Les réseaux privés virtuels

Labo-Unix - http://www.labo-unix.net

2001-2002

Table des matières

Droits de ce document

1 Introduction

Jusqu'à présent, il y a toujours eu une très claire différence entre les réseaux privés et publics. Un réseau public, comme le réseau téléphonique ou Internet, est un ensemble de points entre lesquels ont lieu des échanges plus ou moins libre d'informations. Les personnes qui ont accès à ce réseau sont susceptibles d'avoir un intéret commun et utilisent le réseau pour communiquer les uns avec les autres.
Un réseau privé est composé d'ordinateurs appartenant à une même et unique organisation qui partage ses informations en interne avec la certitude que seuls leurs employés utiliseront le réseau et que les informations en transit sur ce dernier ne pourrons être atteintes que par des membres du même groupe. Les réseaux locaux (LAN) et WAN sont des exemples typiques de réseaux privés. La ligne de séparation entre un réseau privé et un réseau public se trouve au niveau de la passerelle, où la compagnie place généralement un pare-feu pour éviter que des intrus d'un réseau public n'accède à leur réseau privé ou l'inverse.
Il y a peu de temps, les entreprises pouvaient encore se permettrent de construire leurs propres LAN, supportant leurs propres système de nommage, système de messagerie, voire même leur propre protocole réseau. Cependant, comme de plus en plus de données étaient stockées sur ordinateurs, les entreprises ressentirent le besoin d'interconnecter leurs différents bureaux. Grâce à l'utilisation de lignes dédiées, une entreprise avait à la garantie que la connexion entre ses départements seraient toujours disponible et privée. Cependant, cette solution peut être très couteuse, notament si l'entreprise à plusieurs bureaux à travers tout un pays.
De plus, les réseaux privés manque de souplesse par rapport aux situations que l'on peut rencontrer dans une entreprise. En effet, si un représentant a besoin d'accéder à distance au réseau privé de son entreprise alors qu'il est a des milliers de kilomètres de celle-ci, le coût de l'appel téléphonique sera extrêmement élevé.
Ce cours présentera les réseaux virtuels privés ou *VPN*, un concept qui rend flou la différence entre réseau privé et public. Les VPN vous permettent de créer un réseau privé et sûr sur un réseau public tel qu'Internet. On peut les réaliser à l'aide de matériels spécifiques (cartes réseaux, routeurs, ...), de logiciels ou d'une combinaison *hardware/software*.

2 Présentation générale

2.1 Rôle d'un VPN

Un grand nombre de protocoles ne protègent pas leurs informations en les chiffrant. Si nous prenons le cas de POP3, par exemple, on s'aperçoit alors que non seulement les messages électroniques circulent en clair sur le réseau mais également les noms d'utilisateurs et les mots de passe. Il est alors facile pour une personne malintentionnée d'écouter les communications entre client et serveur et de récupérer les informations qui l'intéresse. Sachant, de plus, qu'il existe un certain nombre de logiciels permettant d'automatiser la procédure (*dsniff* par exemple), c'est avec bien des tracas que l'on relève ses mails depuis un endroit peu sûr...
Bien sûr, il existe toujours la possibilité d'utiliser des versions spécifiques de ces protocoles. Mais, dans ce cas, on se heurte rapidement à des problèmes d'incompatibilités entre les applications clientes et serveurs.
La solution au problème qui vient d'être exposé est toute simple. Comme les protocoles peu sûrs ne peuvent être remplacés, il suffit de les utiliser sur un réseau protogé. L'astuce consiste alors à créer une connexion chiffrée entre le client et le serveur et c'est par l'intermédiaire de cette connexion que la liaison en clair s'effectuera. On crée ainsi un tunnel chiffré permettant la communication entre les machines.
Ce tunnel peut être créé sur plusieurs niveaux d'un réseau existant :

Au niveau réseau : les paquets circulants entre les machines sont chiffrés par un logiciel placé directement au-dessus du support réseau.

Au niveau transport : c'est la liaison logique entre des programmes fonctionnant de concert sur les machines qui chiffrent la communication. Ces programmes interceptent les communications les concernant et les traitent à la volée ; c'est le cas pour SSL par exemple.

Au niveau application : ici, c'est l'application elle-même qui protège les communications. Cette méthode est celle qui est offre le moins de compatibilité puisque client et serveur dialogue dans un "patois" qui leur est propre. Une application souhaitant être compatible avec la méthode de communication utilisée doit être calquée sur les spécifications de l'applications d'origine. Tous les logiciels et les plugins des clients mails entrent dans cette catégorie.

Un VPN est une généralisation du concept de tunnel. Il s'agit de faire circuler dans le tunnel non seulement les informations des applications, mais également tout ce qui concerne les couches réseau et transport. Ainsi, on crée un nouveau réseau à l'intérieur du tunnel. Vous imaginerez facilement les avantages d'un VPN en considérant 2 réseaux locaux ayant besoin d'être reliés via un réseau peu sûr comme Internet. Les 2 LAN se relient par un tunnel tranversant Internet et forment un nouveau LAN virtuel et sûr : le VPN.

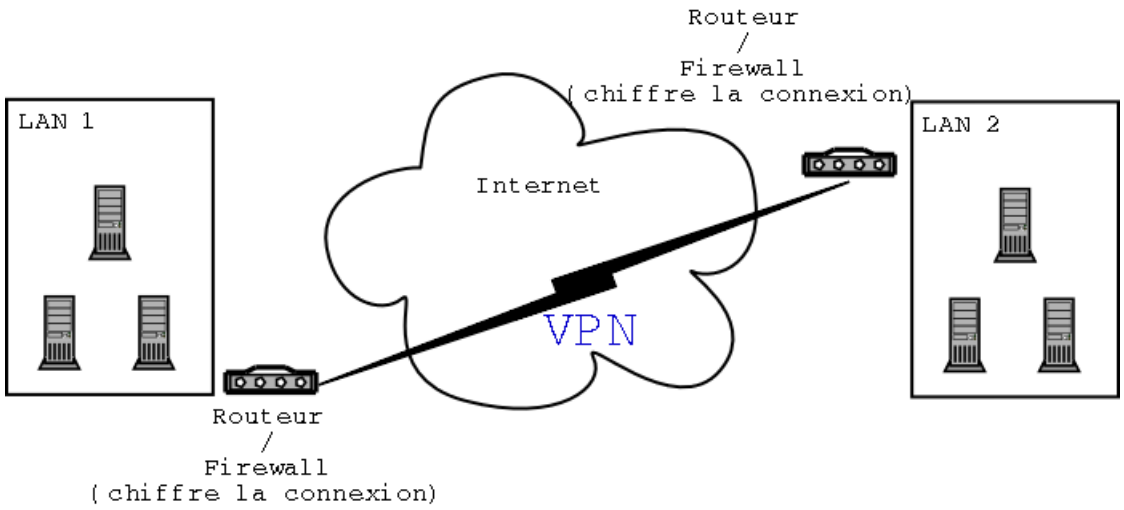

FIG. 1 – Exemple de réseau privé virtuel

Attention, il faut bien comprendre cependant que même si les données (application + réseau) circulant dans le VPN restent inaccessibles depuis l'extérieur, elles continuent à circuler en clair. Ainsi, si une personne malintentionnée se trouve dans le VPN, elle pourra toujours écouter et analyser les informations y circulant.

Avec un système de type Unix (comme GNU/Linux), le VPN utilise un tunnel IP *over* IP, c'est-à-dire que le réseau virtuel est un réseau IP fonctionnant sur un réseau IP. Mais il est également possible de faire transiter n'importe quel type de protocole réseau dans un VPN. Vous pouvez, par exemple interconnecté 2 réseaux IPX via un réseau IP tel qu'Internet.
Concernant le VPN en IP, il est d'usage d'utiliser des plages d'adresses réservées aux LAN (comme 192.168.0.0/16 ou 10.0.0.0/8) afin d'éviter toute confusion entre les adresses IP du VPN et celles d'Internet.

2.2 Vers un VPN sécurisé

Un VPN utilise un ensemble de technologies pour protéger des données qui voyagent d'un bout à l'autre de l'Internet. Les concepts les plus importants sont ceux de *firewall*, d'*authentification*, de *tunnels* et du *chiffrement de données* que nous allons présenter dans les parties suivantes.

2.2.1 Les firewalls

Un firewall Internet a le même but qu'une porte coupe feu dans un immeuble : protéger une certaine zone de l'avancée des flammes ou d'une explosion qu'elles pourraient engendrer. L'avancée des flammes dans un immeuble est contrôlée en plaçant de solides murs à des endroits stratégiques qui aident à contenir les flammes et à réduire les dégats occasionés. Un pare-feu internet a le même rôle : en utilisant des techniques telles que l'examen de l'adresse IP du paquet qu'il reçoit ou le port sur lequel arrive une connexion il décide de laisser passer ou de bloquer le trafic entrant.

Bien que VPN n'implemente pas de firewall standard par défaut, les pares-feu font partie intégrante d'un VPN. L'idée est qu'il doivent être utilisés pour garder les utilisateurs non désirables hors du réseau tout en acceptant les utilisateurs du VPN. Si vous n'avez pas de firewall protégeant votre réseau, ne vous embétez pas à construire un VPN avant d'en avoir un !
Le pare-feu le plus classique est un pare-feu filtrant les paquets, qui bloquera l'accès à certains services (en fonction des ports) au niveau de la passerelle. De nombreux routeurs supportant les technologies VPN, tel que le routeur Cisco Private Internet Exchange (PIX), gèrent en natif ce type de filtrage. Un serveur Proxy est aussi un solution possible pour protéger un réseau en laissant accès aux services VPN. Ce type de serveur tourne généralement sur un système d'exploitation tel que Linux, OpenBSD, Windows ou Novell Netware.

2.2.2 Les tunnels

De nombreuses solutions en matière de VPN utilisent des tunnels pour créer un réseau privé sur un réseau public. C'est le cas notament de du PPtP de Microsoft, du Layer 2 Forwarding Protocol et d'IPSec. Les VPNs vous permettent ainsi de vous connectez à un réseau distant via l'Internet, qui est lui-même un réseau IP. Néanmoins, de nombreux réseau locaux d'entreprise n'utilise pas IP comme base exclusive (même si aujourd'hui le courant va dans cette direction). Des réseaux composés de serveurs Windows NT par exemple, pourraient utiliser NetBEUI, tandis que les serveurs Novell utilisent IPX. Le tunneling vous permet ainsi d'encapsuler un paquet à l'intérieur d'un autre paquet afin de résoudre les problèmes de protocoles incompatibles. De ce fait, le paquet à l'intérieur d'un paquet peut être du même protocole ou d'un protocole complètement différent. Par exemple, le tunneling peut être utilisé pour envoyer des paquets IPX à travers l'Internet pour se connecter à un serveur distant Novell ne supportant que l'IPX. Le serveur distant décapsulera le paquet IP reçu, analysera le paquet IPX qu'il contient et enverra à son tour un paquet IPX encapsulé à l'utilisateur distant.
Grâce au tunneling, vous pouvez également encapsuler un paquet IP à l'intérieur d'un paquet IP. Ceci signifie que vous pouvez envoyer des paquets IP ayant une adresse source et une adresse de destination complètement arbitraire sur Internet en l'encapsulant dans un paquet qui possède une source et une destination joignable sur Internet. L'aspect pratique de la chose est que vous pouvez ainsi utiliser les adresses IP reservées par l'Internet Assigned Numbers Authority (IANA) aux réseaux privés sur votre LAN et continuer d'avoir accès à vos hosts à travers Internet.

2.2.3 Les techniques d'authentification

Les techniques d'authentification sont essentielles au bon fonctionnement d'un VPN puisqu'elles assurent aux utilisateurs d'un VPN qu'ils effectuent un échange de données avec le bon partenaire. L'authentification dans un VPN est semblable à l'authentification sur un système à l'aide d'un nom d'utilisateur et d'un mot de passe. Cependant les méthodes d'authentification des VPN sont souvent bien plus rigoureuses et compliquées. La plupart des systèmes d'authentification s'appuye en effet sur un échange de clefs entre les hôtes. Chaque hôte possède un jeu de clefs, composé d'une clef publique et d'une clef privée. Les clefs parcourent un algorithme de hashage qui produit un valeur hashé de ces

dernières. A l'autre bout de la connexion, l'hôte possédant les mêmes clefs réalise un hash identique, recupère et compare la valeur de hash retournnée à celle qu'il a recue. L'authentification réussie si les deux valeurs correspondent. Notez que la valeur de hash qui transite sur le réseau Internet n'a aucune signification pour un observateur extérieur qui la snifferait dans l'espoir de trouver un mot de passe. C'est de cette façon là que fonctionne le Challenge Handshake Authentification Protocol (CHAP). Une autre méthode commune d'authentification est le système RSA.
L'authentification a généralement lieu au début de la session, puis de manière aléatoire au cours de la session pour s'assurer qu'un imposteur ne se soit pas glisser dans la conversation. L'authentification peut aussi être utilisée pour assurée l'intégrité des données. En effet, une donnée elle-même peut être envoyée au travers de l'algorithme de hashage pour donner une valeur qui sera alors utiliser comme un checksum à l'intérieur d'un message. Si ce checksum n'est pas le même d'un hôte à l'autre, cela signifie que la donnée a été corrompue ou qu'elle a été interceptée et modifiée durant le transport.

2.2.4 Le chiffrement des informations

Tous les VPN supportent au moins un type de chiffrement qui a pour rôle essentiel de placer les données dans une enveloppe sécurisée. Le chiffrement est souvent considéré comme tout aussi important que l'authentification puisqu'il protège les données en cours de transport contre le sniffing. Il existe 2 grandes techniques de chiffrement employées dans la mise en place de VPN : le chiffrement à clef publique et le chiffrement à clef secrète (ou privée).
Dans le chiffrement à clef secrète, il existe un mot de passe ou une phrase connu de tous les hôtes souhaitant accéder à l'information chiffrée. Cette clef unique est utilisée à la fois pour chiffrer et pour déchiffrer l'information. Le système de chiffrement DES (*Data Encryption Standard*) est un exemple de méthode de chiffrement à clef privée.
Un problème posé par cette méthode de chiffrement par clef secrète est que chaque hôte souhaitant accèder à une donnée chiffrée doit connaitre le mot de passe. Si tout se passe bien pour un petit groupe de travail, cette soluton devient vite ingérable pour un réseau plus important. En effet, si un employé quitte la compagnie il vous faudra revoquer l'ancienne clef, en créer une nouvelle et arriver à mettre au courant tout le monde de manière sécurisée que la clef a changé.
Le chiffrement par clef publique suppose une clef publique et une clef privée. Vous transmettez votre clef publiqe à tout le monde tandis que vous ne connaisez que votre clef privée. Si vous souhaitez transmettre à quelqu'un des données sensibles, vous les chiffrer à l'aide de votre clef privée et de sa clef publique. Quand il recevra votre message, il pourra le déchiffrer à l'aide de sa clef privée et de votre clef publique. Selon l'application utilisée pour générer les clefs, ces dernières peuvent être grosses ; trop grosses pour que quelqu'un s'en souvienne par coeur. Ainsi, on les stocke sur la machine de la personne souhaitant chiffrer de données en utilisant cette méthode de chiffrement. A cause de cela, les clefs privées sont généralement stockées en utilisant un principe de chiffrement par clef secrète, tel que DES, et un mot de passe de sorte que si quelqu'un a accès à votre système il ne puisse pas voir ou utiliser votre clef privée. GnuPG est une application signant des données qui fonctionne sur ce modèle ; RSA est un autre system à clef publique que l'on retrouve dans de nombreux produits en vente dans le commerce. Le principal

désavantage du chiffrement par clef publique est que pour un montant équal de donnée à chiffrer, le processus de chiffrement est plus lent qu'avec une clef secrète.
Les VPN, quant à eux, doivent chiffrer les données en temps réel. A cause de cela, les flux chiffrés sur un réseau le sont en utilisant une clef secrète plutôt[1] qu'une clef publique. Néanmoins cette clef n'est valide que pour la liaison. La session secrète elle-même est chiffrée en utilisant le principe de clefs publiques. Les clefs secrètes sont négociées selon un protocole de gestion de clefs clairement défini.
La prochaine étape pour les VPN est l'adoption d'IPSec implémentant le chiffrement au niveau IP.

3 IPSec : IP Security Protocol

3.1 Introduction

Le terme IPsec (IP Security Protocol) désigne un ensemble de mécanismes destinés à protéger le trafic au niveau d'IP (IPv4 ou IPv6). Les services de sécurité offerts sont l'intégrité en mode non connecté, l'authentification de l'origine des données, la protection contre le rejeu et la confidentialité (confidentialité des données et protection partielle contre l'analyse du trafic). Ces services sont fournis au niveau de la couche IP, offrant donc une protection pour IP et tous les protocoles de niveau supérieur. Optionnel dans IPv4, IPsec est obligatoire pour toute implémentation de IPv6. Une fois IPv6 en place, il sera ainsi possible à tout utilisateur désirant des fonctions de sécurité d'avoir recours à IPsec.

IPsec est développé par un groupe de travail du même nom à l'IETF (Internet Engineering Task Force), groupe qui existe depuis 1992. Une première version des mécanismes proposés a été publiée sous forme de RFC en 1995, sans la partie gestion des clefs, qui est elle plus récente. mais IPsec reste une norme non figée qui fait en ce moment même l'objet de multiples Internet drafts, dont les principaux viennent de passer sous forme de RFC. Nous allons ici présenter les principes du fonctionnement de IPsec, les différents éléments qui le composent, la façon dont ils interagissent et les différentes utilisations possibles.

3.2 Architecture d'IPsec

Pour sécuriser les échanges ayant lieu sur un réseau TCP/IP, il existe plusieurs approches, en particulier en ce qui concerne le niveau auquel est effectuée la sécurisation : niveau applicatif (mails chiffrés par exemple), niveau transport (TLS/SSL, SSH,), ou à l'opposé niveau physique (boîtiers chiffrant toutes les données transitant par un lien donné). IPsec, quant à lui, vise à sécuriser les échanges au niveau de la couche réseau.

3.2.1 Les mécanismes AH et ESP

Pour cela, IPsec fait appel à deux mécanismes de sécurité pour le trafic IP, les "protocoles" AH et ESP, qui viennent s'ajouter au traitement IP classique :

[1] Pluto c'est l'ami de Mickey. Ah non ! Pluto, c'est le chien de Mickey ; l'ami de Mickey, c'est Dingo.

l'Authentication Header (AH) est conçu pour assurer l'intégrité et l'authentification des datagrammes IP sans chiffrement des données (i.e. sans confidentialité). Le principe de AH est d'adjoindre au datagramme IP classique un champ supplémentaire permettant à la réception de vérifier l'authenticité des données incluses dans le datagramme.

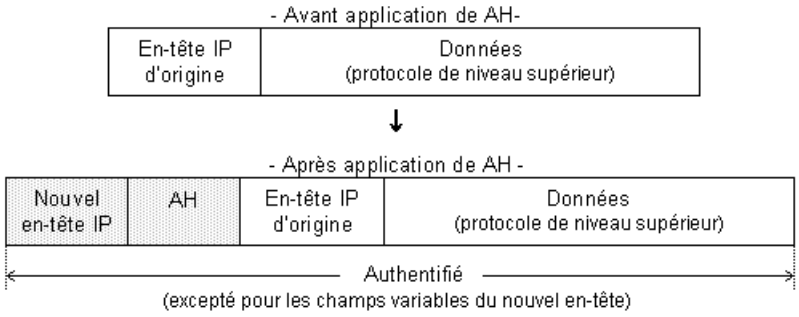

FIG. 2 – Authentification Header

l'Encapsulating Security Payload (ESP) , a pour rôle premier d'assurer la confidentialité, mais peut aussi assurer l'authenticité des données. Le principe de ESP est de générer, à partir d'un datagramme IP classique, un nouveau datagramme dans lequel les données et éventuellement l'en-tête original sont chiffrés.

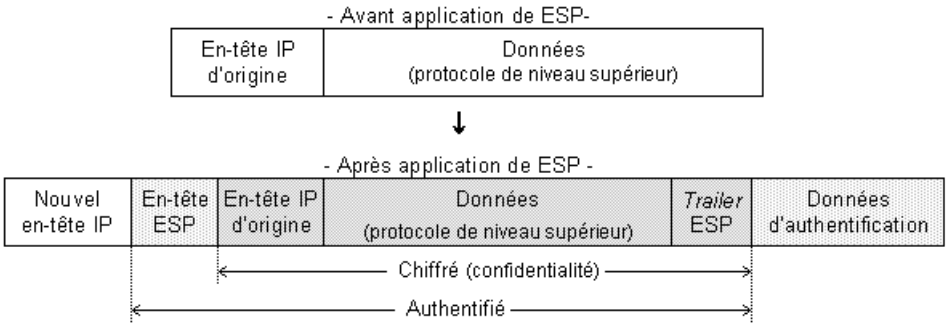

FIG. 3 – Encapsulating Security Payload

Ces mécanismes peuvent être utilisés seuls ou combinés pour obtenir les fonctions de sécurité désirées.

3.2.2 La notion d'association de sécurité

Les mécanismes mentionnés ci-dessus font bien sûr appel à la cryptographie, et utilisent donc un certain nombre de paramètres (algorithmes de chiffrement utilisés, clefs, mécanismes sélectionnés...) sur lesquels les tiers communicants doivent se mettre d'accord. Afin de gérer ces paramètres, IPsec a recours à la notion d'association de sécurité (Security Association, SA).

Une association de sécurité IPsec est une ”connexion” simplexe qui fournit des services de sécurité au trafic qu'elle transporte. On peut aussi la considérer comme une collection de données décrivant l'ensemble des paramètres associés à une communication donnée.

Une SA est unidirectionnelle ; en conséquence, protéger les deux sens d'une communication classique requiert deux associations, une dans chaque sens. Les services de sécurité sont fournis par l'utilisation soit de AH soit de ESP. Si AH et ESP sont tous deux appliqués au trafic en question, deux SA (voire plus) sont créées ; on parle alors de paquet (bundle) de SA. De fait, chaque association est identifiée de manière unique à l'aide d'un triplet composé de :

- L'adresse de destination des paquets.
- L'identifiant d'un protocole de sécurité (AH ou ESP).
- Un index des paramètres de sécurité (Security Parameter Index, SPI). Un SPI est un bloc de 32 bits inscrit en clair dans l'en-tête de chaque paquet échangé ; il est choisi par le récepteur.

Pour gérer les associations de sécurité actives, on utilise une ”base de données des associations de sécurité” (Security Association Database, SAD). Elle contient tous les paramètres relatifs à chaque SA et sera consultée pour savoir comment traiter chaque paquet reçu ou à émettre.

3.2.3 La gestion des clefs et des associations de sécurité

Comme nous l'avons mentionné au paragraphe précédent, les SA contiennent tous les paramètres nécessaires à IPsec, notamment les clefs utilisées. La gestion des clefs pour IPsec n'est liée aux autres mécanismes de sécurité de IPsec que par le biais des SA. Une SA peut être configurée manuellement dans le cas d'une situation simple, mais la règle générale est d'utiliser un protocole spécifique qui permet la négociation dynamique des SA et notamment l'échange des clefs de session.

D'autre part, IPv6 n'est pas destiné à supporter une gestion des clefs ”en bande”, c'est-à-dire où les données relatives à la gestion des clefs seraient transportées à l'aide d'un en-tête IPv6 distinct. Au lieu de cela on utilise un système de gestion des clefs dit ”hors bande”, où les données relatives à la gestion des clefs sont transportées par un protocole de couche supérieure tel que UDP ou TCP. Ceci permet le découplage clair du mécanisme de gestion des clefs et des autres mécanismes de sécurité. Il est ainsi possible de substituer une méthode de gestion des clefs à une autre sans avoir à modifier les implémentations des autres mécanismes de sécurité.

Le protocole de négociation des SA développé pour IPsec s'appelle ”protocole de gestion des clefs et des associations de sécurité pour Internet” (Internet Security Association and Key Management Protocol, ISAKMP). ISAKMP est en fait inutilisable seul : c'est un cadre générique qui permet l'utilisation de plusieurs protocoles d'échange de clef et qui peut être utilisé pour d'autres mécanismes de sécurité que ceux de IPsec. Dans le cadre de la standardisation de IPsec, ISAKMP est associé à une partie des protocoles SKEME et Oakley pour donner un protocole final du nom d'IKE (Internet Key Exchange).

3.2.4 Politique de sécurité

Les protections offertes par IPsec s'appuyent sur des choix définis dans une "base de données de politiques de sécurité" (Security Policy Database, SPD). Cette base de données est établie et maintenue par un utilisateur, un administrateur système ou une application mise en place par ceux-ci. Elle permet de décider, pour chaque paquet, s'il se verra apporter des services de sécurité, s'il sera autorisé à passer outre ou sera rejeté.

La SPD contient une liste ordonnée de règles, chaque règle comportant un certain nombre de critères qui permettent de déterminer quelle partie du trafic est concernée. Les critères utilisables sont l'ensemble des informations disponibles par le biais des en-têtes des couches IP et transport. Ils permettent de définir la granularité selon laquelle les services de sécurité sont applicables et influencent directement le nombre de SA correspondante. Dans le cas où le trafic correspondant à une règle doit se voir attribuer des services de sécurité, la règle indique les caractéristiques de la SA (ou paquet de SA) correspondante : protocole(s), modes, algorithmes requis...

3.3 Principe de fonctionnement

On distingue deux situations :

Trafic sortant Lorsque la "couche" IPsec reçoit des données à envoyer, elle commence par consulter la base de donnée des politiques de sécurité (SPD) pour savoir comment traiter ces données. Si cette base lui indique que le trafic doit se voir appliquer des mécanismes de sécurité, elle récupère les caractéristiques requises pour la SA correspondante et va consulter la base des SA (SAD). Si la SA nécessaire existe déjà, elle est utilisée pour traiter le trafic en question. Dans le cas contraire, IPsec fait appel à IKE pour établir une nouvelle SA avec les caractéristiques requises.

Trafic entrant Lorsque la couche IPsec reçoit un paquet en provenance du réseau, elle examine l'en-tête pour savoir si ce paquet s'est vu appliquer un ou plusieurs services IPsec et si oui quelles sont les références de la SA. Elle consulte alors la SAD pour connaître les paramètres à utiliser pour la vérification et/ou le déchiffrement du paquet. Une fois le paquet vérifié et/ou déchiffré, la SPD est consultée pour savoir si l'association de sécurité appliquée au paquet correspondait bien à celle requise par les politiques de sécurité.

Dans le cas où le paquet reçu est un paquet IP classique, la SPD permet de savoir s'il a néanmoins le droit de passer. Par exemple, les paquets IKE sont une exception. Ils sont traités par IKE, qui peut envoyer des alertes administratives en cas de tentative de connexion infructueuse.

Le schéma ci-dessous représente tous les éléments présentés ci-dessus (en gris), leurs positions et leurs interactions.

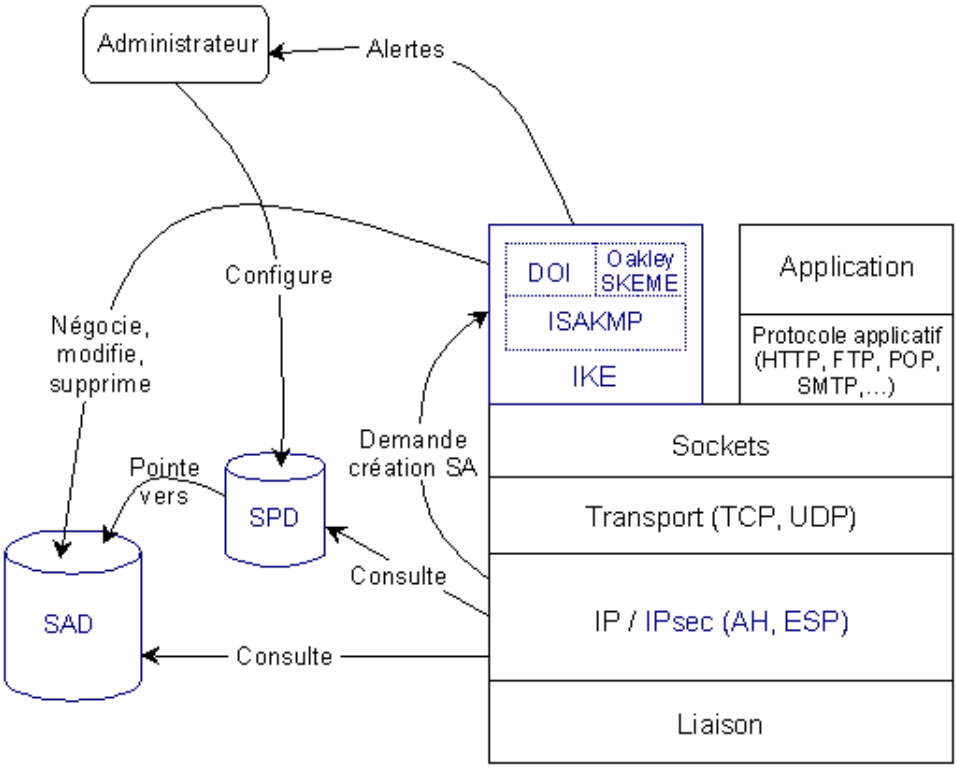

FIG. 4 – Résumé du fonctionnement d'IPSec

4 IPSec en pratique : FreeS/WAN

4.1 Installation

FreeS/WAN est une implémentation libre du protocole IPSec. Cette implémentation se compse de 3 parties distinctes :

- Le code spécifique inséré dans le kernel Linux permettant la manipulation des paquets IPSec (gestion d'en-tête AH et ESP).
- Le démon permettant la négociation de connexion.
- Un ensemble de scripts facilitant l'utilisation de l'application par l'administrateur.

Pour installer cette implémentation d'IPSec, vous devrez donc patcher votre kernel et compiler le démon.
Afin de pouvoir compiler l'ensemble, vous aurez besoin de la bibliothèque arithmétique Gnu MP, disponible sur le site du projet GNU. Dernier point concernant la compilation, aucune confirmation n'est demandée avant que le kernel soit patché ; il est donc très fortement recommandé de faire une sauvegarde de vos sources avant de démarrer l'installation.
Il ne vous reste plus qu'à lancer la procédure avec au choix :

- *make menugo*, pour une configuration intra-kernel avec *menuconfig*
- *make xgo* pour *xconfig*
- *make ogo* pour *config*
- *make omod* pour un module *ipsec* avec *config*

– *make menumod* pour un module avec *menuconfig*
– *make xmod* pour un module *ipsec* avec *xconfig*
Conseil : Préférez l'installation en module qui est beaucoup plus souple.

Si tout se passe bien, vous devez obtenir sur votre système :
– un module kernel *ipsec.o*
– un binaire */usr/local/sbin/ipsec*
– les autres exécutables dans */usr/local/lib/ipsec*
– un script *ipsec* dans */etc/rc.d/init.d*
L'installation achevée, nous pouvons passer à la configuration de l'application.

4.2 Configuration de FreeS/WAN

Pour notre premier exemple de configuration, nous utiliserons la configuration la plus simple possible. Il s'agira de mettre en place une liaison IPSec entre 2 machines d'un LAN.
Pour l'essai nous avons donc notre machine *tsunami* en 2.4.16 ayant l'adresse 172.16.1.23 et un portable IBM nommé *pokemon* en 172.16.1.21.
La première étape consiste a créer un fichier de clefs de signature communes aux deux machines. Nous verrons plus loin comment utiliser RSA, pour le moment, nous créons ce fichier à la main. Ce fichier appelé */etc/ipsec.secrets* est constitué d'une simple ligne :

```
172.16.1.23 172.16.1.21 : PSK
"jxTR11nmSjuJ33n4W51uW3kTR551uUmSmnlRUuWnkjRj3UuTV4T3USSu23Uk55nWu5TkTU
```

Cette ligne est composée de plusieurs parties :
– les hôtes partageant cette clef séparés par un espace
– un double point séparateur
– le type de clef de signature (ici PSK : *Pre Shared Key*)
– la clef entre guillemets
Nous indiquons ici que les hôtes 172.16.1.23 et 172.16.1.21 utiliserons une clef pré-partagée. Cette clef servira a authentifié les hôtes pour l'échange des clefs Diffie-Hellman.
Notes : Les différentes parties de cette ligne doivent IMPERATIVEMENT être séparées par un espace. Le fichier */etc/ipsec.secrets* contient des informations très sensibles, il est donc impératif que seul le *root* puisse lire et écrire dans ce fichier, l'ensemble des autres utilisateurs n'ayant aucun droit.
Il conviendra de placer ce fichier sur les deux machines d'une manière sûre via un réseau protégé ou plus simplement à l'aide d'une disquette.

Le second fichier de configuration est */etc/ipsec.conf*. Celui-ci peut être très simple, comme celui qui va suivre, ou bien plus complexe en fonction de l'utilisation d'IPSec. Voici tout d'abord le fichier de *tsunami* :

```
config setup
       interfaces="ipsec0=eth0"
       plutoload=%search
conn poke-tsu
       left=172.16.1.21
```

```
        right=172.16.1.23
        auto=add
```

Dans la section *config setup*, nous spécifions les interfaces à associer (*ipsec0* est *eth0*) et nous demandons à *pluto* de charger automatiquement les profils existant. Attention, le nom du profil est totalement arbitraire ; nous aurions tout aussi bien pu l'appeler *douglas* ou *sharon*. Mais il est préférable de respecter une certaine convention de nommage, ainsi, le profil *poke-tsu* se réfère à la liaison établie entre *pokemon* et *tsunami*.
Le profil simpliste que nous utilisons définit tout simplement les deux parties en présence, a savoir nous (*tsunami*) et le portable (*pokemon*), sous la forme de 2 côtés. Sur *tsunami*, la partie gauche sera le portable et la partie droite lui-même.
Et voilà, notre configuration est terminée. Mais il faut encore copier le fichier sur *pokemon* sans oublier d'inverser les côtés.
Nous pouvons enfin lancer le démon *ipsec*. Pour cela, il nous suffit d'utiliser la commande du même nom sous la forme :

```
#ipsec setup start
ipsec_setup: Starting FreeS/WAN IPSec 1.94...
```

Le démon est à présent en mémoire et l'interface *ipsec0* prête à servir. Un simple *ifconfig* nous en apportera confirmation :

```
eth0      Link encap:Ethernet  HWaddr 00:48:54:68:05:72
          inet addr:172.16.1.23  Bcast:172.16.255.255  Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2681 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4052 errors:0 dropped:0 overruns:0 carrier:0
          collisions:3 txqueuelen:100
          RX bytes:198132 (193.4 Kb)  TX bytes:5534439 (5.2 Mb)
          Interrupt:11 Base address:0xdf00

ipsec0    Link encap:Ethernet  HWaddr 00:48:54:68:05:72
          inet addr:172.16.1.23  Mask:255.255.0.0
          UP RUNNING NOARP  MTU:16260  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:10
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
```

Comme vous le voyez, l'interface *ipsec0* reprend automatiquement les informations de l'interface *eth0* déjà présente. Ceci est parfaitement normal puisqu'il s'agit en quelque sorte d'un alias ou d'une association, comme le montre la commande :

```
# cat /proc/net/ipsec_tncfg
ipsec0 -> eth0 mtu=16260(1500) -> 1500
ipsec1 -> NULL mtu=0(0) -> 0
ipsec2 -> NULL mtu=0(0) -> 0
ipsec3 -> NULL mtu=0(0) -> 0
```

Nous procédons de même sur l'autre machine. Il ne nous reste plus qu'à établir la connexion IPSec entre les machines :

```
# ipsec auto --up poke-tsu
104 ‘‘poke-tsu’’ #1: STATE_MAIN_I1: initiate
106 ‘‘poke-tsu’’ #1: STATE_MAIN_I2: from STATE_MAIN_I1; sent MI2,
expecting MR2
108 ‘‘poke-tsu’’ #1: STATE_MAIN_I3: from STATE_MAIN_I2; sent MI3,
expecting MR3
004 ‘‘poke-tsu’’ #1: STATE_MAIN_I4: ISAKMP SA established
112 ‘‘poke-tsu’’ #2: STATE_QUICK_I1: initiate
004 ‘‘poke-tsu’’ #2: STATE_QUICK_I2: sent QI2, IPsec SA established
```

L’échange des clefs Diffie-Hellman est terminé et la liaison IPSec est établie. Nous pouvons ensuite obtenir toutes les informations intéressantes avec la commande *ipsec look* :

```
tsunami.eijo.net Wed Jan 16 03:03:41 CET 2002
172.16.1.23/32     -> 172.16.1.21/32     => tun0x1002@172.16.1.21
esp0x9c94c300@172.16.1.21  (0)
ipsec0->eth0 mtu=16260(1500)->1500

esp0x4dfc59b2@172.16.1.23 ESP_3DES_HMAC_MD5: dir=in  src=172.16.1.21
iv_bits=64bits iv=0x7a3f23a8d47f6b30 ooowin=64 alen=128 aklen=128
eklen=192 life(c,s,h)=addtime(602,0,0)

esp0x9c94c300@172.16.1.21 ESP_3DES_HMAC_MD5: dir=out src=172.16.1.23
iv_bits=64bits iv=0xaddeb0f42ab9b1de ooowin=64 alen=128 aklen=128
eklen=192 life(c,s,h)=addtime(602,0,0)

tun0x1001@172.16.1.23 IPIP: dir=in  src=172.16.1.21
life(c,s,h)=addtime(602,0,0)

tun0x1002@172.16.1.21 IPIP: dir=out src=172.16.1.23
life(c,s,h)=addtime(602,0,0)

Destination     Gateway         Genmask         Flags   MSS Window
irtt Iface
0.0.0.0         172.16.1.254    0.0.0.0         UG       40 00 eth0
172.16.0.0      0.0.0.0         255.255.0.0     U        40 00 eth0
172.16.0.0      0.0.0.0         255.255.0.0     U        40 00 ipsec0
172.16.1.21     172.16.1.21     255.255.255.255 UGH      40 00 ipsec0
```

4.3 Utilisation de l’algorithme RSA

Pour l’heure, notre configuration très basique repose sur un secret partagé que nous avons placé sur les 2 machines via un canal de communication sûr. Malheureusement, il arrive que ce genre de manipulation ne soit pas possible (hôte distant de milliers de kilomètres). Dans ce cas, le partage de la clef n’est pas possible et nous devons utiliser un autre algorithme : RSA.
IPSec est capable d’utiliser l’algorithme RSA pour procéder à l’authentification des différentes parties en présence. Une clef RSA est composé de 2 éléments. Une partie privée, la clef

secrète et une partie publique, la clef publique. La première doit être l'objet de la plus paranoiaque des attentions. En revanche, la clef publique peut et doit être connue de tous. Un point reste très important : l'authentification des clefs publiques que vous recevez. En effet, si une personne malintentionnée récupère une telle clef, elle ne pourra rien faire (a moins d'être le mathématicien qui mettra en péril tous les algorithmes à clef publique en faisant une découverte fondamentale sur la factorisation d'une valeur en ses nombres premiers). Par contre, une telle personne peut parfaitement vous fournir une fausse clef publique et ainsi vous tromper. Il est impératif de vous assurez de la provenance des clefs publiques que vous recevez !
Commençons par générer une première paire de clefs à l'aide de :

```
# ipsec rsasigkey --verbose 2048 > clef
getting 128 random bytes from /dev/random...
looking for a prime starting there (can take a while)...
found it after 79 tries.
getting 128 random bytes from /dev/random...
looking for a prime starting there (can take a while)...
found it after 58 tries.
swapping primes so p is the larger...
computing modulus...
computing lcm(p-1, q-1)...
computing d...
computing exp1, exp1, coeff...
output...
```

Vous obtenez alors le fichier *clef* dans le répertoire courant. Il vous suffira d'en copier le contenu dans votre */etc/ipsec.secrets* pour obtenir ceci :

```
: RSA   {
        # RSA 2048 bits   tsunami.eijo.net   Tue Jan 15 23:42:39 2002
        # for signatures only, UNSAFE FOR ENCRYPTION
        #pubkey=0sAQOhxXeVLQiip/tbtK7kvmVT70HmehC31tTPP1...
        #IN KEY 0x4200 4 1 AQOhxXeVLQiip/tbtK7kvmVT70Hme...
        # (0x4200 = auth-only host-level, 4 = IPSec, 1 = RSA)
        Modulus: 0xa1c577952d08a2a7fb5bb4aee4be6553ef41e...
        PublicExponent: 0x03
        # everything after this point is secret
        PrivateExponent: 0x1af63e98dcd6c5c6a9e49e1d261fbb...
        Prime1: 0xe0bbf904355852aa09b9e7a50d88eb910fe0ec0...
        Prime2: 0xb84709a5b819c04ea26cbda3f5b7459bd6f7669...
        Exponent1: 0x95d2a602ce3ae1c6b1269a6e0905f260b540...
        Exponent2: 0x7ada066e7abbd589c19dd3c2a3cf83bd39fa...
        Coefficient: 0xcbfb5626e0ff009250450194f820ca9f0a...
       }
```

Attention, une fois encore, il est nécessaire de respecter une syntaxe bien définie pour cette insertion :

- le double point (:) doit être non indenté et donc à la marge
- toutes les lignes entre { et }, y compris ces symboles, doivent être indentées

– sur la ligne RSA, des espaces doivent séparer les éléments.
```
:RSA {
ou
:   RSA{
```
ne sont, par exemple, pas valides.

Vous remarquerez que la commande *ipsec rsasigkey* génère automatiquement une sortie utilisable dans *ipsec.conf*. En commentaire, vous trouverez une ligne commençant par *#pubkey=0s* ; il s'agit de la clef publique associée à la clef secrète.
Il ne vous reste plus alors qu'à modifier vos fichiers de configuration ainsi (pour pokemon) :

```
config setup
       interfaces="ipsec0=eth0"
       plutoload=%search

conn %default
       authby=rsasig
conn poke-tsu
       right=172.16.1.21
       rightrsasigkey=0sAQPi/eEj42...
       left=172.16.1.23
       leftrsasigkey=0sAQoLIhGbm24...
       auto=add
```

Plusieurs changements ont été apportés :
– Une section *%default* a été ajoutée. Celle-ci contient des paramètres communs à tous les profils qui suivent. Nous y avons placé une mention permettant de définir une authentification par signature RSA.
– Dans le profil poke-tsu, nous avons ajouté les clefs publiques des 2 machines, *rightrsasigkey* est la clef publique de *pokemon*, c'est-à-dire de la machine ou se trouve cet *ipsec.conf*. *leftrsasigkey* est la clef publique de la machine en face, *tsunami*.

Nous procédons de même pour le fichier */etc/ipsec.conf* de *tsunami*.
Le lancement du démon et de la connexion IPSec reste identique à la procédure décrite ci-dessus.

Nous arrivons au bout de notre essai de l'implémentation FreeS/WAN d'IPSec. Bien sûr, nous n'avons pas été exhaustif dans la description des fonctionnalités de FreeS/WAN. Il implémente en effet les 2 modes IPSec, tunnel et transport (sous le nom de Road Warrior), et permet une vaste gamme de configurations possibles. La documentation de FreeS/WAN est relativement bien faite et bourrée d'exemples concrets. N'Hésitez pas à y puiser toutes les informations utiles…

5 Références

Virtual Private Networks : Charlie Scott, Paul Wolfe & Mike Erwin (O'REILLY)

Les réseaux TCP/IP : Douglas E. Comer

IPSec & IETF : http ://www.ietf.org/html.charters/ipsec-charter.html

IPSec et toutes les normes en français : http ://www.eisti.fr/doc/norm/norm.dim

FreeS/WAN : http ://www.freeswan.org

OpenBSD : http ://www.openbsd.org

CIPE : http ://sites.inka.de/ bigred/devel/cipe.html

Le projet Kame : http ://www.kame.net

6 Remerciements

A Linux Magazine France pour leur article sur FreeS/WAN paru dans le numéro de Septembre 2001.

7 GNU Free Documentation License

Version 1.1, March 2000

Preamble

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom : to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation : a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals ; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

7.1 Applicability and Definitions

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The “Cover Texts” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A “Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not “Transparent” is called “Opaque”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The “Title Page” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

7.2 Verbatim Copying

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

7.3 Copying in Quantity

If you publish printed copies of the Document numbering more than 100, and the Document’s license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts : Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material

on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

7.4 Modifications

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version :

- Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).
- State on the Title page the name of the publisher of the Modified Version, as the publisher.
- Preserve all the copyright notices of the Document.
- Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- Include an unaltered copy of this License.

- Preserve the section entitled “History”, and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled “History” in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the “History” section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- In any section entitled “Acknowledgements” or “Dedications”, preserve the section’s title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- Delete any section entitled “Endorsements”. Such a section may not be included in the Modified Version.
- Do not retitle any existing section as “Endorsements” or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version’s license notice. These titles must be distinct from any other section titles.

You may add a section entitled “Endorsements”, provided it contains nothing but endorsements of your Modified Version by various parties – for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another ; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

7.5 Combining Documents

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you in-

clude in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled “History” in the various original documents, forming one section entitled “History” ; likewise combine any sections entitled “Acknowledgements”, and any sections entitled “Dedications”. You must delete all sections entitled “Endorsements.”

7.6 Collections of Documents

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7.7 Aggregation With Independent Works

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an “aggregate”, and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document’s Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

7.8 Translation

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the

original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

7.9 Termination

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

7.10 Future Revisions of This License

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http ://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License ”or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM : How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page :

> Copyright © YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation ; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled “GNU Free Documentation License”.

If you have no Invariant Sections, write “with no Invariant Sections” instead of saying which ones are invariant. If you have no Front-Cover Texts, write “no Front-Cover Texts” instead of “Front-Cover Texts being LIST” ; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.