les primitives de sécurité du système, tout en restant interopérable avec ACID CRYPTOFILER. Tout comme pour la diode montante, les différents chiffrements et déchiffrements sont systématiquement journalisés par le socle CLIP (utilisateur réalisant l'opération, identifiant de clé privée, noms et empreintes cryptographiques des fichiers chiffrés ou déchiffrés, identifiants des clés publiques destinataires dans le cas d'un chiffrement, identifiant de l'émetteur dans le cas d'un déchiffrement).

Le chiffrement d'un ou plusieurs fichiers nécessite le transfert dans la diode depuis RM_H, outre ces fichiers, d'une clé privée ACID v7 (qui servira notamment à signer l'archive), et des clés publiques de

[3] Journalisation entièrement effectuée par le socle CLIP, donc protégée contre d'éventuelles actions malveillantes issues des cages RM.

chacun des destinataires de l'archive. Le déchiffrement nécessite le transfert dans la diode de l'archive chiffrée signée (depuis RM_B) et de la clé privée à utiliser pour déchiffrer (depuis RM_H). Un tel déchiffrement permettra de récupérer, dans RM_H, non seulement le ou les fichiers déchiffrés, mais également (de manière optionnelle) la clé publique ACID v7 du signataire de l'archive. Pour toute opération de chiffrement ou déchiffrement, le socle CLIP interrogera directement l'utilisateur pour lui demander la confirmation de l'opération et la saisie du mot de passe permettant de déverrouiller la clé privée. **Le socle CLIP est donc seul à manipuler les clés « rouges »**. Les clés privées sont effectivement stockées au sein de la cage RM_H, mais uniquement sous forme « noircie » par leur mot de passe.

La diode s'appuie pour les traitements cryptographiques sur la même interface CCSD (cf. [CCSD]) que ACID CRYPTOFILER. Tout comme ce dernier, elle peut utiliser simultanément (pour des opérations de chiffrement successives) **plusieurs bibliothèques cryptographiques** respectant cette interface mais fournissant des primitives cryptographiques différentes, aussi bien civiles que gouvernementales.

2.2.2 Chiffrement IPsec des flux

Le système CLIP peut protéger les flux réseaux à l'aide de transformations IPsec, assurant des propriétés de **confidentialité, d'authenticité et de non-rejeu des paquets émis et reçus**. Une transformation donnée peut être appliquée à **l'ensemble des flux réseau émis et reçus sur toutes les interfaces externes[4] par une cage logicielle** (cf. 2.1.1) donnée, et uniquement à ces flux, assurant ainsi le **prolongement réseau du cloisonnement local**. Les transformations IPsec que peut mettre en oeuvre le poste CLIP utilisent le protocole ESP (confidentialité + authenticité) en mode tunnel, jusqu'à un poste distant de type passerelle (pour un poste local client) ou client (pour un poste local de type passerelle)[5]. Une réencapsulation UDP *Nat-traversal* est activée automatiquement en cas de détection d'une traduction d'adresse (NAT) entre les deux postes extrémités d'un tunnel. Dans ce cas, les paquets effectivement émis sont des paquets UDP de ports source et destination 4500, plutôt que des paquets ESP directement. Les algorithmes cryptographiques associés à ces tunnels sont **normalement les algorithmes gouvernementaux fournis par la bibliothèque CCSD** (cf. [CCSD]) intégrée au noyau CLIP (cf. [CLIP_1205]).

L'établissement des tunnels fait appel à une **négociation dynamique IKEv2**, réalisée par le démon *charon* du projet *strongswan*, spécifiquement modifié dans CLIP de manière à utiliser les mécanismes cryptographiques gouvernementaux fournis par une bibliothèque CCSD pour réaliser la négociation, son authentification et la protection des flux associés. Ces modifications sont détaillées dans le document de référence [CLIP_1503].

Le mécanisme d'authentification des négociations repose sur la mise en oeuvre de **clés privées et publiques au format ACID v7**, produites par un Centre d'Elaboration des Clés (CEC) ACID v7 utilisant une bibliothèque compatible avec celles des postes CLIP. Chaque poste prenant part à une négociation doit disposer d'au moins une clé privée[6], et des clés publiques associées aux clés privées de

[4] Elle ne s'applique en revanche pas à la boucle locale réseau, qui est déjà propre à la cage du fait du mécanisme de cloisonnement logiciel mis en oeuvre.

[5] L'établissement de tunnels IPsec entre deux passerelles CLIP n'a pas été testé et n'est pas supporté par les configurations passerelles CLIP à ce stade, mais ne pose pas de difficulté théorique particulière.

[6] Bien qu'il soit possible d'utiliser des clés privées différentes pour la négociation de chaque tunnel, les distributions CLIP n'utilisent à ce jour qu'une seule et même clé privée pour toutes les négociations.

chacun de ses interlocuteurs.

Alternativement, des déclinaisons spécifiques des distributions binaires CLIP (cf. 1.2) peuvent faire appel à des algorithmes civils pour protéger et authentifier la négociation IKEv2, ainsi que pour la protection des flux IPsec (typiquement, AES et HMAC-SHA256 pour la protection des flux, Diffie-Hellman et authentification par signature RSA pour la négociation).

La négociation IKEv2 génère dynamiquement aussi bien les politiques (SP) que les associations de sécurité (SA), et assure également la détection de terminaison des tunnels (*Dead Peer Detection*), le renouvellement périodique des clés symétriques (IKEv2 et IPsec) et le maintien des éventuelles NAT (*NAT Keep-alive*). Des fonctions de sécurité spécifiques à CLIP assurent par ailleurs qu'**aucun flux réseau ne peut sortir ou entrer en clair** (sans transformation IPsec) dans une cage à laquelle est associé un tunnel IPsec, **même lorsqu'aucune politique de sécurité n'a été mise en place** par le démon IKEv2 pour ces flux.

Des tunnels IPsec sont typiquement mis en oeuvre pour protéger les flux de téléchargement des mises à jour (un tunnel par poste client ou passerelle, sauf passerelle de mise à jour, cf. 3.3), et pour protéger les flux des cages de niveau haut (RM_H) dans les postes CLIP-RM. Cependant, la mise en oeuvre de tels tunnels n'est pas obligatoire : il suffit par exemple de ne pas définir de passerelle de mise à jour sur un poste CLIP pour qu'aucune négociation de tunnel de mise à jour ne soit réalisée. On notera bien que cette absence de négociation ne signifie pas que les flux correspondants sont émis et reçus en clair, mais bien qu'aucun flux à destination ou en provenance de la cage de mise à jour ne pourra être reçu ou émis. L'exemple présenté ici correspond donc au cas type d'un poste pour lequel les mises à jour ne sont pas réalisées en ligne, mais uniquement à partir de supports amovibles.

2.2.3 Chiffrement des données utilisateur sur le disque

Les données des utilisateurs d'un poste CLIP sont systématiquement chiffrées sur le disque. Chaque utilisateur dispose en propre d'une ou plusieurs partitions chiffrées – une pour le socle CLIP et une par cage RM présente sur le système – qui sont automatiquement déchiffrées et montées lors de son ouverture de session, et démontées et chiffrées[7] à la fin de cette même session. Le système garanti que l'utilisateur ne peut écrire – par erreur ou malveillance – nulle part en dehors de sa partition chiffrée et de systèmes de fichiers temporaires en mémoire, qui sont effacés et supprimés en fin de session. Par ailleurs, le *swap* (zone d'échange pour la mémoire virtuelle) du système est également chiffré, à l'aide d'une clé aléatoire générée à chaque démarrage. Cet ensemble de mesures garantit qu'aucune donnée utilisateur n'est jamais écrite sur le disque en clair.

Le chiffrement des partitions utilisateur est réalisé à l'aide de clés secrètes elles-mêmes **protégées par le mot de passe de l'utilisateur**, également utilisé pour l'authentification de ce dernier. Le mécanisme cryptographique de stockage des mots de passe et de déchiffrement des clés secrètes est spécifiquement adapté dans CLIP, de manière à offrir une bonne robustesse face aux risques d'attaque par recherche exhaustive, même hors ligne. Le chiffrement de disque proprement dit est réalisé à l'aide d'algorithmes civils conformes à l'état de l'art.

Ces différents éléments sont décrits plus en détail dans le document de référence [CLIP_1302].

[7] Le chiffrement des données a, à proprement parler, lieu au fur et à mesure de leur écriture, de manière transparente pour l'utilisateur. L'opération effectivement réalisée en fin de session consiste en la suppression de la projection claire (déchiffrée à la volée) de la partition chiffrée.

Remarque 3 : authentification par support externe

Le système CLIP ne supporte pas à ce stade de mécanisme d'authentification reposant sur un support externe de type carte à puce. Il est néanmoins prévu d'intégrer à court terme la possibilité de mettre en oeuvre de tels supports, comme alternative au mot de passe, aussi bien pour réaliser l'authentification de l'utilisateur que le déchiffrement de ses partitions de données.

2.2.4 Gestion des supports amovibles

Le système CLIP permet d'initialiser des supports de stockage amovibles USB (clés, disques, lecteurs de cartes mémoire, etc.), en réalisant un traitement cryptographique de ces supports. Un support USB peut être initialisé de deux manières différentes : **simplement signé**, ou **signé-chiffré**.

L'initialisation d'un support simplement signé consiste en l'ajout de méta-données sur le support, dont un niveau de sécurité (par exemple RM_H), et une **signature cryptographique** portant sur ce niveau et sur des identifiants (fabricant, modèle, numéro de série) du support lui-même. Un support initialisé à un niveau donné ne pourra ensuite être monté sur un poste CLIP que dans une cage associée à ce niveau. La signature d'un support est réalisée à l'aide d'algorithmes asymétriques civils (RSA), avec des clés privées et publiques générées sur le poste (un bi-clé de signature par utilisateur et par niveau CLIP[8], RM_B ou RM_H).

L'initialisation d'un support chiffré reprend la procédure de signature décrite ci-dessus, et y adjoint un **chiffrement du contenu (données) du support**, afin d'offrir une protection en confidentialité de ces données, notamment en cas de perte du support. Le chiffrement est réalisé à l'aide d'algorithmes civils conformes à l'état de l'art, en utilisant une clé symétrique utilisée générée par le poste CLIP au moment de l'initialisation de support. Cette clé est elle-même chiffrée par un algorithme asymétrique (RSA), en utilisant un bi-clé de chiffrement distinct du bi-clé de signature, mais géré de manière similaire à ce dernier (un bi-clé par utilisateur et par niveau, généré par le poste CLIP). La clé symétrique « noircie » est stockée dans les métadonnées du support, et également incluse dans les données couvertes par la signature de ce dernier.

L'initialisation cryptographique des supports permet de **prolonger à ces supports le cloisonnement local** réalisé entre les différentes cages du système. On notera que des supports non initialisés peuvent également être montés dans les différentes cages, au choix de l'utilisateur, mais le cas échéant (en fonction d'une politique de sécurité définie à l'installation du poste) uniquement en lecture seule.

La gestion des supports amovibles sous CLIP est également décrite sur le plan fonctionnel en section 6.2 du présent document, et de manière plus détaillée dans le document de référence [CLIP_1306].

[8] Un support de niveau CLIP pourra être monté dans les différentes cages CLIP, au choix de l'utilisateur et en fonction des droits de montage associés au type de cet utilisateur, cf. 6.1.

3 Configurations CLIP

3.1 Socle commun CLIP

Les deux distributions CLIP actuellement supportées, CLIP-RM et CLIP-GTW, utilisent un socle commun qui diffère peu d'une distribution à l'autre. Ce socle commun est composé du coeur CLIP, comportant le noyau, les bibliothèques de base et les outils de mise à jour, et de cinq cages CLIP :

- La cage de mise à jour, **UPDATE**, qui réalise le téléchargement des mises à jour (CLIP et éventuellement RM), et l'application des mises à jour secondaires CLIP (cf. 4.3).
- La cage d'administration **ADMIN**, accessible par les profils utilisateur disposant des droits d'administration (cf. 6.1), qui permet un accès en écriture aux paramètres modifiables de la configuration du poste CLIP.
- La cage d'audit **AUDIT**, accessible par les profils utilisateur disposant des droits d'audit, qui permet un accès en lecture seule aux fichiers de journaux du poste CLIP.
- La cage d'utilisation **USER**, qui est celle dans laquelle tous les utilisateurs ouvrent initialement leur session. Elle permet selon les profils utilisateur de lancer indirectement des applications dans ADMIN et / ou AUDIT, et le cas échéant de lancer des sessions dans les cages RM (cf. 3.2).
- La cage **X11**, dans laquelle est exécuté le serveur graphique principal du poste, qui réalise l'affichage de la session USER, des clients graphiques ADMIN et AUDIT, et, indirectement, des visionneuses de session dans les éventuelles cages RM (cf. 3.2).

Le coeur CLIP accède uniquement au réseau pour réaliser la configuration initiale ou la reconfiguration de la ou des interfaces réseau du poste (requêtes *dhcp,* configuration *wifi* ou *UMTS*), et pour négocier les tunnels IPsec au profit des cages qui en nécessitent. Parmi les cages CLIP, la cage **UPDATE accède au réseau uniquement à travers un tunnel IPsec** dédié établi avec une passerelle CLIP de type UPDATE (cf. 3.3), pour réaliser le **téléchargement des mises à jour**. Cet tunnel peut également être utilisé pour réaliser une **synchronisation périodique (toutes les heures) de l'horloge** du poste auprès d'un **serveur NTP** situé dans la même zone de service que le serveur HTTPS de mise à disposition des mises à jour. On notera que l'administrateur du poste peut choisir de ne pas définir de passerelle UPDATE, auquel cas aucun tunnel UPDATE ne sera établi et la cage UPDATE n'aura pas accès au réseau (récupération des mises à jour uniquement par support amovible). Par ailleurs, le choix peut être fait à l'installation d'un poste de ne pas installer de clé publique associée à une passerelle UPDATE, ce qui aura pour effet de **désactiver définitivement l'accès réseau de la cage UPDATE.**

Les autres cages CLIP n'ont à ce stade pas accès au réseau. La cage X11 est totalement isolée, tandis que les cages ADMIN, AUDIT et USER partagent leur boucle locale réseau (ce qui permet l'accès au deux premières cages depuis une session USER, l'administration et l'audit se faisant en pratique par des commandes SSH sur la boucle locale). Le support d'un accès réseau distant à ces trois cages, à travers un autre tunnel IPsec, est prévu dans le socle CLIP mais n'a pas été mis en oeuvre à ce stade. Un tel accès permettrait typiquement l'administration et la supervision distantes (en utilisant, à l'intérieur du tunnel, les mêmes commandes SSH que celles déjà employées localement) du poste CLIP.

Le bureau affiché sur l'écran dans une session CLIP comporte aussi une composante commune aux

différentes configurations CLIP, qui est exécutée dans la cage USER. Il s'agit en premier du gestionnaire de fenêtres, qui dessine des bandeaux autour des différents clients graphiques en représentant explicitement leur domaine d'appartenance (cf. 2.1.6), et d'autre part d'une barre de menu, affichée en haut de l'écran, qui contient la barre de tâches permettant de commuter entre les différentes fenêtres et donne accès aux différentes opérations permises par le socle CLIP (gestion des supports amovibles, déconnexion / extinction du poste, action d'administration ou d'audit éventuellement permises par le profil utilisateur, et lancement des sessions RM dans le cas d'un poste CLIP-RM biniveau). Cette barre peut dans certains cas être « repliée » sur elle-même pour occuper moins de place à l'écran (cf. 3.2.2).

3.2 Configurations pour poste client (CLIP-RM)

3.2.1 Configuration nominale

La distribution CLIP-RM est adaptée à un poste client mononiveau ou biniveau, avec une unique interface réseau (filaire ou non). Elle est caractérisée notamment par l'inclusion dans le système de cages logicielles spécifiques (cf. 2.1.1), dites cages RM, qui contiennent chacune un environnement logiciel complet, utilisable pour des besoins de messagerie, bureautique, etc. Deux cages RM peuvent être intégrées dans l'installation d'un système CLIP-RM :

- **RM_B :** cage de niveau bas, permettant la manipulation d'informations dont le niveau de sensibilité est équivalent à celui du réseau physique auquel est connecté le poste (s'il est connecté à un réseau).
- **RM_H :** cage de niveau haut, permettant la manipulation d'informations de niveau de sensibilité plus élevé.

La **principale différence entre ces deux cages logicielles concerne l'accès au réseau**. Les applications de la cage RM_B accèdent directement au réseau auquel est connecté le poste CLIP-RM (par exemple Internet dans le cadre du réseau de test CLIP), avec un filtrage réseau propre à la cage (différent de celui appliqué aux éventuels flux issus du socle), mais sans traitement cryptographique. Au contraire, les applications de la cage RM_H accèdent uniquement à un tunnel IPsec (cf. 2.2.2), établi entre le poste client et une passerelle CLIP (de type passerelle RM_H, comme décrite en section 3.3). Ces applications n'ont donc accès qu'aux serveurs (web, messagerie, annuaire, etc.) du réseau de service protégé par la passerelle RM_H, qui permettent typiquement le traitement d'informations d'un niveau de sensibilité supérieur, par exemple *Diffusion Restreinte* dans le cadre du réseau de test CLIP Cette différence de traitement réseau est liée à l'attribution d'une adresse IP spécifique à chaque cage RM, distincte de l'adresse routable du poste lui-même, qui permettent d'identifier les cages associés aux différents flux dans la pile IP (unique) du poste. Les adresses de cages RM ne sont pas nécessairement routables sur le réseau auquel est connecté le poste, dans la mesure où elles ne sont pas utilisées dans les flux effectivement émis par le poste : les flux émis par la cage RM_B font l'objet d'une transformation de traduction d'adresse source (SNAT) qui leur attribue automatiquement l'adresse principale du poste avant leur émission sur le réseau, tandis que les adresses source et destination des flux émis par RM_H sont masquées par la transformation IPsec qui leur est appliquée (tunnel IPsec, entre l'adresse principale du poste et l'adresse publique de la passerelle).

Une autre différence entre ces deux cages RM tient au fonctionnement de la **diode cryptographique et montante** (cf. 2.2.1): la diode montante est configurée pour permettre le transfert de fichiers en clair de

RM_B vers RM_H, et non l'inverse, et la diode cryptographique réalise un chiffrement de RM_H vers RM_B et un déchiffrement de RM_B vers RM_H. A ces différences près, les deux environnements RM_H et RM_B sont similaires, et fournissent chacun les services applicatifs décrits en section 3.2.3.

Outre ces deux cages RM, un poste CLIP-RM intègre le socle commun CLIP, tel que décrit dans la section précédente. L'environnement local et réseau d'un poste CLIP-RM dans cette configuration « nominale » est résumé dans la Figure 1.

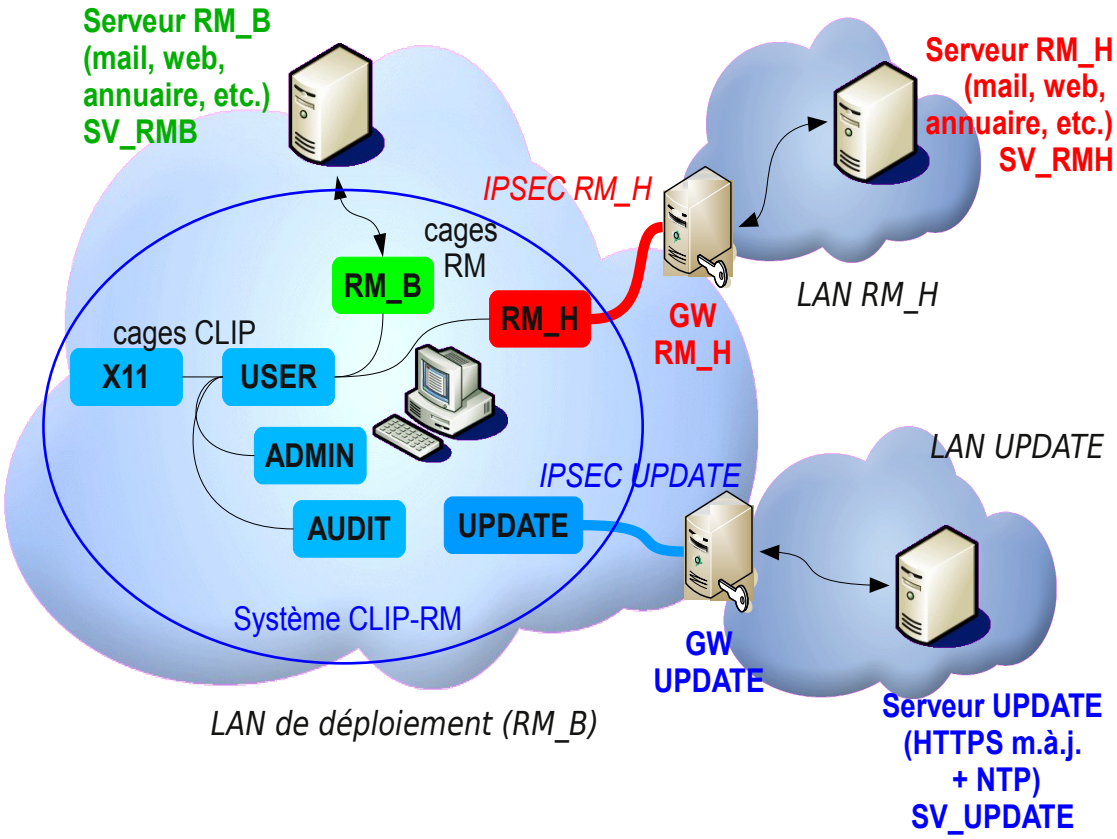

Figure 1: Environnement réseau d'un poste CLIP-RM (à deux cages RM).

Chaque cage RM dispose individuellement de son propre bureau, affiché sur un serveur graphique X11 VNC interne à la cage. Ce bureau virtuel est à son tour affiché dans une fenêtre du bureau principal du poste, qui reste quant à lui géré par la cage USER du socle commun CLIP, et affiché sur le serveur graphique principal exécuté dans la cage X11 du socle. L'affichage des cages RM est ainsi réalisé par

des visionneuses VNC, lancées dans la cage USER. Chacune des deux cages RM se voit associer une visionneuse dédiée, enfermée dans un sous-compartiment (vue, cf. 2.1.1) de USER, et cloisonnée dans un domaine spécifique du serveur d'affichage principal (cf. 2.1.6). Ces visionneuses sont dimensionnées pour occuper chacune tout l'espace d'affichage laissé libre par la barre de menu de confiance (cf. 3.1) et le bandeau du gestionnaire de fenêtres, de telle sorte que seule la visionneuse qui a le focus est visible à un moment donné. La commutation entre visionneuses est possible à travers la barre de tâche de la barre de menu de confiance, ou par un raccourci clavier (*Control* + *Alt* + *Tab*). La Figure 2 schématise le fonctionnement de cet affichage multiniveau.

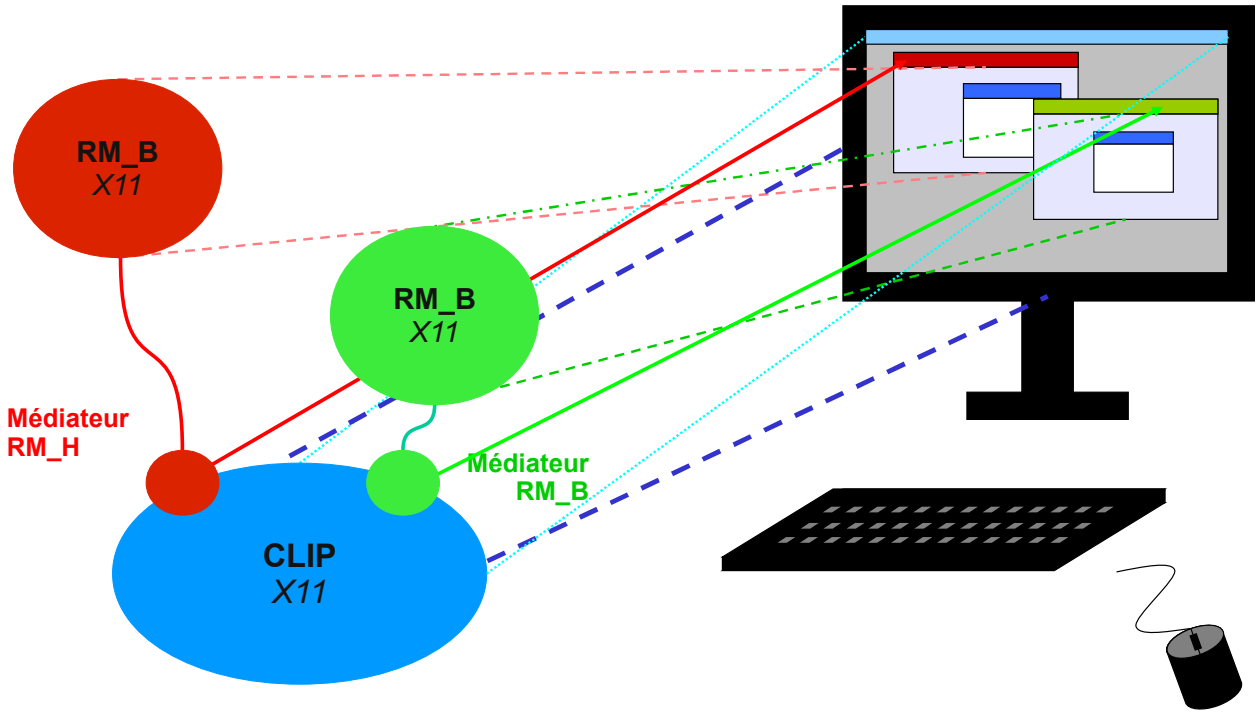

Figure 2: Schéma de principe de l'affichage multiniveau d'un poste CLIP-RM.

Dans la pratique, les fenêtres visionneuses des cages RM occupent chacune l'ensemble de l'espace d'affichage disponible en dehors de la barre de confiance CLIP au sommet de l'écran, et seule la visionneuse qui a le focus est visible à un moment donné.

3.2.2 Variantes d'installation CLIP-RM

La distribution CLIP-RM supporte également des variantes de la configuration nominale, à deux cages connectées au réseau, décrite dans la section précédente.

Configurations hors ligne

Il est possible d'une part, tout comme pour la cage UPDATE (cf. 3.1), de désactiver l'accès réseau de la cage RM_H, temporairement (en ne définissant pas de passerelle RM_H) ou définitivement (en n'incluant pas de clé publique de passerelle RM_H lors de l'installation du poste). La cage RM_H a dans ce cas un fonctionnement hors-ligne, et importe et exporte des documents uniquement à travers des supports amovibles ou la diode cryptographique ou montante (cf. 2.2.1). En combinant une telle configuration RM_H et la désactivation réseau UPDATE décrite en 3.1, on peut obtenir une configuration de poste biniveau hors ligne, interagissant avec l'environnement extérieur uniquement par supports amovibles, qui peut également remplir un service de chiffrement hors ligne de fichiers grâce à la diode cryptographique. Le principe de fonctionnement d'un tel poste est illustré par la Figure 3.

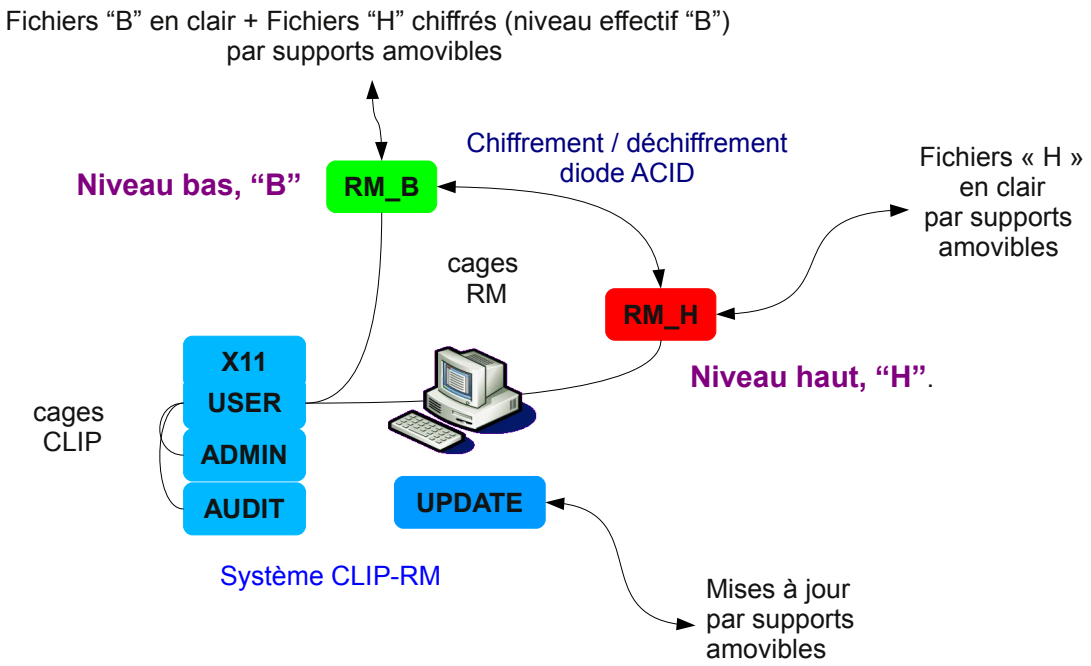

Figure 3: Principe de fonctionnement d'un poste CLIP-RM biniveau hors ligne.

On notera qu'il n'existe pas de moyen équivalent à ceux décrits pour UPDATE et RM_H de désactiver

l'accès réseau de la cage RM_B. Cependant, l'administrateur d'un poste peut configurer le filtrage des flux RM_B par le pare-feu CLIP de manière à interdire effectivement toute connexion réseau.

Configurations mononiveau

Par ailleurs, il est également possible, lors de l'installation d'un poste CLIP-RM (cf. 5.3), de ne le doter que d'une seule des deux cages RM, RM_H ou RM_B. On obtient ainsi une configuration mononiveau, de niveau « haut » ou « bas » selon la cage RM qui a effectivement été installée. La configuration réseau du poste n'est pas modifiée dans son principe par l'installation d'une unique cage RM. Ainsi, si seule la cage RM_H est installée, celle-ci continuera à accéder au réseau uniquement à travers le tunnel IPsec RM_H, établi avec la passerelle correspondante. De même, si seule la cage RM_B est présente sur le système, celle-ci accédera directement au réseau, sans traitement IPsec. En revanche, les diodes cryptographique et montante ne peuvent en l'état pas être mises en oeuvre dans une configuration mononiveau.

Le mécanisme d'affichage du bureau des cages RM est quelque peu adapté dans un contexte mononiveau. Le principe fondamental d'un affichage au sein de la cage RM sur un serveur *Xvnc*, et de l'affichage du bureau de ce serveur par une visionneuse VNC au sein de la session X11 principale, est maintenu. Cependant, lorsqu'une seule visionneuse est utilisée, celle-ci est affichée en plein écran, sans bordure de fenêtre, ce qui place l'utilisateur dans une configuration semblable à l'affichage direct du seul bureau de la cage RM en plein écran. La barre de confiance CLIP (cf. 3.1) est toujours affichée, pour donner accès aux différentes tâches d'administration et d'audit du poste, mais elle est dans ce cas « minimisée » par défaut, réduite à un bouton « + » dans un coin de l'écran, qui permet de « déplier » l'ensemble de la barre. Cette barre est par ailleurs affichée systématiquement (qu'elle soit dans l'état, minimisé ou non) au dessus de la visionneuse de cage RM (dont elle masque potentiellement une partie du bureau), de telle sorte que la cage RM ne puisse pas tromper l'utilisateur en affichant une fausse barre de confiance.

Les autres services du système sont également adaptés automatiquement à la présence d'une unique cage RM, qu'il s'agisse de la gestion des supports amovibles (dont le montage n'est pas proposé dans la cage absente), de celle des mises à jour, ou des outils d'administration (suppression des paramètres réseau spécifiques à la cage absente, pas de création de partitions chiffrées pour cette cage lors de la création de comptes utilisateurs, etc.).

3.2.3 Environnement applicatif des cages RM

L'environnement applicatif des deux cages RM, RM_H et RM_B, est constitué des mêmes paquetages de base (mêmes paquetages obligatoires, cf. 4.1). Ceux-ci offrent dans chaque cage un bureau fondé sur l'environnement KDE (version 4), complété d'applications tierces. Les applicatifs (à interface graphique) ainsi disponibles dans l'installation de base incluent notamment :

- le navigateur web *Mozilla Firefox* (version 3.*), avec *plugin java* ;
- le client de messagerie *Mozilla Thunderbird* (version 3.*) intégrant le gestionnaire de calendrier *Mozilla Lightning* (version 1.0) et le support du chiffrement S/MIME et GPG (*Enigmail* version 1.0);
- la suite bureautique *Open Office* (version 3.2) ;
- l'explorateur de fichiers *Dolphin* (KDE), permettant optionnellement la mise en oeuvre d'une

recherche rapide dans le contenu des fichiers (qui s'appuie sur le service d'indexation *strigi / nepomuk* de l'environnement KDE) ;

- des visionneuses de document pour les formats PDF et *Postscript* (*okular*) et pour les formats d'image (*gwenview*) ;
- les utilitaires classiques de l'environnement KDE : lecture et création d'archives compressées ZIP, RAR, TAR, etc. (*ark* – qui peut également supporter le format 7ZIP après installation d'un paquetage optionnel *p7zip*), calculatrice, gestionnaire d'impressions, éditeur de texte, outils de personnalisation de l'environnement de travail, etc. ;
- un outil de chiffrement et déchiffrement GPG de fichiers (*kgpg*) ;
- un outil de synchronisation de répertoires (*Synkron*), permettant notamment de mettre en place des procédures simples de sauvegarde des données utilisateur sur support amovible.

De manière générale, ces différentes applications se comportent au sein des cages RM d'un poste CLIP exactement comme elles le feraient sur une distribution Linux standard. On notera toutefois que l'installation par l'utilisateur d'extensions téléchargées en ligne pour les applicatifs *Mozilla* (*Firefox* et *Thunderbird*), normalement supportée par ces derniers, n'est pas autorisée[9] afin de ne pas violer la politique de sécurité « W^X » du poste (cf. 2.1.4). L'installation d'un certain nombre d'extensions intégrées à la distribution CLIP reste néanmoins possible pour les utilisateurs disposant d'un profil administrateur ou utilisateur privilégié (cf. 6.1), à travers le mécanisme de paquetages optionnels décrit plus bas. De manière générale, le principe « W^X » bloquera toute forme d'installation directe par l'utilisateur de *plugins* pour quelque application que ce soit. De tels *plugins* devront être intégrés comme paquetages optionnels dans la distribution CLIP pour pouvoir être effectivement déployés sur des postes CLIP.

L'installation de base peut en effet être complétée, indépendamment dans chaque cage RM, par le choix, réalisé par l'administrateur de chaque poste, d'un ensemble de paquetages optionnels parmi ceux autorisés par la configuration. Ces paquetages optionnels (cf. 4.1) comprennent notamment :

- des logiciels clients réseau complémentaires : lecteur de flux *rss* (*akregator*), client de messagerie instantanée *MSN* et *Jabber* (*kopete*), et client IRC (*konversation*) ;
- le lecteur multimédia *kaffeine*, ainsi que le *plugin* associé pour le navigateur *firefox* ;
- différents éditeurs graphiques : *gimp* (dessin et manipulation d'images), *inkscape* (dessin vectoriel SVG), *dia* (éditeur de diagrammes simples) ;
- des outils de gestion de projet : *ganttproject* (éditeur de diagrammes de Gantt, offrant une compatibilité avec les formats récents – XML – de *Microsoft Project*), *freemind* (éditeur de « cartes mentales ») ;
- les logiciels de chiffrement *ACID Cryptofiler* en version 5 et 7 – il s'agit des logiciels *Microsoft Windows* exécutés sous CLIP dans l'émulateur *wine* (voir les remarques sur l'intégration d'applications natives *Windows* en section 4.4) ;
- deux choix alternatifs pour un *plugin flash* pour le navigateur : le *plugin flash* propriétaire (*Adobe*), compatible avec tous les formats *flash* mais qui a historiquement fait l'objet d'un grand nombre de vulnérabilités de sécurité, très rapidement exploitées, ou le *plugin open-source* du

[9] Dans la pratique, l'installation proprement dite d'extensions (c'est-à-dire la copie des fichiers correspondants dans le profil de l'utilisateur) reste permise, mais ces extensions seront ensuite ignorées par *Firefox* et *Thunderbird*.

projet *gnash*, mieux intégré dans le modèle de sécurité de CLIP mais moins compatible avec les formats *flash* récents (*flash 9* et *10* en particulier) ;

- un certain nombre d'outils réseau et de productivité en ligne de commande : *openssh, rsync, subversion, vim, make, LaTeX* ;
- le *proxy* local *davmail*, permettant un accès amélioré depuis *Thunderbird* à un serveur *Microsoft Exchange* ;
- plusieurs extensions *Mozilla Firefox* ou *Thunderbird*, dont notamment *noscript* (blocage paramétrable des scripts dans les pages web consultées), *fireftp* (client FTP intégré à *Firefox*) *mailbox-alert* (notification d'arrivée de messages dans *Thunderbird*), *zotero* (gestion de bibliographie dans *Firefox*)

On notera également que certains paquetages optionnels peuvent n'être proposés à l'installation que dans l'une des deux cages RM, RM_H ou RM_B. Il s'agit notamment d'un paquetage de configuration spécifique à RM_H (qui peut être décliné différemment selon le déploiement CLIP), ainsi que des clients graphiques permettant d'interagir avec les diodes montante et cryptographique (cf. 2.2.1) : seul le client « haut » est proposé dans la cage RM_H et le client « bas » dans la cage RM_B, dans la mesure où, même installé, le client « bas » par exemple ne pourrait pas fonctionner dans la cage RM_H (utilisation d'interfaces différentes de la diode, qui ne sont exposées que dans l'autre cage RM).

3.3 Configurations pour passerelles (CLIP-GTW)

La configuration passerelle du système CLIP est typiquement utilisée sur les passerelles de chiffrement IPsec associées aux flux chiffrés UPDATE et RM_H des clients CLIP-RM. Elle est installée sur des postes de type serveur, comprenant au minimum[10] deux interfaces réseau : une interface externe « noire », sur laquelle arrivent les flux IPsec et IKE des postes clients, et une interface interne « rouge » sur laquelle circulent les flux clairs entre postes clients et serveurs RM_H ou UPDATE.

La configuration passerelle se décline concrètement en deux variantes, respectivement pour les passerelles de services (RM_H) et les passerelles de mise à jour (UPDATE). La différence entre ces deux variantes porte sur le mode de téléchargement des mises à jour : dans le cas d'une passerelle UPDATE, le serveur de mise à disposition des mises à jour est accessible directement à travers l'interface interne, et les mises à jour sont téléchargées « en clair » (sans chiffrement IPsec des flux, ce qui n'exclut pas la mise en oeuvre d'un chiffrement applicatif HTTPS). Dans une telle configuration, la cage UPDATE accède uniquement à l'interface interne. Au contraire, le téléchargement des mises à jour depuis une passerelle de services nécessite l'utilisation d'un tunnel IPsec entre l'interface externe de la passerelle RM_H et celle de la passerelle UPDATE du déploiement. Dans ce cas, la cage UPDATE de la passerelle RM_H n'a accès qu'à l'interface externe, et uniquement à travers le tunnel IPsec UPDATE. Une seule et même distribution binaire CLIP-GTW permet d'installer ou de mettre à jour indifféremment l'une ou l'autre variante : le choix de la variante installée correspond à un paramètre d'installation. L'environnement réseau de chaque variante de passerelle est schématisé dans les Figure 4 pour les passerelles de services et Figure 5 pour les passerelles de mise à jour.

[10] Il est également envisageable d'utiliser une troisième interface réseau pour l'administration et l'audit à distance de la passerelle, mais cette configuration n'a jamais été mise en oeuvre à ce stade.

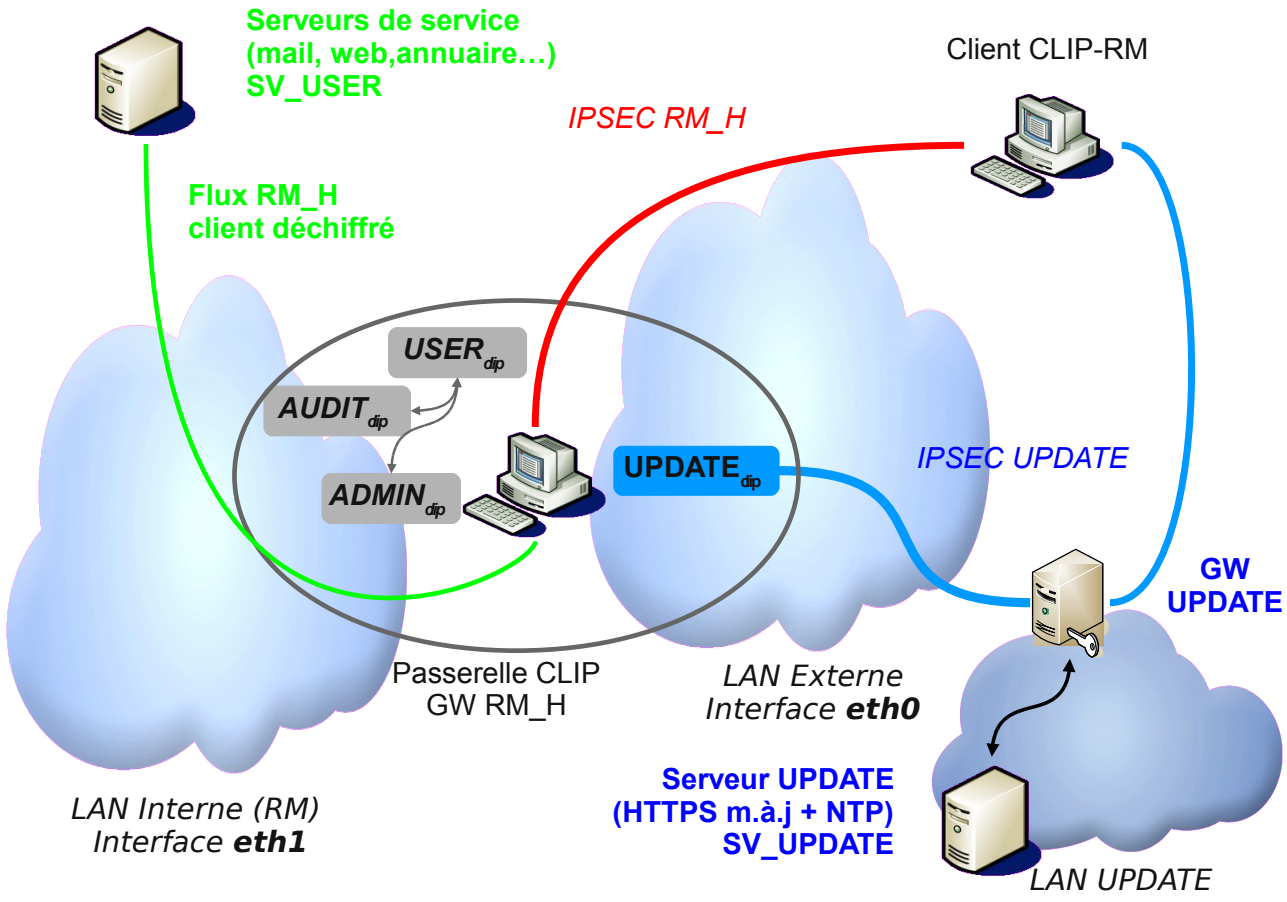

Figure 4: Environnement réseau d'une passerelle CLIP RM_H.

En dehors de cette différence dans l'accès aux mises à jour, les deux types de passerelles fonctionnent de manières identiques. La configuration de chaque passerelle permet de définir un ou plusieurs **réseaux clients**, qui spécifient les adresses internes autorisées comme sources de tunnels IPsec à établir avec la passerelle. Ces adresses correspondent aux adresses sources des en-têtes IP internes (chiffrés) des flux IPsec émis par les clients et à destination des passerelles, et donc aux adresses sources des paquets IP qui sont routés par la passerelle vers son interface interne, après déchiffrement. Les réseaux clients correspondent typiquement à des segments d'adresses IP non routables sur internet, par exemple *192.168.12.0/24* ou *10.1.0.0/16*. Aucune contrainte n'est en revanche associée aux adresses sources externes des tunnels IPsec : la passerelle accepte de négocier un tunnel avec n'importe quel poste client, quelle que soit son adresse publique, du moment que le client est correctement authentifié dans la négociation, et que le tunnel qu'il cherche à établir a pour adresse source interne une adresse d'un des réseaux clients définis pour la passerelle.

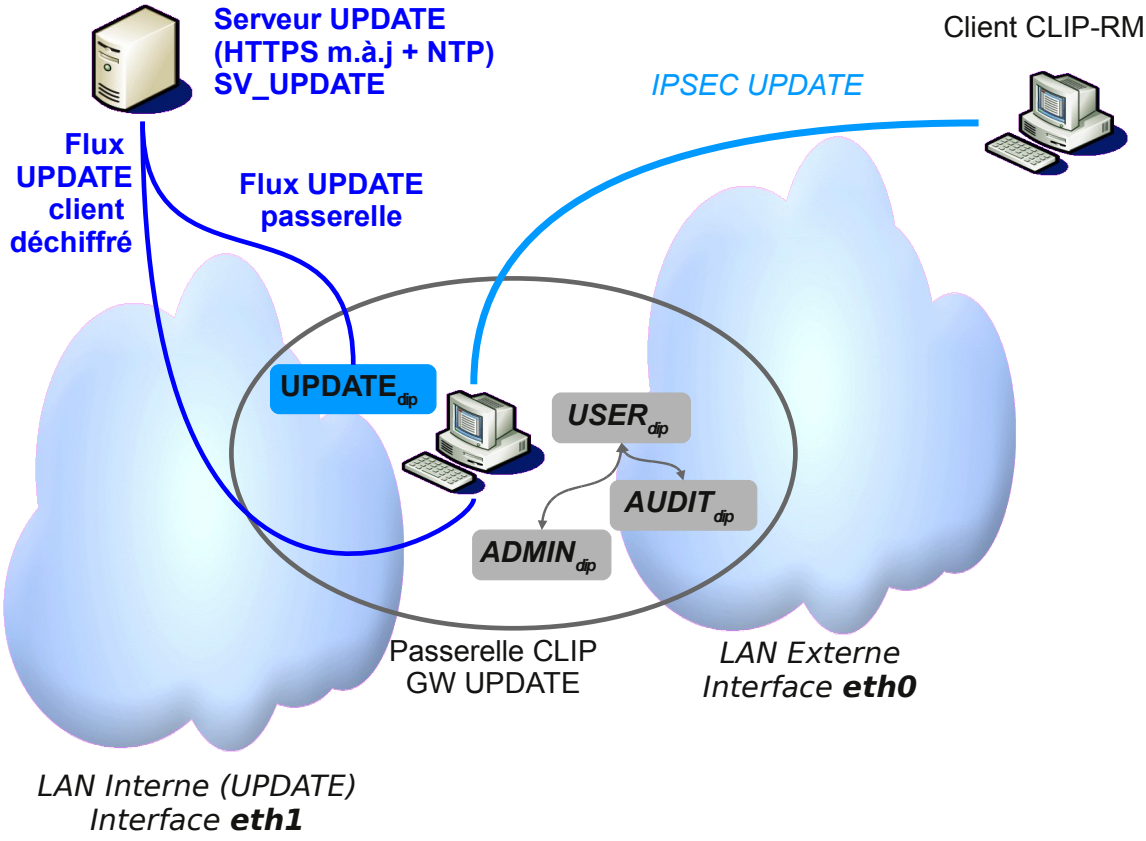

Figure 5: Environnement réseau d'une passerelle CLIP UPDATE.

Les négociations de tunnels IPsec sont réalisées à l'aide du protocole IKEv2, en s'appuyant en général sur des algorithmes et protocoles cryptographiques gouvernementaux, fournis par une bibliothèque CCSD, à l'exception d'éventuelles configurations spécifiques qui ne mettraient en oeuvre que des algorithmes civils (cf. 2.2.2). Lorsque CCSD est utilisée, les négociations sont **authentifiées par des clés privées et publiques ACID v7** : chaque poste (client ou passerelle) dispose de sa propre clé privée, ainsi que des clés publiques de tous les postes avec qui il est appelé à établir des tunnels IPsec. Ainsi, un poste client CLIP-RM biniveau disposera typiquement des clés publiques des passerelles UPDATE et RM_H associées à son déploiement, tandis qu'**une passerelle disposera des clés publiques de chaque client de son déploiement**, ainsi que – uniquement dans le cas d'une passerelle RM_H – de celle de la passerelle UPDATE correspondante. Lors de l'initialisation d'une négociation par un client, celui-ci transmet un identifiant correspondant au champ *SubjectName* de sa clé ACID. Cet identifiant permet à la passerelle de sélectionner la clé publique à utiliser pour authentifier le client, le nom du fichier contenant cette clé correspondant à son *SubjectName*. Ce mode de fonctionnement par bi-clés prépositionnés offre moins de souplesse apparente qu'un fonctionnement classique d'IKE v2 avec infrastructure de gestion de clés, dans lequel chaque poste transmet son propre certificat dans les premiers paquets de la négociation, et n'a besoin localement que du certificat de l'autorité de

certification pour pouvoir authentifier ses pairs. Cependant, la limitation à des bi-clés prépositionnés est le seul mode de fonctionnement sécurisé envisageable, dans la mesure où ACID v7 ne supporte pas à ce stade la mise en oeuvre de listes de révocation de certificats : la révocation d'une clé cliente compromise reste possible sur les passerelles CLIP – il suffit de supprimer la clé publique correspondante sur la passerelle.

Les postes clients jouent toujours le rôle d'*initiator* (émetteur du premier paquet) dans la négociation IKE v2, et les passerelles celui de *responder* (répond aux demandes de négociation, mais ne prend pas lui-même l'initiative d'une négociation). On prendra cependant garde au fait qu'une fois la première association de sécurité négociée avec un client donné, une passerelle est susceptible d'émettre de sa propre initiative des paquets IKE à destination de ce client, par exemple pour renégocier une nouvelle association (renouvellement périodique des clés), ou pour tester la présence du client ou maintenir actives les traductions d'adresses éventuelles qui auraient été détectées entre le client et la passerelle. Ces émissions à l'initiative de la passerelle doivent notamment être prises en compte dans la configuration d'éventuels pare-feux complémentaires mis en oeuvre dans un déploiement (cf. 5.2).

Outre l'établissement de tunnels IPsec, une passerelle CLIP assure également le routage vers son interface interne et le **filtrage des flux clairs associés à ces tunnels**. Le filtrage n'autorise de manière générale que les connexions TCP ou UDP, à destination de l'interface interne, initiées depuis une adresse source appartenant à un des réseaux clients définis pour la passerelle, et issues d'un tunnel IPsec établi sur l'interface externe, ainsi que les réponses à ces connexions (filtrage avec suivi de connexion). La configuration locale de la passerelle permet de définir les ports destination des flux autorisés dans le sens entrant. En revanche, les adresses destination des différents flux autorisés en entrée ne peuvent pas être précisées (toute destination est autorisée, du moment que le routage associe cette destination à l'interface interne). Les flux clairs, non issus d'un tunnel IPsec dans le sens entrant, ou non destinés à être encapsulés dans un tel tunnel dans le sens sortant, ne sont en aucun cas autorisés, pas plus que l'établissement de nouvelles connexions depuis un serveur connecté à l'interface interne à destination de clients sur l'interface externe.

4 Mécanismes de mise à jour

4.1 Types de paquetages

4.1.1 Catégories de paquetages

Les mises à jour des postes CLIP sont réalisées paquetage par paquetage. Chaque paquetage correspond à un module logiciel qui peut être décorrélé (aux dépendances près) des autres modules. Les paquetages déployés au sein d'un système CLIP peuvent être catégorisés selon plusieurs logiques, en fonction de leur mode de mise à jour et d'installation.

On distinguera en premier lieu les **paquetages CLIP** des **paquetages RM** : les cages RM d'un système CLIP-RM utilisent une distribution complète de paquetages, différente de celle utilisée dans le socle CLIP. Ainsi, un même paquetage pourra être installé indépendamment dans le socle CLIP (installation partagée potentiellement par toutes les cages CLIP), et dans chaque cage RM. De plus, l'appartenance aux autres catégories décrites dans les paragraphes suivants pourra varier selon la distribution : un même paquetage pourra par exemple être essentiel dans la distribution CLIP, et secondaire dans la distribution RM. Naturellement, seule la distribution CLIP est présente au sein d'une configuration de type passerelle.

La distinction entre paquetages dit « **essentiels** » et « **secondaires** » tient au niveau de criticité des paquetages pour le fonctionnement du système, et conditionne le mode de mise à jour des paquetages concernés. Les paquetages essentiels sont ceux dont le fonctionnement correct est indispensable à la mise en oeuvre du service de mise à jour. Tous les paquetages qui ne sont pas essentiels sont considérés comme secondaires. Par définition, un fonctionnement incorrect d'un paquetage secondaire ne peut pas bloquer les mises à jour, et peut donc toujours être corrigé par une mise à jour. Les deux types de paquetages ne sont pas mis à jour de la même manière. Les mises à jour de paquetages secondaires sont appliquées au fur et à mesure, dès leur disponibilité, tandis que les mises à jour de paquetages essentiels ne sont appliquées qu'au cours de la séquence de démarrage. Ainsi, si une mise à jour de paquetages essentiels devient disponible en cours de fonctionnement du système, cette mise à jour sera stockée dans l'attente du redémarrage du poste, qui permettra de l'appliquer. Par ailleurs, l'application d'une mise à jour de paquetage essentiel CLIP s'accompagnera d'une bascule entre les deux systèmes CLIP installés sur le poste, comme décrit dans la section suivante.

Enfin, les paquetages secondaires sont répartis selon deux sous-catégories : paquetages **obligatoires** et **optionnels**. Les paquetages obligatoires sont ceux qui sont systématiquement installés sur chaque poste CLIP, et inclus dans la procédure d'installation initiale. A contrario, les paquetages optionnels ne sont installés sur un poste que sur choix de l'administrateur du poste, et ce indépendamment pour chaque poste. Ils ne sont pas nécessairement inclus dans l'installation initiale du poste. Ainsi, l'ensemble des paquetages optionnels d'une distribution CLIP constitue une liste blanche de modules logiciels complémentaires disponibles pour installation par l'administrateur de chaque poste. Des listes de paquetages optionnels dont l'installation est requise sur le poste peuvent être définies et modifiées par tout utilisateur disposant des privilèges d'administration sur le poste (utilisateur privilégié ou administrateur, cf. 6.1). On notera qu'il n'est pas nécessaire de spécifier tous les paquetages optionnels nécessaires à la satisfaction des dépendances, celles-ci étant gérées automatiquement. Afin de

simplifier la gestion, les interfaces graphiques de configuration des paquetages optionnels n'affichent d'ailleurs que les paquetages optionnels qui incluent une description en français (champ *debian description-fr:*, spécifique à CLIP), et pas leur dépendances. L'utilisateur verra ainsi par exemple le paquetage optionnel *gimp*, mais pas ses dépendances *babl* et *gegl*, qui sont également des paquetages optionnels mais qui n'ont pas de raison d'être installées seules.

4.1.2 Paquetages « configurations »

La répartition entre paquetages essentiels et secondaires se fait principalement en fonction des miroirs sources des téléchargement (cf. 4.3 ci-dessous) : pour chaque distribution, CLIP ou RM, on aura typiquement un miroir de paquetages essentiels et un de paquetages secondaires. En revanche, la distinction entre paquetages obligatoires et optionnels repose sur la notion de « **configuration** ». Une configuration est un paquetage spécifique, virtuel au sens où il n'installe pas de fichiers dans le système (à l'exception d'un fichier *changelog* listant les évolutions successives). Une configuration est associée à chaque miroir[11]. Elle référence de manière unique tous les paquetages disponibles dans le miroir, selon des modalités différentes selon qu'ils sont obligatoires ou optionnels.

Les paquetages obligatoires sont listés dans les **dépendances obligatoires** (champ *debian depends:*) de la configuration, avec des versions uniques (la dépendance n'est satisfaite que par une version unique du paquetage, et pas par exemple par toutes les versions supérieures ou égales à une version donnée). Ainsi, l'installation de la configuration entraîne, par le jeu des dépendances, celle de tous les paquetages obligatoires associés, et donc de tous les paquetages obligatoires pour la distribution et la priorité (essentiel ou secondaire) considérées, dans les versions exactes correspondant à la mise à jour considérée.

Les paquetages optionnels sont également listés dans des versions uniques, mais cette fois comme des **dépendances optionnelles** (champ *debian suggests:*) du paquetage configuration. Ainsi, ces paquetages ne sont pas automatiquement installés lors de l'installation de la configuration associée, les dépendances optionnelles n'étant pas prises en compte automatiquement par *apt-get*. Les scripts de mise à jour (téléchargement et installation) spécifiques à CLIP extraient ces champs *suggests* pour établir les listes blanches de paquetages optionnels autorisés.

4.2 Signature des mises à jour

Les paquetages utilisés pour les mises à jour sont individuellement signés. Chaque paquetage incorpore deux signatures, ajoutées au format de paquetage *debian* en fin de fichier, comme schématisé dans la Figure 6. Ces signatures représentent organisationnellement deux rôles distincts dans la chaîne de production de mises à jour :

- La première signature est dite **signature développeur**, elle couvre l'ensemble du paquetage hors signatures. Elle est normalement créée par le développeur qui a construit et compilé le paquetage considéré.
- La deuxième signature est dite **signature validateur** et couvre l'ensemble du paquetage hors signatures ainsi que la signature du développeur. Elle est créée ultérieurement à la signature

[11] Il y a donc typiquement quatre configurations pour une distribution binaire CLIP-RM : paquetages essentiels et secondaires, CLIP et RM. Une distribution binaire CLIP-GTW ne comporte que deux configurations : paquetages essentiels et secondaires CLIP.

développeur, par la personne ayant assemblé et validé[12] l'ensemble de la configuration constituant la mise à jour.

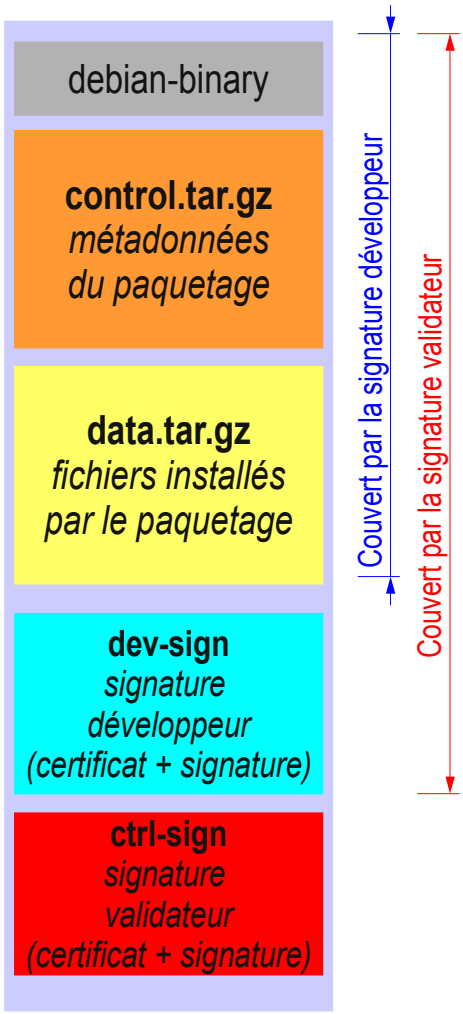

Figure 6: Positionnement et champs couverts par les signatures de paquetages.

Les deux rôles, développeur et validateur, ne sont pas nécessairement détenus par deux personnes physiques distinctes, mais peuvent l'être selon les principes organisationnels adoptés pour la production des mises à jour pour un déploiement donné, par exemple dans les cas de figure suivants :

- Le développeur est un industriel produisant des paquetages CLIP dans le cadre d'un contrat, tandis que le validateur appartient à l'entité de l'Administration responsable du déploiement.

[12] La procédure de validation n'est pas formellement définie. Elle peut inclure notamment la validation du fonctionnement correct de la mise à jour elle-même et du poste après son application, l'innocuité en termes de sécurité de la mise à jour, et le bien fondé de cette mise à jour.

- Le développeur appartient à l'équipe de développement principale du coeur CLIP, produisant des paquetages génériques pour tous les déploiements, tandis que le validateur appartient à une « équipe de marque » chargée de valider les mises à jour pour le déploiement particulier concerné.

Quelle que soit la logique organisationnelle sous-jacente, la présence de deux signatures correctes est nécessaire à l'utilisation du paquetage par un poste CLIP. Les signatures sont systématiquement vérifiées aussi bien lors du téléchargement que de l'installation des paquetages (voir sections suivantes), et tout paquetage incorrectement signé est rejeté avec journalisation.

Les signatures utilisées pour les paquetages CLIP sont produites à l'aide d'une bibliothèque cryptographique gouvernementale compatible avec l'API CCSD (cf. [CCSD]), à partir de clés privées nominatives issues de deux Centres d'Elaboration des Clés (CEC) ACID v.7 distincts, un CEC développeurs et un CEC validateur. Les fichiers de signatures insérés dans les paquetages combinent en fait la signature proprement dite et le certificat du signataire, de telle sorte que seules les « **clés de vérification** », équivalents pour la logique cryptographique ACID / CCSD des clés publiques d'autorité de certification associées à chaque CEC, doivent être présentes sur les postes clients. Ces clés de vérification font partie des éléments cryptographiques installés lors de l'installation initiale du poste, et non modifiables ensuite (cf. 5.3).

4.3 Téléchargement et installation des mises à jour

Le téléchargement et l'application des mises à jour sont au sein du système CLIP deux opérations décorrélées, réalisées indépendamment à des moments différents, et ce quelque soit le type de paquetage concerné (aussi bien essentiel que secondaire, optionnel ou non). L'intermédiaire entre les deux se fait par des miroirs de paquetages locaux, à raison d'un miroir pour CLIP, et un par cage RM. Ces miroirs sont des miroirs de paquetages *debian* simples, comportant chacun un répertoire de stockage des paquetages et un fichier *Packages.gz* décrivant ces paquetages. Chaque miroir local ne contient à un moment donné que les dernières versions des paquetages disponibles. Ainsi, lorsque le système est à jour, le miroir contient uniquement les paquetages présentement installés. A contrario, lorsqu'une mise à jour vient d'être téléchargée et n'a pas encore été appliquée, le miroir local contient les paquetages dans leur version mise à jour, et non plus dans la version présentement installée (dans le cas où les deux diffèrent, c'est-à-dire lorsque le paquetage concerné est effectivement à mettre à jour). On notera qu'il n'y a qu'un seul miroir local pour les paquetages essentiels et secondaires de chaque compartiment CLIP, RM_H ou RM_B : les deux types de paquetages sont mélangés au sein d'un seul miroir local, alors qu'ils proviennent de miroirs sources distincts.

4.3.1 Téléchargement des mises à jour

Le téléchargement des mises à jour est réalisé entièrement dans la cage UPDATE, y compris pour le téléchargement des mises à jour des éventuelles cages RM. Il est lancé périodiquement pendant le fonctionnement normal du système, à raison d'une fois par heure et par miroir (donc trois fois par heure pour un poste CLIP-RM comportant deux cages RM) – on notera que les mises à jour essentielles et secondaires pour chaque compartiment sont téléchargées simultanément. L'heure précise de déclenchement d'une tâche périodique de téléchargement est tirée aléatoirement pour chaque poste, de manière à éviter que les postes ne viennent tous télécharger leurs mises à jour au même instant. Un téléchargement peut par ailleurs être lancé explicitement, à un instant arbitraire et indépendamment

pour chaque miroir, par un utilisateur local du poste disposant des privilèges d'administration, c'est-à-dire soit un administrateur, soit un utilisateur privilégié (mais pas un utilisateur nomade, cf. 6.1)[13]. Les sources de téléchargement de mises à jour peuvent être de trois types :

- miroir HTTPS en ligne, accessible à travers le tunnel IPsec associé à la cage UPDATE (donc hébergé sur un serveur placé derrière la passerelle UPDATE du déploiement) ;
- support amovible CD-ROM monté (en lecture seule) sur */mnt/cdrom* dans la cage UPDATE ;
- support amovible USB monté (en lecture seule ou lecture-écriture) sur */mnt/usb* dans la cage UPDATE.

Pour les deux derniers types de sources (supports amovibles), le montage n'est pas réalisé automatiquement : il devra avoir été réalisé au préalable de manière explicite par un utilisateur disposant des privilèges d'administration. On notera bien que seul un accès en lecture est nécessaire pour le téléchargement des mises à jour – un support amovible USB utilisé à cette fin n'a pas besoin de pouvoir être monté en lecture-écriture dans la cage UPDATE (il n'est notamment pas nécessaire que le support soit signé). Pour les sources réseau (HTTPS), le téléchargement périodique est automatiquement désactivé lors de l'utilisation d'un profil réseau UMTS (cf. 6.3), afin de ne pas consommer de bande passante à l'insu de l'utilisateur. Le téléchargement reste possible dans ce cas sur commande explicite d'un utilisateur disposant des privilèges d'administration[14].

Outre le miroir local, le téléchargement fait également intervenir un cache local de téléchargement, répertoire dans lequel les paquetages sont téléchargés et vérifiés avant leur injection dans le miroir local. Les derniers paquetages téléchargés restent présents dans le cache, afin d'éviter les téléchargements multiples et de permettre la reprise sur interruption. De manière à limiter l'occupation de disque et les copies inutiles, les paquetages présents à la fois dans le cache et dans le miroir local sont en fait des « liens durs » UNIX pointant vers un fichier sous-jacent commun, qui est supprimé dès lors que les deux liens ont été supprimés.

Quelle que soit la source utilisée, un téléchargement de mises à jour consiste à lister les nouvelles configurations disponibles dans la source, à télécharger dans le cache ces configurations (s'il y en a) et leur dépendances obligatoires, puis à télécharger dans le cache les mises à jour disponibles de paquetages optionnels localement sélectionnés, si ces paquetages sont autorisés par la liste blanche de paquetages optionnels. Cette liste blanche est constituée par la concaténation des dépendances optionnelles des différentes configurations présentes dans le miroir, ou nouvellement téléchargées le cas échéant. Après téléchargement dans le cache, les signatures des paquetages sont vérifiées, puis, en l'absence de signature incorrecte, les nouveaux paquetages sont copiés dans le miroir local. Ce dernier, et le cache de téléchargement, sont ensuite nettoyés en ne conservant que les dernières versions des paquetages.

Le téléchargement des mises à jour est robuste aux interruptions. En cas d'interruption d'un téléchargement, la prochaine invocation du téléchargement procédera en premier lieu à la vérification des signatures des paquetages précédemment téléchargés dans le cache mais pas injectés dans le miroir local, de manière à conserver ceux qui ont été correctement téléchargés avant de reprendre le

[13] Un téléchargement automatique des mises à jour peut également être réalisé à chaque démarrage du système, mais l'option de configuration correspondante n'est pas activée par défaut, dans la mesure où il peut en résulter un long délai au démarrage.

[14] Il est par ailleurs recommandé, au sein d'un poste CLIP-RM à deux cages RM, de laisser le miroir local associé à chaque cage RM comme source de téléchargement pour l'autre cage RM, afin d'éviter de télécharger deux fois des paquetages identiques.

téléchargement.

4.3.2 Installation des mises à jour

L'installation des mises à jour consiste à installer au sein du système les nouveaux paquetages disponibles dans le miroir local, et uniquement dans celui-ci (il n'y a pas d'installation directe depuis un miroir « distant » - réseau ou sur support amovible). L'installation d'un paquetage est systématiquement précédée d'une vérification de sa double signature, et, dans le cas d'un paquetage optionnel, de son appartenance à la liste blanche de paquetages optionnels définie par les dépendances des configurations déjà installées ou en cours d'installation.

Contrairement au téléchargement, l'installation est réalisée séparément pour les paquetages essentiels et secondaires de chaque compartiment. Les mises à jour essentielles ne sont installées que lors de la séquence de démarrage du poste, la mise à jour des paquetages essentiels CLIP donnant par ailleurs lieu à traitement particulier décrit plus bas. Les mises à jour secondaires disponibles sont également installées au démarrage du poste, après les mises à jour essentielles correspondantes, mais à la différence de ces dernières, elles sont de plus appliquées périodiquement (toutes les heures) pendant le fonctionnement du système. Ainsi, une mise à jour téléchargée par la tâche périodique de téléchargement sera installée :

- dans l'**heure qui suit le téléchargement**, ou au **prochain redémarrage** s'il intervient dans l'heure, dans le cas de paquetages secondaires ;
- au **prochain redémarrage uniquement** dans le case de paquetages essentiels.

On notera cependant que certaines mises à jour secondaires peuvent dépendre d'une mise à jour essentielle pour leur installation, et ne seront donc installée qu'après redémarrage et mise à jour des paquetages essentiels. L'objectif visé par la restriction des mises à jour aux seules séquences de démarrage est double :

- fonctionnel d'une part, car la mise à jour des paquetages essentiels peut perturber le fonctionnement normal du poste, et nécessite de toutes manières un redémarrage pour une prise en compte correcte ;
- sécuritaire d'autre part, car il est souhaitable de ne modifier les modules logiciels critiques au sens de la sécurité du système que dans un environnement particulièrement intègre ; l'installation par un système qui vient d'être démarré, et n'a pas encore activé ses interfaces réseau ni permis l'ouverture de session utilisateur remplit cette objectif en limitant fortement la possibilité que le système ait pu être corrompu en mémoire à ce stade.

A l'instar du téléchargement, l'installation des mises à jour fait intervenir, outre le miroir local, un répertoire cache d'installation (distinct du cache de téléchargement), dans lequel les paquetages sont copiés avant leur installation, et dans lequel est ensuite conservée uniquement la dernière version installée de chaque paquetage du compartiment. La copie dans un cache sert d'une part à la mise en oeuvre des vérifications, et d'autre part et surtout à la procédure de retour en arrière sur erreur associée aux mises à jour. Ainsi, lorsqu'une installation de paquetages échoue à mi-course, les scripts de mise à jour peuvent automatiquement réinstaller les anciennes versions des paquetages qui ont été mis à jour (ou désinstallés) avant la sortie en erreur, les fichiers de paquetages associés à ces anciennes versions étant alors encore disponibles dans le cache d'installation. Cette même technique de retour en arrière

assure la résistance aux interruptions de la procédure d'installation : si l'installation d'un groupe de paquetages a été interrompue en cours de traitement (par exemple suite au redémarrage du poste), la prochaine invocation de la procédure d'installation commencera par rétablir un état cohérent par réinstallation des anciennes versions de paquetages, avant de reprendre au début l'installation. Tout comme le cache de téléchargement, le cache d'installation est géré sous la forme de liens durs potentiellement partagés avec le miroir local, et nettoyé automatiquement à l'issue d'une installation réussie pour ne conserver que les versions effectivement installées des paquetages.

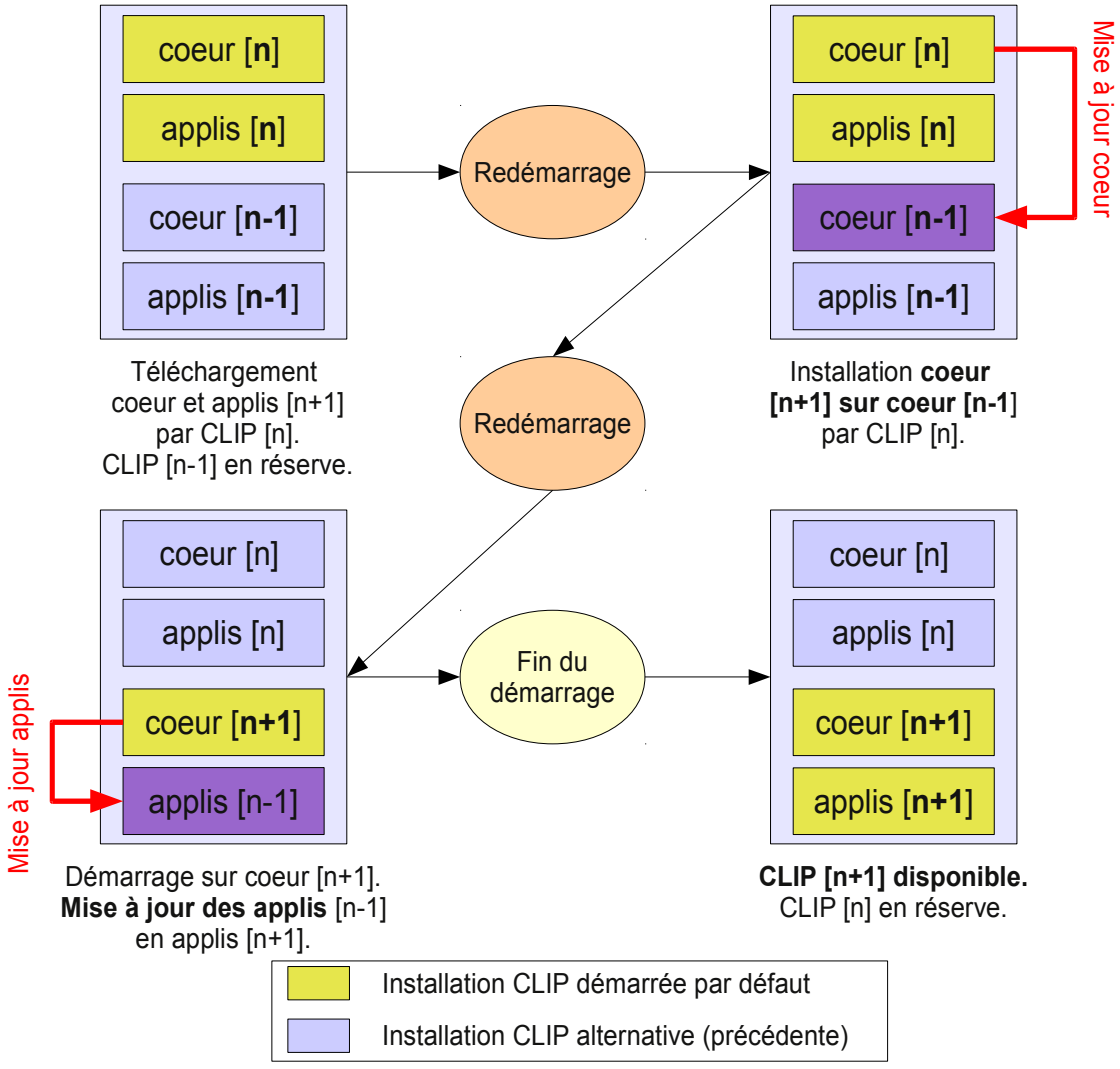

Figure 7: Principe de mise à jour du coeur CLIP, avec changement de jeu de partitions.
Dans cette figure, le terme "*coeur*" désigne l'ensemble des paquetages essentiels CLIP. Le terme "*applis*" réprésente quant à lui l'ensemble des paquetages secondaires CLIP, et les éventuels paquetages essentiels et secondaires RM.

Le coeur CLIP (paquetages essentiels CLIP) est traité spécifiquement pour ce qui est de l'installation des mises à jour. En effet, les systèmes de fichiers sur lesquels sont installés les paquetages essentiels

CLIP ne sont jamais accessibles en écriture, même au démarrage du système[15], afin de protéger l'intégrité de ces fichiers essentiels, aussi bien en termes fonctionnels que sécuritaires. Ainsi, la mise à jour des paquetages essentiels CLIP, outre le fait qu'elle n'est possible qu'au démarrage du poste, s'accompagne nécessairement d'un basculement sur une installation CLIP alternative. A cette fin, **deux systèmes CLIP complets sont systématiquement installés côte à côte** sur chaque poste. Le système sur lequel démarre le poste par défaut est toujours le plus à jour (version *[n]*), tandis que le deuxième système, sur lequel il est possible de démarrer par un choix alternatif proposé par le chargeur de démarrage, correspond à la version précédente (*[n-1]*) des paquetages essentiels CLIP, et souvent à des versions également moins à jour des paquetages secondaires CLIP et des paquetages RM. Lors d'une mise à jour des paquetages essentiels au démarrage du poste, le système CLIP *[n]* monte temporairement les partitions du système CLIP *[n-1]*, et procède à l'installation sur celles-ci des paquetages essentiels en version *[n+1]*. A l'issue de cette installation, les miroirs locaux du système *[n],* les plus à jour, sont copiés sur le nouveau système *[n+1]* (précédemment *[n-1]*), qui devient le nouveau système par défaut au démarrage, puis le poste redémarre immédiatement. Lors de ce second démarrage sur le poste *[n+1]*, celui-ci met automatiquement à jour ses paquetages secondaires CLIP et ses éventuels paquetages RM à l'aide des miroirs locaux recopiés depuis le système *[n]*, afin d'être entièrement à jour avant de permettre toute ouverture de session utilisateur. Ce principe de mise à jour est résumé dans la Figure 7[16].

Cette utilisation de deux installations alternatives, avec bascule automatique lors de la mise à jour, présente deux avantages majeurs :

- Elle permet de maintenir la propriété d'accès en lecture seule au coeur du système y compris pendant les mises à jour : les outils de mise à jour ne se modifient pas eux-même.
- Elle offre une sécurité supplémentaire en cas d'échec « catastrophique » d'une mise à jour (non rattrapé par les fonctions de gestion d'erreur des outils de mise à jour), ou d'application correcte d'une mise à jour non fonctionnelle – un système CLIP alternatif reste toujours disponible dans ce cas, et permet aux utilisateurs de continuer à utiliser leur poste dans l'attente d'une mise à jour corrective.

On notera que l'application des mises à jour de paquetages essentiels CLIP n'est pas complètement automatique au démarrage : l'utilisateur peut décider s'il souhaite appliquer immédiatement les mises à jour disponibles, ou continuer le démarrage sans application des mises à jour, par exemple car il ne souhaite pas attendre la fin de la procédure de mise à jour pour pouvoir utiliser son poste. Cependant, le refus d'application des mises à jour n'est autorisé que cinq fois consécutives : au redémarrage suivant le cinquième refus, les mises à jour seront installées automatiquement. De même, l'absence de réponse de l'utilisateur entraîne l'application automatique des mises à jour après 30 secondes d'attente.

[15] Au contraire de ceux qui hébergent les éventuels paquetages essentiels RM, qui ne sont jamais accessibles en écriture depuis l'intérieur des cages RM, mais qui le sont depuis le socle CLIP lors du démarrage du poste.

[16] Le schéma de mise à jour présenté ici correspond au cas nominal. Il est cependant aussi possible de réaliser la mise à jour depuis le système CLIP « précédent » (*[n-1]*), en écrasant alors le système *[n]* (qui serait par exemple non fonctionnel suite à un problème de mise à jour antérieur) avec le système *[n+1]*.

4.4 Intégration de nouveaux paquetages dans une distribution CLIP

L'ajout de nouveaux logiciels à un système CLIP nécessite leur intégration préalable comme paquetages (obligatoires ou optionnels) dans la distribution CLIP concernée. Cette intégration est en général possible dès lors qu'une version du logiciel est disponible pour un système Linux, avec quelques réserves énumérées ci-dessous (pour des éléments techniques plus détaillés concernant le portage dans CLIP d'un logiciel, le lecteur pourra également se référer au document de référence [CLIP_1104]).

Disponibilité du logiciel et de ses sources

La disponibilité du logiciel dans une **version native Linux** est un prérequis quasiment incontournable (voir remarque plus bas sur l'utilisation de logiciels Windows). L'intégration du logiciel dans la distribution *Gentoo*[17] tend à faciliter l'intégration dans CLIP, mais n'est pas strictement nécessaire. Par ailleurs, la **disponibilité du code source du logiciel**, et la possibilité de compiler ce code source (dans le cas où la compilation a un sens) et de distribuer le résultat, est très fortement souhaitable, à plusieurs titres :

- elle permet d'appliquer au logiciel les mécanismes de durcissement liés à la compilation qui sont généralement mis en oeuvre dans CLIP ;
- elle permet au besoin de modifier le logiciel, soit pour l'adapter à des particularités de l'environnement CLIP, soit pour corriger des problèmes (bugs, vulnérabilités de sécurité) pour lesquels aucun correctif n'a encore été publié ;
- elle facilite l'analyse des éventuels problèmes rencontrés dans l'intégration du logiciel sur CLIP.

L'intégration d'un logiciel disponible uniquement sous forme binaire (native Linux) est la plupart du temps possible moyennant un travail supplémentaire, mais ne peut pas être garantie.

Au delà de ces éléments de nature technique, d'autres contraintes peuvent empêcher l'intégration d'un logiciel dans CLIP :

- Il n'est pas souhaitable d'intégrer dans CLIP un logiciel qui n'est **plus maintenu par ses auteurs**, car le maintien en conditions opérationnelle et de sécurité de ce logiciel risque de s'avérer impossible avec le temps.
- Les **conditions de distribution** (licence) du logiciel doivent être compatible de sa distribution dans CLIP. Cette contrainte est de manière générale satisfaite par tous les logiciels libres sous licence *open source*, ainsi que par les logiciels autorisant une redistribution libre sous forme binaire. En dehors de ces deux cas simples, il est probable que le logiciel ne puisse pas être légalement intégré à la distribution CLIP, à moins qu'une étude juridique approfondie n'établisse le contraire.

Remarque 4 : mise en oeuvre de logiciels natifs Windows

La mise en oeuvre dans CLIP de logiciels natifs Windows est dans certains cas possibles à travers la couche d'émulation Wine. *Celle-ci est par exemple utilisée pour permettre*

[17]Existence d'un *ebuild* soit dans la distribution *Gentoo* de base, soit dans l'un des *overlays* associés à cette distribution, qui mettent à disposition des paquetages complémentaires, généralement à titre expérimental.

l'utilisation des logiciels ACID CRYPTOFILER *sur CLIP. Cependant, cette approche doit rester exceptionnelle – ne serait-ce que parce qu'elle rompt largement le modèle de sécurité CLIP, et il ne faut pas en général compter sur la possibilité d'intégrer un logiciel Windows dans CLIP. Outre les considérations sécuritaires, l'émulation avec Wine se heurte en effet à de nombreux obstacles :*

- *la couche d'émulation* Wine *est loin d'être complète, et ne supporte que certains logiciels (parfois uniquement avec des fonctionnalités réduites) ;*
- *même si l'émulation s'avère fonctionnelle, elle n'offrira qu'une ergonomie limitée : performances réduites, mauvaise intégration dans l'environnement d'utilisation (copier-coller, appels à des utilitaires externes, chemins différents dans le système de fichiers, ...) ;*
- *la distribution de logiciels Windows sous licence propriétaire dans CLIP est le plus souvent impossible au regard des contraintes juridiques évoquées plus haut.*

Compatibilité de l'environnement d'exécution

L'environnement d'exécution de code binaire dans un système CLIP présente quelques spécificités par rapport à un environnement Linux traditionnel, principalement liées aux **mécanismes de durcissement mémoire** (cf. 2.1.2) et à la mise en oeuvre du **principe « W^X »** (cf. 2.1.4). En particulier, aucune zone mémoire ne peut normalement être simultanément ou successivement exécutable et inscriptible, ce qui interdit notamment le fonctionnement d'exécutables qui génèrent du code à la volée (par exemple interpréteurs). De même, l'ensemble de l'espace mémoire des processus est « randomisé », ce qui interdit le fonctionnement d'exécutables qui nécessitent des adresses fixes (cas très rare en pratique). Ces différentes protections mémoire peuvent être désactivées au cas par cas pour certains exécutables, mais cette pratique est à éviter autant que possible, car elle réduit le niveau de sécurité du système. Par ailleurs, le principe « W^X » interdit tout exécution depuis des répertoires accessibles en écriture, ce qui peut éventuellement poser problème à des logiciels qui nécessiteraient un tel mécanisme (par exemple pour stocker des *plugins* ou scripts spécifiques dans le répertoire personnel de l'utilisateur ou dans */tmp*). Ces restrictions ne pouvant pas être contournées, la seule solution envisageable dans ce cas consiste à modifier le comportement du logiciel concerné.

Au delà du code binaire, un système CLIP supporte également l'exécution de logiciels (scripts, etc.) dans un certains nombre de **langages interprété**s, plus spécifiquement :

- Le langage *shell* UNIX – on notera toutefois que le *shell* disponible dans les cages RM n'est pas le *shell bash* standard sur les systèmes Linux, mais un *shell* plus simple qui ne supporte pas toutes les constructions syntaxiques de *bash* ;
- *perl*
- *python*
- *java* (uniquement dans les cages RM)
- *flash*(uniquement dans les cages RM, paquetage optionnel)

L'ajout d'autres interpréteurs est possible, mais nécessite un travail spécifique d'adaptation de l'interpréteur concerné afin qu'il respecte le principe de sécurité « W^X ».

Autres contraintes

L'intégration dans CLIP de logiciels mettant en oeuvre des privilèges élevés, par exemple pour

configurer certains périphériques matériels, peut nécessiter des adaptations supplémentaires au modèle de sécurité de CLIP, par exemple la séparation du logiciel en deux composants, un exécuté dans le socle CLIP et disposant des privilèges nécessaires, et l'autre exécuté dans une cage utilisateur, sans privilèges mais faisant appel au besoin au composant privilégié à travers une interface simple (principe de **séparation des privilèges**). De même, des logiciels nécessitant une authentification de l'utilisateur depuis un environnement non privilégié (typiquement cage RM) devront être adaptés pour réaliser celle-ci à l'aide des mécanismes d'authentification spécifiques à CLIP (démon *pwcheckd*, etc.). Ces différentes contraintes sont détaillées dans le document de référence [CLIP_1104].

Par ailleurs, la mise en oeuvre de périphériques spécifiques nécessitera au préalable l'intégration de leur support dans le système CLIP, selon les modalités décrites en section 5.1.

5 Déploiement de systèmes CLIP

La mise en oeuvre d'un système CLIP dans le cadre d'un nouveau déploiement impose une réflexion préalable concernant les matériels sur lesquels CLIP doit être installé, l'environnement réseau du déploiement, et la procédure d'installation et d'administration des postes.

5.1 Matériel supportés

Notion de configuration matérielle

L'installeur CLIP supporte par défaut un certain nombre de configurations matérielles prédéfinies, qui sont celles sur lesquelles le système a déjà été installé avec succès et pour lesquelles il est officiellement supporté. Il n'est généralement pas difficile d'ajouter de nouvelles configurations matérielles, sous réserve que celles-ci soient supportées par le noyau Linux (contraintes détaillées plus bas). A titre indicatif, les configurations matérielles supportées au moment de la rédaction du présent document (liste susceptible d'évoluer) sont les suivantes :

- configurations pour poste client portable (CLIP-RM) :
  - portables *DELL Latitude D420, D430, D520, D530, E4300* et *E5500* ;
  - portables *LENOVO Thinkpad X61* et *X200* ;
- configurations serveur pour passerelles (CLIP-GTW) :
  - *DELL PowerEdge 2900* (serveur non rackable) ;
  - *DELL PowerEdge R200* et *R610* (serveurs rackables) :

Chaque configuration matérielle prédéfinie correspond essentiellement à une liste de pilotes de périphériques (modules Linux), et le cas échéant de *firmwares* binaires associés à ces périphériques, à inclure dans le noyau CLIP. Ce dernier est en effet distribué sous une forme générique, incluant un ensemble de pilotes modulaires pour les différents périphériques susceptibles d'être mis en oeuvre par le système CLIP : contrôleurs de disques dur SATA, SCSI et IDE, contrôleurs RAID, cartes réseau (filaires ou sans fil), cartes son et cartes graphiques, etc... Lors de l'installation sur un poste CLIP donné, une liste de modules est utilisée afin de définir ceux de ces pilotes qui seront effectivement chargés au démarrage du poste, à l'exclusion de tous les autres. Cette liste constitue un paramètre d'installation, et n'est pas modifiable (autrement que de manière automatique par une mise à jour) pendant le fonctionnement ultérieur du poste.

Ajout d'une nouvelle configuration

Le support d'un nouveau matériel par CLIP consiste dans la plupart des cas à définir la liste de modules adaptée à la configuration, et à l'incorporer dans une nouvelle version de l'installeur. Cette approche simple n'est cependant suffisante que lorsque les prérequis suivants sont vérifiés :

- les **modules nécessaires** au fonctionnement du système sont **inclus dans le noyau Linux standard** (noyau « vanilla »), et présents dans le noyau CLIP générique (paquetage *clip-kernel-modular*) ;

- les éventuels ***firmwares* binaires** associés à ces modules sont **présents au sein du système CLIP**, soit directement dans le paquetage *clip-kernel-modular* lui-même (*firmwares* inclus dans le noyau Linux standard), soit dans le paquetage *wireless-firmwares* (*firmwares* complémentaires associés aux cartes *Wifi* supportées par CLIP).

La non-satisfaction du premier prérequis n'est pas nécessairement bloquante, sous réserve que le pilote manquant soit effectivement inclus dans le noyau Linux standard, mais pas dans le noyau CLIP : il suffira dans ce cas de générer une nouvelle révision du paquetage *clip-kernel-modular*, incorporant le module manquant. De même, l'absence d'un *firmware* spécifique peut être résolue par l'ajout de celui-ci dans un nouveau paquetage, sous réserve que le *firmware* concerné soit disponible et redistribuable dans des conditions légales compatibles avec CLIP (cf. 4.4).

En revanche, le support d'un matériel est **beaucoup plus problématique lorsque les pilotes associés ne sont pas disponibles dans le noyau Linux standard**. L'absence complète de pilotes natifs Linux est évidemment bloquante, à moins d'obtenir le développement de tels pilotes par une entité tierce (de tels développements ne peuvent pas être réalisés par l'équipe de développement CLIP). Même dans le cas où des pilotes natifs Linux seraient disponibles en dehors des sources du noyau Linux standard (sous formes de modules distribués séparément par exemple), l'intégration de ces pilotes dans la distribution CLIP sera généralement refusée, pour plusieurs raisons :

- La non-intégration d'un module dans le noyau Linux standard est souvent liées à des incompatibilités juridiques, interdisant la distribution de ce module sous les mêmes conditions que Linux. De telles incompatibilités interdiront généralement aussi sa distribution dans CLIP.
- En dehors du cas précédent, le fait qu'un module ne soit pas intégré dans le noyau Linux standard est généralement lié à la qualité insuffisante du code de ce module. En raison du niveau de criticité du noyau pour la sécurité du système CLIP, il n'est pas acceptable d'intégrer dans ce dernier des modules de qualité insuffisante.
- Enfin, indépendamment de leur qualité, les modules externes au noyau Linux standard n'offrent aucune garantie de pérennité, ni surtout de maintien à jour au fur et à mesure des évolutions du noyau standard. Intégrer un tel module dans la distribution pourrait interdire une mise à jour future du noyau CLIP, en raison de l'incompatibilité du module externe avec la nouvelle version. Cette situation n'est pas acceptable au regard des contraintes de sécurité, qui imposent de pouvoir mettre à jour le noyau au plus vite en cas de vulnérabilité avérée.

Dans tous les cas, et indépendamment des contraintes énumérées ci-dessus, il n'est **en aucun cas acceptable d'inclure dans le système CLIP un module noyau disponible uniquement sous forme binaire**[18].

Support d'un nouvelle catégorie de périphériques

Le support d'une nouvelle catégorie de périphériques, outre celles généralement supportées par le système CLIP, nécessite un travail complémentaire qui dépasse la simple intégration du pilote concerné dans le noyau CLIP. Il est en effet nécessaire dans ce cas d'intégrer la configuration et la mise en oeuvre du nouveau type de périphérique dans la couche utilisateur du système. Cette intégration concerne en général et au minimum les scripts de démarrage du poste (pour la configuration initiale du périphérique) et les interfaces d'administration (pour le paramétrage du périphérique par

[18] Cette contrainte ne s'applique pas aux *firmwares* binaires, dont le code est exécuté sur les périphériques auxquels ils sont associés, et non directement sur le processeur.

l'administrateur, si un tel paramétrage est nécessaire), ainsi éventuellement que les logiciels mis en oeuvre dans les cages d'utilisations (cages RM), pour la mise en oeuvre du périphérique. D'autres développements complémentaires peuvent s'imposer selon les cas, par exemple celui d'un démon spécifique si le reparamétrage à la volée du périphérique doit être possible, ou l'intégration dans le gestionnaire *hotplug* spécifique à CLIP si le branchement à chaud doit être supporté.

A titre informatif, les catégories de périphériques correctement supportées par le système CLIP au moment de la rédaction du présent document sont les suivantes :

- contrôleurs de disque (SATA, SCSI, IDE), contrôleurs RAID matériels (ainsi que RAID logiciel, uniquement dans les configurations passerelles) ;
- cartes réseau *ethernet* et cartes *wifi* ;
- cartes réseau téléphoniques 3G de marque *Option* (on notera que vu la grande hétérogénéité de ces types de périphériques, le support d'une carte du même type mais d'une autre marque entraînerait a priori des difficultés comparables au support d'une nouvelle catégorie de périphérique) ;
- cartes graphiques (voir aussi le paragraphe spécifique à ces dernières ci-dessous) ;
- cartes son (sortie et entrée son, dans une seule cage RM uniquement) ;
- supports de stockage de masse USB (clés USB, disques externes, lecteurs de cartes mémoire) ;
- CD-ROM et DVD-ROM (lecture seulement, pas de support de la gravure) ;
- imprimantes USB et réseau ;
- claviers / souris USB et équivalents.

La mise en oeuvre de cartes à puce pour l'authentification locale des utilisateurs n'est pas supportée à ce stade, mais devrait l'être dans une prochaine version. Tout comme pour les cartes téléphoniques 3G, l'hétérogénéité des interfaces de cartes à puce ne permet toutefois pas de garantir que le support d'un modèle de carte donné pourra être facilement étendu à d'autres modèles.

Périphériques d'affichage

Le système CLIP peut mettre en oeuvre deux méthodes d'affichage, selon le niveau de support matériel de la carte graphique :

- **affichage accéléré**, mettant en oeuvre l'infrastructure *Direct Rendering Manager* (DRM) du noyau et des pilotes spécifiques à la carte graphique au sein du serveur graphique ;
- **affichage non accéléré**, reposant uniquement sur les fonctionnalités VESA de la carte graphique, pilotée par le noyau en mode *framebuffer* VESA générique (*uvesafb*), avec un pilote générique *fbdev* dans le serveur X11.

Outre l'accélération graphique (rendus 2D et 3D accélérés matériellement), la première approche est la seule à permettre une gestion fine des différentes sorties vidéos (par exemple l'activation simultanée en mode « miroir » ou « bureau étendu » de deux sorties, typiquement LCD interne et port VGA d'un portable), et le changement de résolution à la volée. Une telle accélération n'est cependant possible que sur certains types de cartes graphiques : la carte doit être supportée par l'infrastructure DRM du noyau, et par des pilotes X11 *open-source*[19], et doit par ailleurs pouvoir être utilisée en mode *Kernel Mode*

[19] Tout comme pour les pilotes noyau, il n'est pas envisageable, pour des raisons fonctionnelles et de sécurité, de mettre en

Setting (KMS) par le noyau et le serveur X11, pour être compatible avec les réductions de privilèges imposées dans le système CLIP (le serveur X11 ne peut pas directement configurer le matériel). Ainsi, seules les cartes graphiques Intel (hors modèles *Poulsbo*), qui sont les mieux supportées par le KMS Linux, sont officiellement supportées en mode accéléré dans les configurations matérielles CLIP. Le support du mode accéléré est théoriquement possible, mais n'a pas été testé à ce stade, avec certains modèles de cartes ATI (modèles supportés par le KMS Linux) et NVIDIA (modèles supportés en mode KMS par les pilotes *Nouveau*, le support expérimental correspondant étant intégré dans le DRM du noyau CLIP). Le support de telles cartes nécessitera dans tous les cas l'ajout à la distribution CLIP des pilotes X11 associés.

En l'absence de support du mode accéléré, le système CLIP fonctionne par défaut en mode non accéléré, qui ne nécessite que le support des fonctionnalités VESA (2.0 et 3.0) standard par la carte graphique. Ce mode est également utilisé sur toutes les configurations matérielles normalement associées à des systèmes CLIP de type passerelle, pour lesquels l'accélération graphique ne présente pas d'intérêt fonctionnel.

Ressources minimales nécessaires

Le système CLIP n'est pas particulièrement gourmand en ressources, mais le fonctionnement multiniveau (deux environnements logiciels complets exécutés simultanément) et certaines spécificités de CLIP (partitionnement très fin, présence de deux systèmes CLIP installés côte-à-côte) peuvent avoir tendance à augmenter la consommation de mémoire ou d'espace disque. Les ressources à allouer à un poste CLIP sont à prévoir en fonction du niveau de performance attendu, mais on peut définir les contraintes minimales suivantes :

- Processeur compatible Intel 32 bits (*i686* ou plus), avec support du drapeau *NX* (*Intel*) ou *XD* (*AMD*) pour désactiver l'exécutabilité de pages mémoires. CLIP est capable d'utiliser à bon escient des processeurs ou coeurs multiples (fonctionnement en mode *Symetric Multi Processing*). En revanche, il ne peut pas fonctionner en mode 64 bits, pour les processeurs qui supportent un tel mode.
- La quantité de mémoire nécessaire sur un poste client est fonction du nombre de cages RM à mettre en oeuvre : 512 Mo peuvent suffire avec une seule cage RM, mais un minimum de 1 Go est recommandé pour l'utilisation simultanée de deux cages RM. La quantité de mémoire nécessaire aux passerelles est essentiellement fonction du nombre de clients avec lesquels elles seront appelées à établir des tunnels IPsec.
- L'installation d'environnements logiciels multiples côte-à-côte est également consommatrice d'espace disque. Ainsi, tandis que 10 Go suffisent à installer un système passerelle (hors partition de données utilisateurs et partition de stockage des journaux), un système CLIP-RM complet, avec deux installations alternatives et chacune deux cages RM, occupera 40G, là encore sans données utilisateur ni journaux. Il est donc nécessaire dans une telle configuration de disposer d'un espace disque d'au moins 60 Go.

oeuvre dans CLIP des pilotes X11 disponibles uniquement sous forme binaire, par exemple les pilotes propriétaires ATI ou NVIDIA.

5.2 Environnement réseau

Comme évoqué plus haut (cf. section 3), un poste client CLIP-RM peut sans problème fonctionner sans accès au réseau. Dans ce cas, les mises à jour doivent être mises à disposition du client par supports amovibles, et l'import et l'export de documents utilisateurs n'est possible que par supports amovibles également (le cas échéant couplés à la mise en oeuvre de la diode cryptographique du poste). Les configurations passerelles n'ont en revanche naturellement pas vocation à être déployées hors-ligne.

Méthodes et paramètres de connexion

Lorsqu'un accès au réseau est souhaitable dans un déploiement, celui-ci est possible de trois manières différentes :

- accès réseau filaire, grâce à une connectique *ethernet* ;
- accès réseau sans fil à travers une interface *wifi* ;
- accès réseau sans fil à l'aide d'une carte réseau téléphonique 3G (seules les cartes de marque *Option* sont supportées à ce stade).

Lors d'un accès réseau filaire, les paramètres réseau du poste peuvent être définis statiquement, ou bien obtenus dynamiquement par *dhcp*, ou encore par une combinaison de paramètres dynamiques et statiques (par exemple adresse IP obtenue par *dhcp*, mais route par défaut ou serveurs DNS définis statiquement). Ces différentes options de configuration peuvent être également mises en oeuvre pour une connexion *wifi*, même si une configuration entièrement dynamique est généralement la norme pour ce type de connexion. En revanche, aucun paramètre de connexion ne peut être défini statiquement pour une connexion 3G, à l'exception des paramètres 3G eux-mêmes (code PIN de la carte SIM, identifiant opérateur, etc.).

Les connexions *wifi* peuvent être non protégées, ou protégées par un chiffrement WEP (clés de 40 ou 104 bits), WPA/WPA2 (chiffrement TKIP ou CCMP, authentification par mot de passe PSK ou par certificats client et serveur EAP-TLS). La mise en oeuvre d'une authentification de type RADIUS sur les connexions filaires, bien que prévue à terme, n'est pas supportée à ce stade.

Il est à noter qu'un poste CLIP-RM peut mettre en oeuvre plusieurs configurations DNS simultanément : on aura typiquement des configurations DNS distinctes dans chacune des cages RM et dans la cage UPDATE du socle CLIP, de manière logique dans la mesure où les réseaux auxquels accèdent ces cages sont distincts. Lorsque la configuration DNS est obtenue dynamiquement par *dhcp*, une seule de ces différentes configurations est affectée, les autres restant statiques. Le choix de la configuration DNS obtenue par *dhcp* est un paramètre d'installation du poste ; il s'agit typiquement de RM_B (seule à avoir accès directement au réseau en clair). Les autres composantes du système (autres cages CLIP, coeur CLIP) ne mettent pas en oeuvre de résolution DNS (autre que la résolution statique de */etc/hosts*).

Les possibilités de configuration du routage se limitent à la définition d'une passerelle par défaut sur les postes clients. La possibilité de définir des routes spécifiques autres que la route par défaut est prévue sur les configurations passerelles, mais n'a pas encore été implémentée au moment de la rédaction du présent document. Le système CLIP met en oeuvre le protocole ARP de manière standard pour résoudre les adresses IP. La définition de correspondances statiques entre adresses IP et MAC n'est pas supportée.

Les différents paramètres réseau d'un poste CLIP sont regroupés dans un *profil* réseau, l'administrateur et les utilisateurs privilégiés ayant la possibilité de définir plusieurs profils distinct et de basculer de l'un à l'autre, comme détaillé en section 6.3.

Environnement réseau

Le système CLIP n'impose pas de contrainte particulière sur son environnement réseau en dehors de la mise en oeuvre de tunnels IPsec. Les tunnels mis en oeuvre peuvent être de deux types :

- Tunnel de téléchargement des mises à jour, accessible depuis la cage UPDATE du poste. La mise en oeuvre de ce tunnel nécessite le déploiement d'une passerelle CLIP en configuration UPDATE (cf. 3.3), dont l'interface externe est joignable depuis le réseau de déploiement des postes clients, et d'un serveur HTTPS de téléchargement de mises à jour connecté à l'interface interne de cette passerelle, laquelle doit autoriser les flux HTTPS depuis le réseau client (adresses des cages UPDATE des clients) vers le serveur. Un serveur NTP peut également être colocalisé avec le serveur HTTPS pour assurer la synchronisation horaire des clients, auquel cas la passerelle doit également autoriser les flux NTP depuis les clients vers la passerelle.
- Tunnel d'accès aux services réseau depuis la cage RM_H, dont la mise en oeuvre nécessite le déploiement d'une passerelle CLIP en configuration RM, dont l'interface externe doit être joignable depuis le réseau de déploiement des postes clients, et d'un ou plusieurs serveurs hébergeant les services réseau (messagerie, annuaire, etc.) connectés à l'interface interne de cette passerelle. Les flux à autoriser sur la passerelle sont fonction des services réseau mis en oeuvre. Si une résolution de nom dynamique est nécessaire dans les cages RM_H des clients, le serveur DNS devra également être placé du côté de l'interface interne de la passerelle RM_H.

Lorsque des tunnels IPsec sont mis en oeuvre, les éventuels pare-feux utilisés sur le réseau de déploiement ne doivent pas bloquer les flux UDP associés aux négociations IKE entre les clients et les passerelles[20], ainsi que les flux IPsec (ESP) entre ces mêmes machines. La mise en oeuvre de NAT (aussi bien source – typiquement pour les clients – que destination – typiquement pour les passerelles) ne pose pas de problème particulier : les tunnels passeront automatiquement dans ce cas en mode *NAT Traversal* avec encapsulation des flux ESP dans des flux UDP de ports source et destination 4500. Il n'est pas nécessaire dans ce cas d'autoriser la circulation de paquets ESP : seuls des paquets UDP seront émis sur le réseau.

5.3 Installation de postes CLIP

Procédure d'installation

La procédure d'installation d'un poste CLIP se veut aussi simple et surtout aussi automatique que possible. Elle a vocation à faciliter autant que faire se peut l'installation parallèle de nombreux systèmes CLIP. L'installeur CLIP se présente comme un support amovible *bootable*, de type CD-ROM/DVD-ROM ou clé USB. Il incorpore un ou plusieurs miroirs complets de paquetages CLIP, ainsi

[20] Les flux correspondants sont à l'origine entre les ports 500 et 4500 des clients et les ports 500 et 4500 des passerelles. On prendra cependant garde aux effets d'éventuelles NAT qui pourraient faire « flotter » les ports sources. Par ailleurs, bien que les clients soient normalement initiateurs des négociations, il n'est pas recommandé d'imposer ce sens de connexion dans le filtrage réseau : une fois une négociation établie, une passerelle peuvent contacter un client de sa propre initiative, par exemple pour vérifier qu'il est toujours actif, et un tel contact peut être vu par des pare-feux comme une nouvelle connexion initiée par la passerelle.

que les scripts d'installation permettant d'initialiser un poste CLIP. Le même installeur peut permettre aussi bien l'installation de postes CLIP-RM que de passerelles CLIP-GTW, auquel cas deux miroirs de paquetages (un pour chaque distribution) doivent être présents sur le support. En revanche, le fonctionnement normal de l'installeur ne permet pas d'installer alternativement deux versions différentes d'une même distribution à partir du même support. Au terme de la séquence de démarrage sur le support amovible, l'installeur CLIP présente une console texte ouverte sous l'identité *root*, depuis laquelle l'installation peut être lancée par une unique commande *full_install.sh*, assortie d'options obligatoires permettant de définir :

- le type de système à installer, CLIP-RM ou CLIP-GTW (UPDATE ou RM_H) ;
- le disque (ou les disques en cas de mise en oeuvre de RAID logiciel) sur lequel réaliser l'installation ;
- la configuration matérielle à installer (cf. 5.1 ci-dessus) ;
- le chemin de l'arborescence de configuration (décrite plus bas) à utiliser pour l'installation.

Une fois l'installation lancée, l'installeur propose à l'utilisateur un choix de partitionnement du disque (tailles allouées aux différentes partitions). Une fois ce choix validé, l'installation se poursuit de manière entièrement automatique, sans interaction avec l'utilisateur. Elle consiste à initialiser les différentes partitions correspondant aux deux systèmes CLIP à installer côte-à-côte (afin de supporter le mécanisme de mise à jour du coeur CLIP, cf. 4.3.2), à y installer les différents paquetages constituant ces systèmes, puis à appliquer une phase de configuration initiale de ces systèmes, à partir des paramètres fournis par l'arborescence de configuration, comme décrit plus bas. Au terme de l'installation, un simple redémarrage donne accès au système CLIP nouvellement installé, sur lequel un compte utilisateur initial, disposant des privilèges d'administration, a été automatiquement créé. Ce compte peut être utilisé pour créer les autres comptes utilisateurs du poste, avant d'être supprimé. Cette procédure d'installation est décrite plus en détail dans le document de référence [CLIP_2001]. Le poste CLIP fraîchement installé dispose de deux installations complètes du système, qui sont initialement dans la même version.

Arborescences de configuration

La pré-configuration du poste par l'installeur pendant et après l'installation des paquetages est réalisée sur la base d'une **arborescence de configuration**, arborescence de fichiers propre au poste considéré et contenant un certain nombre de paramètres à appliquer au poste. Ces paramètres se répartissent entre les **paramètres dynamiques**, qui sont ceux qui sont ensuite modifiables par l'administrateur du poste pendant son fonctionnement normal, et les **paramètres d'installation**, qui ne peuvent plus être modifiés après l'installation du poste (sauf éventuellement à travers une mise à jour).

Les paramètres dynamiques configurables à l'installation comprennent notamment un ou plusieurs profils réseau (cf. 6.3), dont au moins le profil par défaut (*default*), ainsi éventuellement qu'une liste initiale de paquetages optionnels à installer dans les cages RM. S'y ajoutent une liste de comptes utilisateurs à créer initialement (en pratique, un script à exécuter dans le système fraîchement installé pour créer ces comptes), généralement réduite à un unique compte administrateur (*config*) qui sera utilisé pour créer ultérieurement les autres comptes.

Les principaux paramètres dynamiques sont les suivants :

- nombre et désignations des cages RM à créer, pour un poste CLIP-RM uniquement ;

- paramétrage de la gestion des utilisateurs : autorisation ou non de créer des comptes utilisateurs privilégiés (cf. 6.1) ;
- paramétrage de la gestion des supports amovibles : liste de cages dans lesquelles est autorisé le montage de supports USB signés mais non chiffrés, et liste similaire des cages où des supports USB non initialisés peuvent être montés en lecture-écriture (cf. 6.2) ;
- paramétrage de la gestion de certains périphériques (poste CLIP-RM) : cage RM dans laquelle est exposée la carte son, cage RM dans laquelle est exposée une imprimante USB ;
- liste de bibliothèques CCSD dont l'utilisation par la diode cryptographique est autorisée (cf. 2.2.1).

Outre ces différents éléments de paramétrage, une arborescence de configuration peut également contenir un certain nombre d'éléments cryptographiques :

- clés de vérification des signatures développeur et validateur des paquetages (équivalent des clés publiques des centres d'élaboration des clés, cf. 4.2) - paramètre d'installation ;
- clés publiques RSA utilisées pour l'export des clés de chiffrement et de signatures des supports amovibles USB (une par niveau, CLIP, RM_H et RM_B) – paramètre d'installation ;
- clés publiques et privées ACID pour l'authentification des tunnels IPsec, si ceux-ci doivent être mis en oeuvre, soit :
  - clé privée du poste et son mot de passe – paramètre dynamique ;
  - clés publiques des passerelles avec lesquelles le poste doit établir des tunnels – paramètre d'installation ;
  - clés publiques des clients avec lesquels le poste doit établir des tunnels (uniquement pour une configuration passerelle) – paramètre dynamique ;
- certificat de l'autorité de certification associée aux téléchargement HTTPS (autorité certifiant le certificat du serveur HTTPS), si de tels téléchargements doivent être mis en oeuvre – paramètre dynamique ;

On notera que le seul élément « secret » parmi ces différentes clés est la clé privée IPsec et son mot de passe. Comme ceux-ci constituent des paramètres dynamiques, ils peuvent être omis de l'arborescence de configuration et ajoutés ultérieurement au poste, ce qui permet de transmettre des arborescences de configuration sans éléments secrets. La génération de ces différents éléments cryptographiques est décrite dans la section suivante.

Les différentes arborescences de configuration peuvent selon les cas être stockées sur le support d'installation lui-même, ou sur un support distinct, qui doit dans ce cas être monté après le démarrage sur le support d'installation, et avant le lancement de l'installation.

Génération des éléments cryptographiques

Un déploiement CLIP peut nécessiter la mise en oeuvre d'une ou plusieurs Infrastructures de Gestion des Clés (IGC), selon le cadre exact du déploiement. Il s'agit plus précisément :

- d'un Centre d'Elaboration des Clés (CEC) ACID v7 pour la génération des clés IPsec (une par poste), si des tunnels IPsec doivent être mis en oeuvre (et sauf utilisation d'algorithmes cryptographiques civils, qui nécessiteront alors leur propre IGC) ;

- d'un CEC ACID v7 pour la génération de clés validateur (une par validateur) pour la signature des mises à jour, si une validation spécifique des mises à jour doit être réalisée ;
- d'un CEC ACID v7 pour la génération de clés développeur (une par développeur) pour la signature des mises à jour, si une signature développeur spécifique doit être mise en oeuvre (ce qui n'a de sens que si une validation spécifique est également mise en oeuvre) ;
- d'une IGC SSL pour la signature du certificat HTTPS du serveur de téléchargement des mises à jour, si des téléchargements HTTPS sont mis en oeuvre ;

Les clés RSA utilisées pour l'export des clés de chiffrement et de signature des supports amovibles USB sont de simples bi-clés, sans infrastructure de certification associée.

5.4 Administration et supervision des postes

L'administration (configuration des paramètres ajustables du poste) et la supervision (lecture des journaux) des postes CLIP se font à ce stade **uniquement de manière locale au poste**, depuis les cages ADMIN et AUDIT, respectivement, du socle CLIP. Cependant, le fait que celles-ci reposent en pratique sur des connexions SSH locales depuis la cage USER à destination des cages ADMIN et AUDIT permet d'envisager sans difficulté majeure une transposition des fonctionnalités d'administration et de supervision à un fonctionnement à distance, dans lequel un poste distant, par exemple d'administration, viendrait directement ce connecter en SSH à la cage ADMIN d'un poste CLIP, à travers un tunnel IPsec spécifique. A cette fin, les scripts de configuration réseau CLIP prévoient dès à présent la définition d'une troisième passerelle IPsec, permettant d'établir des tunnels IPsec entre des postes distants et les trois cages locales ADMIN, AUDIT et USER (tunnel IPsec commun pour les flux d'administration et de supervision). Cette configuration n'a cependant jamais été testée à ce stade. On pourrait également envisager, moyennant une légère adaptation, un déport systématique des journaux du poste CLIP à travers ce même tunnel IPsec d'administration / supervision.

Les rôles d'administration et de supervision peuvent être selon les cas associés à deux comptes utilisateur distincts (de profils administrateur et auditeur, respectivement, cf. 6.1), ou combinés dans un unique compte utilisateur privilégié. Les principales tâches d'administration et de supervision sont réalisables à l'aide d'interfaces graphiques spécifiques, accessibles à travers des menus dans la session graphique CLIP de l'utilisateur concerné. La mise en oeuvre de la ligne de commande est également possible comme alternative à ces interfaces graphiques.

Les paramètres administrables localement concernent plus spécifiquement :

- la configuration du réseau : définition de nouveaux profils réseaux (cf. 6.3), ajustement dans chaque profil du type d'interface utilisé pour la connexion, des différents paramètres de connexion (adresse IP, DNS, etc.), et du filtrage des flux (ajout ou suppression de ports autorisés en sortie pour chaque cage) ;
- la gestion des mises à jour : définition des miroirs sources de téléchargement (cf. 4.3.1), sélection de paquetages optionnels à installer localement, lancement explicite d'un téléchargement de mise à jour ;
- la gestion des comptes utilisateurs : création ou suppression de comptes, verrouillage temporaire de comptes existants ;
- la configuration de l'heure du système et d'une éventuelle synchronisation NTP périodique ;

- la gestion des éléments secrets du poste : installation ou mise à jour de la clé privée ACID v7 pour IPsec, ajout ou suppression de clés publiques IPsec de clients (passerelles uniquement), mise à jour du certificat de l'autorité de certification HTTPS pour les téléchargement de mises à jour ;
- configuration de l'archivage des journaux.

Les journaux consultables par le rôle de supervision sont quant à eux répartis entre différents fichiers, selon leur provenance (socle CLIP, cages RM_H ou RM_B) et leur nature (journaux des tâches périodiques, alertes levées par les différents mécanismes de sécurité, journaux d'authentification des utilisateurs, etc.). Les prérogatives du rôle de supervision se limitent à ce stade à la lecture de ces différents fichiers : aucun outil d'analyse plus avancé des journaux n'est déployé pour l'instant sur les postes CLIP.

6 Utilisation d'un poste CLIP

6.1 Types de comptes utilisateur

Plusieurs comptes utilisateurs peuvent être définis sur un même poste CLIP, chacun disposant de son jeu de partitions chiffrées pour le stockage des données. Une seule session utilisateur peut être ouverte à un moment donné, les données des autres utilisateurs restant inaccessibles dans le cadre de cette session.

Chaque compte utilisateur CLIP est caractérisé par son type, ou rôle, qui définit ses privilèges et ses prérogatives. Les rôles supportés par le poste CLIP sont les suivants :

- **Utilisateur simple :** permet le lancement de session dans les cages RM d'un système CLIP-RM (ce rôle n'a pas vraiment de sens sur un système CLIP-GTW), sans accès aux fonctionnalités d'administration ni de supervision.
- **Administrateur :** dispose des seuls privilèges d'administration, permet de configurer les paramètres dynamiques du poste, mais ne donne accès ni aux éventuelles cages RM, ni à la lecture des journaux. Le privilège d'administration correspond en pratique à l'accès à une clé privée SSH (dans la partition chiffrée de niveau CLIP de l'utilisateur) qui permet la connexion à la cage ADMIN du poste.
- **Auditeur :** dispose des seuls privilèges de supervision, permet de consulter les journaux du système, mais ne donne accès ni aux éventuelles cages RM, ni à l'administration du poste. Le privilège d'administration correspond en pratique à l'accès à une clé privée SSH (dans la partition chiffrée de niveau CLIP de l'utilisateur) qui permet la connexion à la cage AUDIT du poste.
- **Utilisateur privilégié :** combine les privilèges et prérogatives des trois comptes précédents, permet de configurer les paramètres dynamiques du poste, de consulter les journaux, et de lancer des sessions dans les éventuelles cages RM (contrairement au rôle d'utilisateur simple, ce rôle peut avoir un sens sur un système CLIP-GTW, pour combiner les rôles d'administrateur et d'auditeur).
- **Utilisateur nomade :** cas particulier d'utilisateur privilégié, disposant en pratique des mêmes privilèges, mais avec des menus d'administration bridés lui permettant uniquement de configurer le réseau et la date du système (mais pas par exemple de gérer les mises à jour ou les comptes utilisateurs). Ce type de compte est adapté au cas d'un système multi-utilisateur utilisé dans un cadre nomade, pour lequel chaque utilisateur doit être en mesure de reconfigurer le réseau en fonction de sa localisation géographique, sans pour autant avoir nécessairement toutes les prérogatives d'un administrateur.

Chaque utilisateur se voit présenter une interface commune lors de l'ouverture de session CLIP, consistant essentiellement en une barre de menu *fbpanel*[21] (cf. 3.1). Les menus inclus dans cette barre sont cependant adaptés en fonction du rôle utilisateur, pour par exemple intégrer ou non un menu d'administration ou des icônes de lancement de session RM. Un certain nombre d'opérations ne sont

[21] Une cage RM peut également être lancée automatiquement à l'ouverture de session pour certains rôles sur un poste CLIP-RM à une seule cage RM.

pas considérées comme privilégiées et peuvent être réalisées par n'importe quel utilisateur : configuration de l'affichage (activation de la sortie vidéo externe par exemple), changement du délai de verrouillage de session (uniquement valable pour la session courante), changement de mot de passe de l'utilisateur courant[22], arrêt ou redémarrage du poste.

L'espace alloué pour les partitions utilisateur est défini lors de la création du compte utilisateur, et ne peut à ce stade pas être modifié après. Les comptes administrateurs et auditeurs, qui n'ont pas vocation à manipuler des données applicatives, se voient attribuer une partition CLIP de taille fixe et minimale (8 Mo) et pas de partition RM. La partition CLIP n'est pas directement accessible de l'utilisateur, mais permet de stocker pour lui certains éléments, notamment ses clés SSH. Les profils utilisateur (y compris utilisateurs privilégiés et nomades) se voient également attribuer une partition CLIP de 8 Mo, mais disposent par ailleurs de partitions RM de taille quelconque.

6.2 Gestion des supports amovibles

Types de supports USB

Le système CLIP permet la mise en oeuvre de plusieurs types de supports de stockage amovibles USB, qui sont utilisables au sein des différentes cages interactives du système (ADMIN, AUDIT, USER, RM_H et / ou RM_B dans le cas d'un poste CLIP-RM), ainsi que de manière indirecte dans la cage non interactive UPDATE, pour la mise à disposition de mises à jour (cf. 4.3.1). On distinguera notamment des supports USB qui ont été spécifiquement initialisés par le poste CLIP considéré, de supports standards n'ayant pas fait l'objet d'une initialisation particulière. Une politique de sécurité définie à l'installation du poste, et non modifiable ensuite (paramètre d'installation, cf. 5.3), permet de définir les modalités d'utilisation des différents types de supports. Plus précisément, les trois types de supports reconnus par le système CLIP sont les suivants :

- **Supports signés :** Ces supports sont nécessairement initialisés sur un poste CLIP. Ils sont **associés par une signature cryptographique à un niveau** (CLIP, RM_H ou RM_B) et à **un utilisateur uniques** (cf. 2.2.4). Ils peuvent être montés en lecture-écriture au sein de l'environnement CLIP, uniquement dans la cage correspondant à leur niveau, et sont également susceptibles d'être montés comme des supports USB standard (formatés en FAT32) sur des postes non CLIP. Ils permettent ainsi l'import et l'export de fichiers entre le poste CLIP et des postes tierce partie. La politique définie à l'installation du poste spécifie explicitement les niveaux pour lesquels la mise en oeuvre de supports signés est autorisée. En l'absence de chiffrement du contenu de ces supports, toute information stockée sur un support perdu ou branché par erreur sur un poste de niveau inadapté doit être considérée comme compromise.
- **Supports chiffrés signés :** Ces supports sont similaires aux supports signés, à la différence près que leur **contenu est également chiffré**, de manière à en garantir la confidentialité en cas de perte ou de branchement par erreur à un poste de niveau différent. Le montage de supports amovibles chiffrés signés sur un poste autre que CLIP nécessite des pilotes et utilitaires spécifiques, ainsi que l'importation sur le poste des clés cryptographiques associées au support. La mise en oeuvre de tels supports, en l'absence de risque de compromission du fait du chiffrement, est systématiquement permise à tous les niveaux.

[22] La mise en oeuvre d'un chiffrement de partitions basé sur le mot de passe fait que l'ancien mot de passe est absolument indispensable au changement de mot de passe. De ce fait, il est rigoureusement impossible, même pour un administrateur, de réinitialiser le mot de passe d'un utilisateur qui aurait oublié son ancien mot de passe.

- **Supports non initialisés :** Les supports non initialisés correspondent à des supports USB standards, au format FAT32, qui n'ont pas fait l'objet d'une initialisation spécifique par un poste CLIP. Ces supports peuvent être montés à n'importe quel niveau au sein d'un système CLIP, au choix de l'utilisateur. En revanche, ce montage sous CLIP est par défaut réalisé en **lecture seule**, ce qui interdit toute écriture de fichier sur le support depuis un poste CLIP. Ainsi, ces supports ne peuvent être utilisés que pour l'import d'informations au sein du système CLIP, et en aucun cas pour l'export (et potentiellement la compromission) d'informations sensibles issues du poste. La politique définie à l'installation peut néanmoins autoriser le montage en lecture-écriture de tels supports dans certaines cages.

Ces différents types sont compatibles a priori avec n'importe quel support de stockage amovible USB, qu'il s'agisse de clés USB, de disques externes, ou encore de lecteurs de cartes mémoires (ces derniers ne sont néanmoins pas testés à ce stade).

Les supports amovibles USB, quel que soit leur type, sont montés automatiquement dans la cage de niveau approprié (le cas échéant après choix de ce niveau par l'utilisateur, pour un support non initialisé), lors de leur branchement et après confirmation par l'utilisateur courant du système.

Gestion des clés cryptographiques

La mise en oeuvre de supports USB initialisés nécessite de disposer de clés cryptographiques pour signer, et le cas échéant chiffrer et déchiffrer, les supports. Le système utilise à cette fin un bi-clé RSA par usage (chiffrement ou signature), par utilisateur et par niveau (CLIP, RM_H ou RM_B), généré localement sur le poste. La génération de clés n'est pas réalisée automatiquement à la création du compte utilisateur, mais doit être demandée explicitement par l'utilisateur, qui choisit alors un mot de passe maître protégeant ses clés privées de signature et de chiffrement. Ces clés sont ensuite stockées dans la partition chiffrée de niveau CLIP de l'utilisateur, qui n'y a par conséquent pas directement accès.

Les clés cryptographiques de l'utilisateur peuvent par ailleurs être exportées à la demande de l'utilisateur sur un support amovible non initialisé, typiquement afin de pouvoir déchiffrer et monter sur un autre poste (non CLIP[23]) les supports signés chiffrés initialisés sur le poste CLIP. Lors de cet export, les clés exportées sont automatiquement chiffrées en utilisant les clés publiques d'export de clés, propres au poste et fournies à l'installation de ce dernier (cf. 5.3). Un utilisateur disposant des clés privées associées à ces clés publiques (non présentes sur le poste CLIP) pourra ensuite déchiffrer les clés exportées.

Supports de type CD-ROM

CLIP permet également l'utilisation de supports de type CD-ROM et DVD-ROM, qui peuvent être montés en lecture seule dans les mêmes cages que les supports amovibles USB, le choix de la cage étant laissé à l'utilisateur. Il n'y a pas de politique de sécurité paramétrable associée aux supports amovibles de ce type : le système CLIP n'offrant pas la possibilité de graver un tel support, il n'y pas de risque de compromission des données sensibles manipulées sur le poste.

[23] Un ensemble de scripts permet ainsi le montage d'un support chiffré signé CLIP sur un poste Linux standard disposant des clés nécessaires. En revanche, le pilote permettant de réaliser un tel montage sur un poste Microsoft Windows, écrit pour l'implémentation initiale des supports chiffrés signés CLIP, n'a pas été maintenu et n'est plus fonctionnel à ce jour.

6.3 Gestion de la configuration réseau

La configuration réseau du système CLIP est gérée par profils. Un profil réseau contient l'ensemble des éléments de configuration du réseau : type d'interface réseau à utiliser, paramétrage de la connexion (adresse IP, route par défaut, DNS, *proxys* éventuels) et définition des flux autorisés par le pare-feu local. Il est possible de définir plusieurs profils réseau alternatifs sur le même poste : un utilisateur disposant des privilèges d'administration (administrateur, utilisateur privilégié ou nomade, cf. 6.1) peut librement ajouter ou supprimer des profils, correspondant à autant de configurations réseau différentes. Ce principe de fonctionnement est particulièrement adapté à une utilisation nomade, dans laquelle l'utilisateur est appelé à mettre en oeuvre des moyens et paramètres de connexion variables selon sa localisation géographique.

Lorsque plusieurs profils réseau sont définis sur un poste, l'utilisateur est invité au cours de la séquence de démarrage à choisir le profil à utiliser initialement. Ultérieurement, en cours d'utilisation du poste, les rôles administrateurs, utilisateurs privilégiés ou et nomades peuvent demander explicitement à activer un nouveau profil, et ainsi changer de configuration réseau (par exemple pour basculer d'une connexion filaire à une connexion *wifi*). Le changement de configuration est réalisé de manière sécurisée, sans créer d'état intermédiaire plus permissif que les profils complets. On notera qu'il est également possible de réactiver le profil courant en cas de dysfonctionnement réseau (par exemple perte d'une connexion *wifi*, ou fonctionnement incorrect des tunnels IPsec[24]). Le profil courant est d'ailleurs susceptible d'être réactivé de manière automatique par le client *dhcp* du poste, quand il est employé pour obtenir dynamiquement les paramètres de connexion. Une telle réactivation automatique est notamment employée lorsque le renouvellement de bail *dhcp* nécessiterait un changement d'adresse IP, qui ne peut être réalisé de manière sécurisée qu'en réactivant complètement le profil réseau. Elle peut également être la conséquence d'une perte de signal *wifi,* entraînant l'échec d'une renégociation de bail *dhcp*.

L'installation du poste définit normalement un profil réseau particulier, nommé *default*, qui ne peut en aucun cas être supprimé (mais qui peut être modifié par un utilisateur disposant des privilèges d'administration). Ce profil est notamment celui qui est utilisé automatiquement au démarrage, lorsque aucun des profils possibles n'a été sélectionné après 30 secondes d'attente. Il est éventuellement le seul profil de connexion du poste, auquel cas aucun choix n'est proposé au démarrage.

[24] Un tel fonctionnement incorrect peut en effet résulter d'une incohérence entre les associations de sécurité IPsec du client et d'une passerelle, typiquement suite à la perte de certains paquets dans les échanges IKE. Il ne constitue pas un problème de sécurité, dans la mesure où le chiffrement des flux reste garanti – mais pas nécessairement leur déchiffrement. Ces situations sont automatiquement corrigées par les démons IKE, mais avec une certaine latence, tandis que la réactivation du profil réseau force explicitement une renégociation immédiate des associations de sécurité.

Annexe A Références

Remarque 5 : organisation de la documentation CLIP

Les documents descriptifs de CLIP produits par l'ANSSI se voient chacun associer un numéro identifiant unique sur quatre chiffres, ainsi qu'un numéro de révision. Les numéros identifiants sont attribués selon les principes d'organisation suivants :

- ***0001 à 0999** : descriptifs non détaillés du système CLIP et de ses environnements de déploiement, documents* Diffusion Restreinte – Spécial France.
- ***1001 à 1999 :** documents techniques détaillées, classifiés* Confidentiel Défense – Spécial France.
  - *1001 à 1099 : architecture globale (fonctionnelle, de sécurité, etc.)*
  - *1101 à 1199 : outils de développement*
  - *1201 à 1299 : noyau*
  - *1301 à 1399 : socle CLIP (y compris « cages CLIP »)*
  - *1401 à 1499 : cages d'utilisation (cages RM)*
  - *1501 à 1599 : configuration réseau*
  - *1601 à 1699 : environnement (serveurs, infrastructures de gestion de clés, etc.)*
- ***2000 à 2999 :** documents d'utilisation du système CLIP,* Diffusion Restreinte – Spécial France.
  - *2001 à 2099 : documents à destination des responsables de déploiements CLIP*
  - *2101 à 2199 : documents à destination des utilisateurs finaux (y compris administrateurs de leurs postes)*
- ***3000 à 3999 :** documents liés à l'évaluation (certification, qualification) du système CLIP*
  - *3001 à 3099 : documents* Confidentiel Défense – Spécial France
  - *3101 à 3199 : documents* Diffusion Restreinte – Spécial France

Documentation CLIP

[CLIP_1002]	*Documentation CLIP – 1002 – Architecture de sécurité*
[CLIP_1101]	*Documentation CLIP – 1101 – Génération de paquetages*
[CLIP_1104]	*Documentation CLIP – 1104 – Guide de développement* (**incomplet à ce jour**)
[CLIP_1201]	*Documentation CLIP – 1201 – Patch CLIP LSM*
[CLIP_1202]	*Documentation CLIP – 1202 – Patch Vserver*
[CLIP_1203]	*Documentation CLIP – 1203 – Patch Grsecurity*
[CLIP_1204]	*Documentation CLIP – 1204 – Privilèges Linux*

[CLIP_1205] *Documentation CLIP – 1205 – Implémentation CCSD en couche noyau*

[CLIP_1206] *Documentation CLIP – 1206 – Génération de nombres aléatoires*

[CLIP_1301] *Documentation CLIP – 1301 – Séquences de démarrage et d'arrêt*

[CLIP_1302] *Documentation CLIP – 1302 – Fonctions d'authentification CLIP*

[CLIP_1303] *Documentation CLIP – 1303 – X11 et cloisonnement graphique*

[CLIP_1304] *Documentation CLIP – 1304 – Socle et cages CLIP*

[CLIP_1305] *Documentation CLIP – 1305 – Gestion des mises à jour*
(**non rédigé à ce jour**)

[CLIP_1306] *Documentation CLIP – 1306 – Gestion des supports amovibles*
(**non rédigé à ce jour**)

[CLIP_1307] *Documentation CLIP – 1307 – Diodes cryptographique et montante*
(**non rédigé à ce jour**)

[CLIP_1401] *Documentation CLIP – 1401 – Cages RM*

[CLIP_1501] *Documentation CLIP – 1501 – Configuration réseau*

[CLIP_1502] *Documentation CLIP – 1502 – Racoon2*

[CLIP_1503] *Documentation CLIP – 1503 – Strongswan*
(**non rédigé à ce jour,** se référer au document 1502 pour la description des traitements cryptographiques, qui sont similaires)

[CLIP_2001] *Documentation CLIP – 2001 – Procédure d'installation*

[CLIP_2002] *Documentation CLIP – 2002 – Guide de configuration du BIOS*
(version adaptée au type de matériel concerné)

Autres références

[CC] *Common Criteria for Information Technology Security Evaluation, version 3.1*

[CCSD] *Couche Cryptographique pour la Sécurité de Défense*
Document d'Interface Client version 3.2

[CERTIF] *Certificat DCSSI-2009/12 du 28 avril 2009, Produit CLIP-RM, version 03.01.03*

[Gentoo] *Gentoo Linux* : http://www.gentoo.org

[QUALIF] *Qualification au niveau standard, CLIP-RM v03.01.03, n° 1170/SGDN/DCSSI/DR-SF du 11 mai 2009*

Annexe B Liste des figures

Annexe C Liste des remarques