Sécurisation du réseau dans Oracle® Solaris 11.2

Référence: E53809-02
Septembre 2014

Table des matières

Liste des tableaux

Liste des exemples

Utilisation de la présente documentation

- **Présentation** : décrit la manière de fournir la sécurité du réseau. Inclut la sécurité des liens, les paramètres de réseau analysables, la protection par pare-feu, les protocoles IPsec et IKE et la protection du noyau SSL pour les serveurs Web.
- **Public visé** : les administrateurs de sécurité.
- **Connaissances requises** : exigences en matière de sécurité du site.

Bibliothèque de documentation du produit

Les informations de dernière minute et les problèmes connus pour ce produit sont inclus dans la bibliothèque de documentation accessible à l'adresse : http://www.oracle.com/pls/topic/lookup?ctx=E56338.

Accès aux services de support Oracle

Les clients Oracle ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info ou le site http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs si vous êtes malentendant.

Commentaires

Faites part de vos commentaires sur cette documentation à la page suivante : http://www.oracle.com/goto/docfeedback.

♦ ♦ ♦ CHAPITRE 1

Utilisation de la protection des liens dans des environnements virtualisés

Ce chapitre décrit la protection des liens et leur configuration sur un système Oracle Solaris. Ce chapitre comprend les sections suivantes :

Nouveautés concernant la sécurité du réseau dans Oracle Solaris 11.2

Cette section met en évidence des informations pour les clients existants à propos d'importantes nouveautés concernant la sécurité du réseau dans cette version.

IKE Version 2 (IKEv2) permet la gestion automatique des clés pour IPsec l'aide de la dernière version du protocole IKE. IKEv2 et IPsec utilisent des algorithmes de la fonctionnalité de structure cryptographique d'Oracle Solaris.

Remarque - La fonctionnalité de structure cryptographique de Oracle Solaris est validée pour FIPS 140-2, Niveau 1. Afin d'établir le mode d'utilisation pour IKE, reportez-vous au Tableau 8-1, “Implémentation d'IKEv2 et d'IKEv1 dans Oracle Solaris”. Pour plus de détails sur le matériel et le logiciel, reportez-vous à Oracle FIPS 140 Software Validations (http://www.oracle.com/technetwork/topics/security/fips140-software-validations-1703049.html).

La prise en charge d'IKE Version 1 (IKEv1) est toujours disponible. Pour plus d'informations, reportez-vous au Chapitre 8, A propos d'Internet Key Exchange.

A propos de la protection des liens

Avec la généralisation de la virtualisation dans les configurations système, un administrateur hôte peut accorder à des machines virtuelles (VM) invitées un accès exclusif à un lien physique ou virtuel. Cette configuration améliore les performances du réseau en permettant au trafic du réseau de l'environnement virtuel d'être isolé du trafic général envoyé ou reçu par le système hôte. Dans le même temps, cette configuration peut exposer le système et l'ensemble du réseau à des paquets nuisibles susceptibles d'être générés par un environnement invité.

La protection des liens vise à prévenir les dommages pouvant être causés par des machines virtuelles potentiellement dangereuses invitées sur le réseau. Cette fonction offre une protection contre les menaces de base suivantes :

- Usurpation IP, DHCP et MAC
- Usurpation de trame de niveau 2 telle que les attaques BPDU (Bridge Protocol Data Unit)

Remarque - La protection des liens ne remplace pas le déploiement d'un pare-feu, en particulier pour des configurations présentant des besoins de filtrage complexes.

Types de protection des liens

Le mécanisme de protection des liens dans Oracle Solaris fournit les types de protection suivants :

`mac-nospoof`

Active la protection contre l'usurpation de l'adresse MAC du système. Si le lien appartient à une zone, l'activation de `mac-nospoof` empêche le propriétaire de la zone de modifier l'adresse MAC du lien.

`ip-nospoof`

Active la protection contre l'usurpation d'IP. Par défaut, les paquets sortants avec des adresses DHCP et des adresses IPv6 locales de lien sont autorisés.

Vous pouvez ajouter des adresses à l'aide de la propriété de lien `allowed-ips`. Pour les adresses IP, l'adresse source du paquet doit correspondre à une adresse de la liste `allowed-ips`. Pour un paquet ARP, l'adresse du protocole de l'expéditeur du paquet doit se trouver dans la liste `allowed-ips`.

`dhcp-nospoof`

Active la protection contre l'usurpation du client DHCP. Par défaut, les paquets DHCP dont l'ID correspond à l'adresse MAC du système sont autorisés.

Vous pouvez ajouter des clients autorisés à l'aide de la propriété de lien `allowed-dhcp-cids`. Les entrées de la liste `allowed-dhcp-cids` doivent être formatées comme indiqué dans la page de manuel `dhcpagent`(1M).

`restricted`

Limite les paquets sortants aux paquets IPv4, IPv6 et ARP. Ce type de protection a été conçu pour empêcher le lien de générer des trames de contrôle de niveau 2 potentiellement dangereuses.

Remarque - Les paquets ignorés en raison de la protection des liens font l'objet d'un suivi par les statistiques du noyau pour les quatre types de protection : `mac_spoofed`, `dhcp_spoofed`, `ip_spoofed` et `restricted`. Pour consulter ces statistiques par lien, reportez-vous à la section “Affichage de la configuration et des statistiques de protection des liens” à la page 21.

Pour plus d'informations sur ces types de protection, reportez-vous à la page de manuel `dladm`(1M).

Configuration de la protection des liens

Pour utiliser la protection des liens, définissez la propriété `protection` du lien. Si le type de protection fonctionne avec d'autres fichiers de configuration, tels que `ip-nospoof` avec `allowed-ips` ou `dhcp-nospoof` avec `allowed-dhcp-cids`, vous devez effectuer deux opérations globales. Vous devez tout d'abord activer la protection des liens. Puis vous devez personnaliser le fichier de configuration pour identifier les autres paquets dont la transmission est autorisée.

Remarque - Vous devez configurer la protection des liens dans la zone globale.

La liste des tâches suivante indique les procédures de configuration de la protection des liens sur un système Oracle Solaris.

TABLEAU 1-1 Liste de tâches pour la protection des liens

Tâche	Description	Pour obtenir des instructions
Activation de la protection des liens	Limite les paquets envoyés à partir d'un lien et protège les liens contre l'usurpation.	“Activation de la protection des liens” à la page 18
Désactivation de la protection des liens	Supprime les protections des liens.	“Désactivation de la protection des liens” à la page 19
Spécification du type de protection des liens IP	Indique les adresses IP autorisées à traverser le mécanisme de protection des liens.	“Spécification des adresses IP pour la protection contre l'usurpation d'adresses IP” à la page 19

Tâche	Description	Pour obtenir des instructions
Spécification du type de protection des liens DHCP	Indique les adresses DHCP autorisées à traverser le mécanisme de protection des liens.	“Spécification des clients DHCP pour assurer une protection contre l'usurpation d'adresses DHCP” à la page 20
Affichage de la configuration de la protection des liens	Répertorie les liens protégés et les exceptions, et présente les statistiques de mise en oeuvre.	“Affichage de la configuration et des statistiques de protection des liens” à la page 21

▼ Activation de la protection des liens

Cette procédure restreint les types de paquets sortants et empêche l'usurpation des liens.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Link Security (sécurité des liens du réseau). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

1. **Affichez les types de protection des liens disponibles.**

```
# dladm show-linkprop -p protection
LINK     PROPERTY        PERM VALUE        EFFECTIVE    DEFAULT   POSSIBLE
net0     protection      rw   --           --           --        mac-nospoof,
                                                                   restricted,
                                                                   ip-nospoof,
                                                                   dhcp-nospoof
```

Pour obtenir une description des types possibles, reportez-vous à la section “Types de protection des liens” à la page 16 et à la page de manuel `dladm`(1M).

2. **Activez la protection des liens en spécifiant un ou plusieurs types de protection.**

```
# dladm set-linkprop -p protection=value[,value,...] link
```

Dans l'exemple suivant, les quatre types de protection des liens sont activés sur le lien `vnic0` :

```
# dladm set-linkprop \
-p protection=mac-nospoof,restricted,ip-nospoof,dhcp-nospoof vnic0
```

Attention - Testez toutes les valeurs de protection une par une avant de les activer. Une mauvaise configuration système peut empêcher la communication.

3. **Vérifiez que les protections des liens sont activées.**

```
# dladm show-linkprop -p protection vnic0
LINK     PROPERTY        PERM VALUE        EFFECTIVE    DEFAULT   POSSIBLE
net0     protection      rw   mac-nospoof  mac-nospoof  --        mac-nospoof,
                              restricted   restricted   --        restricted,
                              ip-nospoof   ip-nospoof   --        ip-nospoof,
```

```
                                dhcp-nospoof dhcp-nospoof --        dhcp-nospoof
```

Désactivation de la protection des liens

Cette procédure permet de réinitialiser la protection des liens sur la valeur par défaut, pas de protection des liens.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Link Security (sécurité des liens du réseau). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Désactivez la protection des liens en réinitialisant la propriété `protection` sur sa valeur par défaut.**

   ```
   # dladm reset-linkprop -p protection link
   ```

2. **Vérifiez que les protections de liens sont désactivées.**

   ```
   # dladm show-linkprop -p protection vnic0
   LINK     PROPERTY        PERM VALUE        EFFECTIVE    DEFAULT   POSSIBLE
   net0     protection      rw   --           --           --        mac-nospoof,
                                                                     restricted,
                                                                     ip-nospoof,
                                                                     dhcp-nospoof
   ```

Spécification des adresses IP pour la protection contre l'usurpation d'adresses IP

Avant de commencer Le type de protection `ip-nospoof` est activé, comme indiqué dans la section "Activation de la protection des liens" à la page 18.

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Link Security (sécurité des liens du réseau). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Vérifiez que vous avez activé la protection contre l'usurpation d'adresses IP.**

   ```
   # dladm show-linkprop -p protection link
   LINK     PROPERTY        PERM VALUE        EFFECTIVE    DEFAULT   POSSIBLE
   link     protection      rw   ip-nospoof   ip-nospoof   --        mac-nospoof,
                                                                     restricted,
                                                                     ip-nospoof,
   ```

```
                                                                dhcp-nospoof
```

2. **Ajoutez des adresses IP à la liste des valeurs par défaut pour la propriété de lien `allowed-ips`.**

   ```
   # dladm set-linkprop -p allowed-ips=IP-addr[,IP-addr,...] link
   ```

   L'exemple suivant illustre l'ajout des adresses IP `10.0.0.1` et `10.0.0.2` à la propriété `allowed-ips` du lien `vnic0` :

   ```
   # dladm set-linkprop -p allowed-ips=10.0.0.1,10.0.0.2 vnic0
   ```

   Pour plus d'informations, reportez-vous à la page de manuel `dladm`(1M).

▼ Spécification des clients DHCP pour assurer une protection contre l'usurpation d'adresses DHCP

Avant de commencer

Le type de protection `dhcp-nospoof` est activé, comme indiqué dans la section “Activation de la protection des liens” à la page 18.

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Link Security (sécurité des liens du réseau). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

1. **Vérifiez que vous avez activé la protection contre l'usurpation d'adresses DHCP.**

   ```
   # dladm show-linkprop -p protection link
   LINK     PROPERTY        PERM VALUE        EFFECTIVE    DEFAULT   POSSIBLE
   link     protection      rw   dhcp-nospoof dhcp-nospoof --        mac-nospoof,
                                                                     restricted,
                                                                     ip-nospoof,
                                                                     dhcp-nospoof
   ```

2. **Indiquez une phrase ASCII pour la propriété de lien `allowed-dhcp-cids`.**

   ```
   # dladm set-linkprop -p allowed-dhcp-cids=CID-or-DUID[,CID-or-DUID,...] link
   ```

   L'exemple suivant indique comment spécifier la chaîne `hello` en tant que valeur de la propriété `allowed-dhcp-cids` du lien `vnic0` :

   ```
   # dladm set-linkprop -p allowed-dhcp-cids=hello vnic0
   ```

   Pour plus d'informations, reportez-vous à la page de manuel `dladm`(1M).

▼ Affichage de la configuration et des statistiques de protection des liens

Avant de commencer

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Link Security (sécurité des liens du réseau). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. Affichez les valeurs des propriétés de protection des liens.

```
# dladm show-linkprop -p protection,allowed-ips,allowed-dhcp-cids link
```

L'exemple suivant affiche les valeurs des propriétés `protection`, `allowed-ips` et `allowed-dhcp-cids` du lien `vnic0` :

```
# dladm show-linkprop -p protection,allowed-ips,allowed-dhcp-cids vnic0
LINK    PROPERTY            PERM  VALUE         EFFECTIVE   DEFAULT POSSIBLE
vnic0   protection          rw    mac-nospoof   mac-nospoof --      mac-nospoof,
                                  restricted    restricted          restricted,
                                  ip-nospoof    ip-nospoof          ip-nospoof,
                                  dhcp-nospoof  dhcp-nospoof        dhcp-nospoof
vnic0   allowed-ips         rw    10.0.0.1,     10.0.0.1,   --      --
                                  10.0.0.2      10.0.0.2
vnic0   allowed-dhcp-cids   rw    hello         hello       --      --
```

Remarque - La propriété `allowed-ips` n'est utilisée que si `ip-nospoof` est activé, comme indiqué sous `EFFECTIVE`. La propriété `allowed-dhcp-cids` n'est utilisée que si `dhcp-nospoof` est activé.

2. Affichez les statistiques de protection des liens.

La sortie de la commande `dlstat` est validée, cette commande est donc appropriée pour les scripts.

```
# dlstat -A
...
 vnic0
  mac_misc_stat
               multircv                    0
              brdcstrcv                    0
               multixmt                    0
              brdcstxmt                    0
           multircvbytes                   0
            bcstrcvbytes                   0
           multixmtbytes                   0
            bcstxmtbytes                   0
                txerrors                   0
             macspoofed                    0  <----------
              ipspoofed                    0  <----------
            dhcpspoofed                    0  <----------
```

```
            restricted                      0  <----------
              ipackets                      3
                rbytes                    182
...
```

La sortie indique qu'il n'y a eu aucune tentative de transmission d'un paquet falsifié ou faisant l'objet d'une restriction.

Vous pouvez utiliser la commande `kstat`, mais la sortie correspondante n'est pas validée. Par exemple, la commande suivante affiche les statistiques `dhcpspoofed` :

```
# kstat vnic0:0:link:dhcpspoofed
module: vnic0                           instance: 0
name:   link                            class:    vnic
        dhcpspoofed                     0
```

Pour plus d'informations, reportez-vous aux pages de manuel `dlstat`(1M) et `kstat`(1M).

♦ ♦ ♦ CHAPITRE 2

Réglage du réseau

Ce chapitre explique le réglage des paramètres réseau qui affectent la sécurité dans Oracle Solaris.

Réglage du réseau

TABLEAU 2-1 Liste des tâches pour le réglage du réseau

Tâche	Description	Pour obtenir des instructions
Désactivez le démon de routage réseau.	Limite l'accès aux systèmes par des renifleurs de réseau.	“Désactivation du démon de routage réseau” à la page 24
Prévention de la diffusion d'informations sur la topologie du réseau.	Empêche la diffusion de paquets.	“Désactivation du transfert de paquets de diffusion” à la page 25
	Désactivation des réponses aux demandes d'écho de diffusion et aux demandes d'écho multidiffusion.	“Désactivation des réponses aux demandes d'écho” à la page 25
Pour les systèmes constituant des passerelles vers d'autres domaines, tels qu'un pare-feu ou un noeud de réseau privé virtuel (VPN), activation du multihébergement strict de la source et de la destination.	Empêche les paquets qui ne possèdent pas dans leur en-tête l'adresse de la passerelle de se déplacer au-delà de la passerelle.	“Définition du multihébergement strict” à la page 26
Prévention des attaques DOS en contrôlant le nombre de connexions incomplètes au système.	Limite le nombre maximal de connexions TCP incomplètes pour un processus d'écoute TCP.	“Définition du nombre maximal de connexions TCP incomplètes” à la page 27
Prévention des attaques DOS en contrôlant le nombre de connexions entrantes autorisées.	Spécifie le nombre maximal par défaut de connexions TCP en attente pour un processus d'écoute TCP.	“Définition du nombre maximal de connexions TCP en attente” à la page 27
Vérifiez que de nombres aléatoires forts sont générés pour les connexions TCP initiale.	Respecte la valeur de génération de numéro de séquence spécifiée par RFC 6528.	“Spécification d'un numéro aléatoire fort pour la connexion TCP initiale” à la page 28
Prévention de la redirection ICMP.	Supprime les indicateurs de la topologie du réseau.	“Prévention des redirections ICMP” à la page 28
Restauration des valeurs sécurisées par défaut des paramètres réseau.	Renforce la sécurité lorsqu'elle a été réduite par des actions d'administration.	“Réinitialisation des paramètres réseau sur des valeurs sécurisées” à la page 29

Désactivation du démon de routage réseau

Cette procédure permet d'empêcher le routage réseau après l'installation en spécifiant un routeur par défaut. Cette procédure peut aussi être effectuée après une configuration manuelle du routage.

Remarque - De nombreuses procédures de configuration requièrent la désactivation du démon de routage. Vous avez donc peut-être désactivé ce démon dans le cadre d'une procédure de configuration générale.

Avant de commencer

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Management (gestion du réseau). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Assurez-vous que le démon de routage est en cours d'exécution.**

```
$ svcs -x svc:/network/routing/route:default
svc:/network/routing/route:default (in.routed network routing daemon)
 State: online since April 10, 2014 05:15:35 AM PDT
   See: in.routed(1M)
   See: /var/svc/log/network-routing-route:default.log
Impact: None.
```

Si le service n'est pas en cours d'exécution, vous pouvez vous arrêter ici.

2. **Désactivez le démon de routage.**

```
# routeadm -d ipv4-forwarding -d ipv6-forwarding
# routeadm -d ipv4-routing -d ipv6-routing
# routeadm -u
```

3. **Vérifiez que le démon de routage est désactivé.**

```
$ svcs -x routing/route:default
svc:/network/routing/route:default (in.routed network routing daemon)
 State: disabled since April 11, 2014 10:10:10 AM PDT
Reason: Disabled by an administrator.
   See: http://support.oracle.com/msg/SMF-8000-05
   See: in.routed(1M)
Impact: This service is not running.
```

Voir aussi

Page de manuel routeadm(1M)

Désactivation du transfert de paquets de diffusion

Par défaut, Oracle Solaris transmet les paquets de diffusion. Si la stratégie de sécurité de votre site exige une réduction du risque de saturation par diffusions, modifiez la valeur par défaut à l'aide de cette procédure.

Remarque - En désactivant la propriété `_forward_directed_broadcasts`, vous désactivez les commandes ping des diffusions.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Management (gestion du réseau). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Définissez sur 0 la propriété de transfert des paquets de diffusion pour les paquets IP.**

```
# ipadm set-prop -p _forward_directed_broadcasts=0 ip
```

2. **Contrôlez la valeur en cours.**

```
# ipadm show-prop -p _forward_directed_broadcasts ip
PROTO  PROPERTY                   PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
ip     _forward_directed_broadcasts  rw   0         --           0         0,1
```

Voir aussi Page de manuel `ipadm`(1M)

Désactivation des réponses aux demandes d'écho

Cette procédure permet d'empêcher la diffusion d'informations sur la topologie du réseau.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Management (gestion du réseau). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Définissez sur 0 la propriété de réponse aux demandes d'écho de diffusion pour les paquets IP, puis contrôlez la valeur en cours.**

```
# ipadm set-prop -p _respond_to_echo_broadcast=0 ip

# ipadm show-prop -p _respond_to_echo_broadcast ip
PROTO  PROPERTY                 PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
ip     _respond_to_echo_broadcast rw   0         --           1         0,1
```

2. **Définissez sur 0 la propriété de réponse aux demandes d'écho multidiffusion pour les paquets IP, puis contrôlez la valeur en cours.**

```
# ipadm set-prop -p _respond_to_echo_multicast=0 ipv4
# ipadm set-prop -p _respond_to_echo_multicast=0 ipv6

# ipadm show-prop -p _respond_to_echo_multicast ipv4
PROTO  PROPERTY                 PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
ipv4  _respond_to_echo_multicast rw   0         --           1         0,1
# ipadm show-prop -p _respond_to_echo_multicast ipv6
PROTO  PROPERTY                 PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
ipv6  _respond_to_echo_multicast rw   0         --           1         0,1
```

Voir aussi Pour plus d'informations, reportez-vous à la section " _respond_to_echo_broadcast et _respond_to_echo_multicast (ipv4 ou ipv6) " du manuel " Manuel de référence des paramètres réglables d'Oracle Solaris 11.2 " et à la page de manuel `ipadm`(1M).

▼ Définition du multihébergement strict

Pour les systèmes constituant des passerelles vers d'autres domaines, tels qu'un pare-feu ou un noeud de réseau privé virtuel (VPN), cette procédure permet d'activer le multihébergement strict. La propriété `hostmodel` contrôle le comportement d'envoi et de réception de paquets IP sur un système à multihébergement.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Management (gestion du réseau). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Définissez la propriété `hostmodel` sur `strong` pour les paquets IP.**

```
# ipadm set-prop -p hostmodel=strong ipv4
# ipadm set-prop -p hostmodel=strong ipv6
```

2. **Vérifiez la valeur actuelle et notez les valeurs possibles.**

```
# ipadm show-prop -p hostmodel ip
PROTO  PROPERTY    PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
ipv6   hostmodel   rw   strong    strong       weak      strong,src-priority,weak
ipv4   hostmodel   rw   strong    strong       weak      strong,src-priority,weak
```

Voir aussi Pour plus d'informations, reportez-vous à la section " hostmodel (ipv4 ou ipv6) " du manuel " Manuel de référence des paramètres réglables d'Oracle Solaris 11.2 " et à la page de manuel `ipadm`(1M).

Pour plus d'informations sur l'utilisation du multihébergement strict, reportez-vous à la section “Protection de la connexion entre deux réseaux locaux à l'aide d'IPsec en mode Tunnel” à la page 120.

▼ Définition du nombre maximal de connexions TCP incomplètes

Cette procédure permet d'empêcher les attaques par déni de service en contrôlant le nombre de connexions en attente incomplètes.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Management (gestion du réseau). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

1. **Définissez le nombre maximal de connexions entrantes.**

```
# ipadm set-prop -p _conn_req_max_q0=4096 tcp
```

2. **Contrôlez la valeur en cours.**

```
# ipadm show-prop -p _conn_req_max_q0 tcp
PROTO  PROPERTY         PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
tcp    _conn_req_max_q0 rw   4096      --           128       1-4294967295
```

Voir aussi Pour plus d'informations, reportez-vous à la section “ _conn_req_max_q0 ” du manuel “ Manuel de référence des paramètres réglables d’Oracle Solaris 11.2 ” et à la page de manuel `ipadm`(1M).

▼ Définition du nombre maximal de connexions TCP en attente

Cette procédure permet d'empêcher les attaques par déni de service en contrôlant le nombre de connexions entrantes autorisées.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Management (gestion du réseau). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

1. **Définissez le nombre maximal de connexions entrantes.**

```
# ipadm set-prop -p _conn_req_max_q=1024 tcp
```

2. **Contrôlez la valeur en cours.**

```
# ipadm show-prop -p _conn_req_max_q tcp
PROTO  PROPERTY         PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
tcp    _conn_req_max_q  rw   1024      --           128       1-4294967295
```

Voir aussi Pour plus d'informations, reportez-vous à la section " _conn_req_max_q " du manuel " Manuel de référence des paramètres réglables d'Oracle Solaris 11.2 " et à la page de manuel `ipadm`(1M).

▼ Spécification d'un numéro aléatoire fort pour la connexion TCP initiale

Cette procédure garantit que le paramètre de génération du numéro de séquence initial TCP est conforme à RFC 6528 (http://www.ietf.org/rfc/rfc6528.txt).

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant de l'autorisation `solaris.admin.edit/etc.default/inetinit`. Par défaut, le rôle `root` dispose de cette autorisation. Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Vérifiez si la valeur par défaut de la variable `TCP_STRONG_ISS` est bien `2`.**

```
# grep TCP_STRONG /etc/default/inetinit
# TCP_STRONG_ISS sets the TCP initial sequence number generation parameters.
# Set TCP_STRONG_ISS to be:
TCP_STRONG_ISS=2
```

2. **Si la valeur de `TCP_STRONG_ISS` n'est pas `2`, configurez-la sur `2`.**

```
# pfedit /etc/default/inetinit
TCP_STRONG_ISS=2
```

3. **Réinitialisez le système.**

```
# /usr/sbin/reboot
```

▼ Prévention des redirections ICMP

Les routeurs utilisent les messages de redirection ICMP afin d'informer les hôtes de l'existence de routes plus directes vers une destination. Un message de redirection ICMP illicite risque de provoquer une attaque Man-in-the-middle.

Avant de commencer

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Management (gestion du réseau). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Définissez sur 1 la propriété ignorer les réacheminements, puis contrôlez la valeur en cours.**

   Les messages de redirection ICMP modifient la table de routage de l'hôte et ne sont pas authentifiés. En outre, le traitement de paquets redirigés augmente la sollicitation de la CPU des systèmes.

```
# ipadm set-prop -p _ignore_redirect=1 ipv4
# ipadm set-prop -p _ignore_redirect=1 ipv6
# ipadm show-prop -p _ignore_redirect ipv4
PROTO  PROPERTY         PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
ipv4  _ignore_redirect  rw   1         1            0         0,1
# ipadm show-prop -p _ignore_redirect ipv6
PROTO  PROPERTY         PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
ipv6  _ignore_redirect  rw   1         1            0         0,1
```

2. **Empêchez l'envoi de messages de redirection ICMP.**

   Les messages de ce type incluent des informations issues de la table de routage qui peuvent révéler une partie de la topologie du réseau.

```
# ipadm set-prop -p send_redirects=off ipv4
# ipadm set-prop -p send_redirects=off ipv6
# ipadm show-prop -p send_redirects ipv4
PROTO PROPERTY          PERM CURRENT  PERSISTENT   DEFAULT  POSSIBLE
ipv4  send_redirects    rw   off      off          on       on,off

# ipadm show-prop -p send_redirects ipv6
PROTO  PROPERTY        PERM CURRENT   PERSISTENT   DEFAULT  POSSIBLE
ipv6  send_redirects   rw   off       off          on       on,off
```

   Pour plus d'informations, reportez-vous à la section " send_redirects (ipv4 ou ipv6) " du manuel " Manuel de référence des paramètres réglables d'Oracle Solaris 11.2 " et à la page de manuel `ipadm`(1M).

▼ Réinitialisation des paramètres réseau sur des valeurs sécurisées

De nombreux paramètres réseau, sécurisés par défaut, sont réglables et peuvent donc avoir été modifiés. Si les conditions du site le permettent, restaurez les valeurs par défaut des paramètres réglables suivants.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Management (gestion du réseau). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Définissez sur 0 la propriété de transfert des packages source pour les paquets IP, puis contrôlez la valeur en cours.**

   La valeur par défaut empêche les attaques par déni de service provenant de paquets falsifiés.

```
# ipadm set-prop -p _forward_src_routed=0 ipv4
# ipadm set-prop -p _forward_src_routed=0 ipv6
# ipadm show-prop -p _forward_src_routed ipv4
PROTO  PROPERTY             PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
ipv4  _forward_src_routed   rw   0         --           0         0,1
# ipadm show-prop -p _forward_src_routed ipv6
PROTO  PROPERTY             PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
ipv6  _forward_src_routed   rw   0         --           0         0,1
```

   Pour plus d'informations, reportez-vous à la section " forwarding (ipv4 ou ipv6) " du manuel " Manuel de référence des paramètres réglables d'Oracle Solaris 11.2 ".

2. **Définissez sur 0 la propriété de réponse netmask pour les paquets IP, puis contrôlez la valeur en cours.**

   La valeur par défaut empêche la diffusion d'informations sur la topologie du réseau.

```
# ipadm set-prop -p _respond_to_address_mask_broadcast=0 ip
# ipadm show-prop -p _respond_to_address_mask_broadcast ip
PROTO PROPERTY                       PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
ip    _respond_to_address_mask_broadcast rw   0         --           0         0,1
```

3. **Définissez sur 0 la propriété de réponse de l'horodatage pour les paquets IP, puis contrôlez la valeur en cours.**

   La valeur par défaut supprime les demandes supplémentaires des CPU par rapport aux systèmes et empêche la diffusion d'informations sur le réseau.

```
# ipadm set-prop -p _respond_to_timestamp=0 ip
# ipadm show-prop -p _respond_to_timestamp ip
PROTO  PROPERTY                     PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
ip    _respond_to_timestamp         rw   0         --           0         0,1
```

4. **Définissez sur 0 la propriété de réponse de diffusion de l'horodatage pour les paquets IP, puis contrôlez la valeur en cours.**

   La valeur par défaut supprime les demandes supplémentaires des CPU par rapport aux systèmes et empêche la diffusion d'informations sur le réseau.

```
# ipadm set-prop -p _respond_to_timestamp_broadcast=0 ip
# ipadm show-prop -p _respond_to_timestamp_broadcast ip
PROTO  PROPERTY                     PERM CURRENT   PERSISTENT   DEFAULT   POSSIBLE
ip    _respond_to_timestamp_broadcast  rw   0         --           0         0,1
```

5. **Empêchez le routage IP par la source.**

   La valeur par défaut empêche les paquets de passer outre les mesures de sécurité du réseau. Les paquets routés par la source autorisent la source du paquet à suggérer un chemin différent du chemin configuré sur le routeur.

   Remarque - Ce paramètre peut être défini sur `1` à des fins de diagnostic. Une fois le diagnostic terminé, revenez à la valeur `0`.

```
# ipadm set-prop -p _rev_src_routes=0 tcp
# ipadm show-prop -p _rev_src_routes tcp
PROTO PROPERTY          PERM CURRENT  PERSISTENT  DEFAULT  POSSIBLE
tcp   _rev_src_routes   rw   0        --          0        0,1
```

   Pour plus d'informations, reportez-vous à la section " _rev_src_routes " du manuel " Manuel de référence des paramètres réglables d'Oracle Solaris 11.2 ".

Voir aussi Page de manuel `ipadm`(1M)

♦ ♦ ♦ CHAPITRE 3

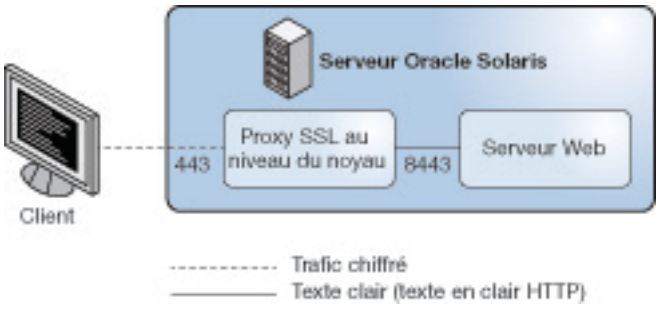
Serveurs Web et protocole SSL (Secure Sockets Layer)

Cette section décrit l'utilisation du protocole SSL (Secure Sockets Layer) pour chiffrer et accélérer les communications des serveurs Web d'un système Oracle Solaris.

Le proxy SSL au niveau du noyau chiffre les communications des serveurs Web

Tout serveur Web qui s'exécute sur Oracle Solaris peut être configuré de manière à utiliser le protocole SSL au niveau du noyau, c'est-à-dire le *proxy SSL au niveau du noyau*. C'est le cas par exemple du serveur Web Apache 2.2 et du serveur Oracle iPlanet Web Server. Le protocole SSL assure la confidentialité, l'intégrité des messages et l'authentification des points d'extrémité entre deux applications. Lorsque le proxy SSL au niveau du noyau s'exécute sur le serveur Web, les communications sont accélérées. L'illustration suivante présente la configuration de base.

FIGURE 3-1 Communications du serveur Web chiffrées par le noyau



Le proxy SSL au niveau du noyau met en oeuvre le côté serveur du protocole SSL. Il offre plusieurs avantages.

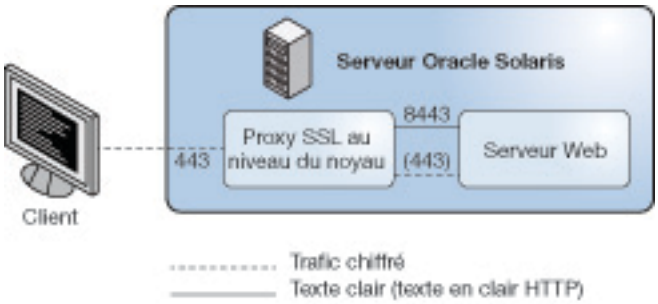
- Le proxy accélère les performances SSL pour les applications serveur, telles que les serveurs Web, de manière à ce qu'elles offrent de meilleures performances que des applications utilisant des bibliothèques SSL au niveau de l'utilisateur. L'amélioration des performances peut dépasser 35 %, en fonction de la charge de travail de l'application.
- Le proxy SSL au niveau du noyau est transparent. Aucune adresse IP ne lui est affectée. Par conséquent, les serveurs Web voient les adresses IP client réelles et les ports TCP.
- Le proxy SSL au niveau du noyau et les serveurs Web sont conçus pour fonctionner ensemble.

  La Figure 3-1, “Communications du serveur Web chiffrées par le noyau” illustre un scénario de base avec un serveur Web utilisant le proxy SSL au niveau du noyau. Le proxy SSL au niveau du noyau est configuré sur le port `443`, tandis que le serveur Web est configuré sur le port `8443`, où il reçoit des communications HTTP non chiffrées.
- Le proxy SSL au niveau du noyau peut être configuré de manière à utiliser le chiffrement au niveau de l'utilisateur comme solution de repli dans les cas où il ne prend pas en charge le chiffrement demandé.

  La Figure 3-2, “Communications de serveur Web chiffrées par le noyau avec option de repli sur le chiffrement au niveau de l'utilisateur” illustre un scénario plus complexe. Le serveur Web et le proxy SSL au niveau du noyau sont configurés de manière à utiliser le protocole SSL de serveur Web au niveau de l'utilisateur comme solution repli.

  Le proxy SSL au niveau du noyau est configuré sur le port `443`. Le serveur Web est configuré sur deux ports. Le port `8443` reçoit des communications HTTP non chiffrées, tandis que le port `443` est un port de repli. Le port de repli reçoit le trafic SSL chiffré pour les suites de chiffrement qui ne sont pas prises en charge par le proxy SSL au niveau du noyau.

FIGURE 3-2 Communications de serveur Web chiffrées par le noyau avec option de repli sur le chiffrement au niveau de l'utilisateur



Le proxy SSL au niveau du noyau prend en charge les protocoles SSL 3.0 et TLS 1.0, ainsi que la plupart des suites de chiffrement courantes. La page de manuel `ksslcfg`(1M) contient la liste complète. Le proxy peut être configuré de manière à se replier sur le serveur SSL au niveau de l'utilisateur pour les suites de chiffrement qui ne sont pas prises en charge.

Protection de serveurs Web à l'aide du proxy SSL au niveau du noyau

Les procédures suivantes indiquent comment configurer des serveurs Web de manière à ce qu'ils utilisent le proxy SSL au niveau du noyau :

- “Configuration d'un serveur Web Apache 2.2 afin qu'il utilise le proxy SSL au niveau du noyau” à la page 36
- “Configuration d'un serveur Oracle iPlanet Web Server afin qu'il utilise le proxy SSL au niveau du noyau” à la page 38
- “Configuration du proxy SSL au niveau du noyau de manière à utiliser le protocole SSL d'Apache 2.2 comme solution de repli” à la page 39
- “Utilisation du proxy SSL au niveau du noyau dans les zones” à la page 43

Configuration d'un serveur Web Apache 2.2 afin qu'il utilise le proxy SSL au niveau du noyau

Le proxy SSL au niveau du noyau peut améliorer la vitesse de traitement des paquets SSL sur un serveur Web Apache 2.2. Cette procédure implémente le scénario simple illustré à la Figure 3-1, “Communications du serveur Web chiffrées par le noyau”.

Avant de commencer Vous avez configuré un serveur Web Apache 2.2. Ce serveur Web est inclus dans Oracle Solaris.

Vous devez prendre le rôle `root`.

1. **Arrêtez le serveur Web.**

```
# svcadm disable svc:/network/http:apache22
```

2. **Placez la clé privée du serveur et le certificat du serveur dans un seul fichier.**

   Si seul le paramètre `SSLCertificateFile` est indiqué dans le fichier `ssl.conf`, le fichier spécifié peut être utilisé directement pour le proxy SSL au niveau du noyau.

   Si le paramètre `SSLCertificateKeyFile` est également indiqué, vous devez combiner le fichier de certificat et le fichier de clé privée. Exécutez une commande semblable à l'exemple suivant pour combiner les fichiers :

```
# cat cert.pem key.pem > cert-and-key.pem
```

3. **Déterminez les paramètres à utiliser avec la commande `ksslcfg`.**

   La page de manuel `ksslcfg`(1M) contient la liste complète des options. Voici les paramètres que vous *devez* fournir :

   - *key-format* : utilisé avec l'option `-f` pour définir le certificat et le format de clé. Pour le proxy SSL au niveau du noyau, les formats pris en charge sont les suivants : `pkcs11`, `pem` et `pkcs12`.
   - *key-and-certificate-file* : utilisé avec l'option `-i` pour définir l'emplacement du fichier qui stocke la clé de serveur et le certificat pour les options *key-format* `pem` ou `pkcs12`.
   - *password-file* : utilisé avec l'option `-p` pour obtenir le mot de passe servant à chiffrer la clé privée pour les options *key-format*`pem` ou `pkcs12`. Pour `pkcs11`, le mot de passe est utilisé pour authentifier le marqueur PKCS #11. Vous devez protéger le fichier de mots de passe à l'aide d'autorisations `0400`. Ce fichier est requis pour les réinitialisations sans l'intervention d'un opérateur.
   - *token-label* : utilisé avec l'option `-T` pour indiquer le marqueur PKCS #11.
   - *certificate-label* : utilisé avec l'option `-C` pour sélectionner l'étiquette de l'objet de certificat dans le marqueur PKCS#11.

- *proxy-port* : utilisé avec l'option `-x` pour définir le port proxy SSL. Vous devez spécifier un port différent du port standard `80`. Le serveur Web est à l'écoute du trafic de texte en clair non chiffré sur le port proxy SSL. Généralement, la valeur est `8443`.
- *ssl-port* : indique le port d'écoute pour le proxy SSL au niveau du noyau. Généralement, la valeur est `443`.

4. Créez l'instance de service pour le proxy SSL au niveau du noyau.

Indiquez le port proxy SSL et les paramètres associés à l'aide de l'un des formats suivants :

- **Indiquez PEM ou PKCS #12 comme format de clé.**

```
# ksslcfg create -f key-format -i key-and-certificate-file \
-p password-file -x proxy-port ssl-port
```

- **Indiquez PKCS #11 comme format de clé.**

```
# ksslcfg create -f pkcs11 -T PKCS11-token -C certificate-label \
-p password-file -x proxy-port ssl-port
```

5. Vérifiez que l'instance de service est en ligne.

```
# svcs svc:/network/ssl/proxy
STATE          STIME    FMRI
online         02:22:22 svc:/network/ssl/proxy:default
```

La sortie suivante indique que l'instance de service n'a pas été créée :

```
svcs: Pattern 'svc:/network/ssl/proxy' doesn't match any instances
STATE          STIME    FMRI
```

6. Configurez le serveur Web pour qu'il soit à l'écoute sur le port proxy SSL.

Modifiez le fichier `/etc/apache2/2.2/http.conf` et ajoutez une ligne pour définir le port proxy SSL. Si vous utilisez l'adresse IP du serveur, le serveur Web écoute uniquement sur cette interface. La ligne est similaire à ce qui suit :

```
Listen proxy-port
```

7. Définissez une dépendance SMF pour le serveur Web.

Le service du serveur Web peut uniquement démarrer une fois que l'instance proxy SSL au niveau du noyau a démarré. Les commandes suivantes établissent cette dépendance :

```
# svccfg -s svc:/network/http:apache22
svc:/network/http:apache22> addpg kssl dependency
...apache22> setprop kssl/entities = fmri:svc:/network/ssl/proxy:kssl-INADDR_ANY-443
...apache22> setprop kssl/grouping = astring: require_all
...apache22> setprop kssl/restart_on = astring: refresh
...apache22> setprop kssl/type = astring: service
...apache22> end
```

8. **Activez le service du serveur Web.**

```
# svcadm enable svc:/network/http:apache22
```

Configuration d'un serveur Oracle iPlanet Web Server afin qu'il utilise le proxy SSL au niveau du noyau

Le proxy SSL au niveau du noyau peut améliorer la vitesse de traitement des paquets SSL sur un serveur Oracle iPlanet Web Server. Cette procédure implémente le scénario simple illustré à la Figure 3-1, “Communications du serveur Web chiffrées par le noyau”.

Avant de commencer Vous avez installé et configuré un serveur Oracle iPlanet Web Server. Le serveur peut être téléchargé à partir de Oracle iPlanet Web Server (http://www.oracle.com/technetwork/middleware/iplanetwebserver-098726.html?ssSourceSiteId=ocomen). Pour plus d'instructions, reportez-vous à Oracle iPLANET WEB SERVER 7.0.15 (http://docs.oracle.com/cd/E18958_01/index.htm).

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network Security (sécurité réseau). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

1. **Arrêtez le serveur Web.**

   Utilisez l'interface Web de l'administrateur pour arrêter le serveur. Pour plus d'instructions, reportez-vous à Oracle iPLANET WEB SERVER 7.0.15 (http://docs.oracle.com/cd/E18958_01/index.htm).

2. **Déterminez les paramètres à utiliser avec la commande `ksslcfg`.**

   La page de manuel `ksslcfg`(1M) contient la liste complète des options. Pour obtenir la liste des paramètres que vous *devez* fournir, reportez-vous à Étape 3 in “Configuration d'un serveur Web Apache 2.2 afin qu'il utilise le proxy SSL au niveau du noyau” à la page 36.

3. **Créez l'instance de service pour le proxy SSL au niveau du noyau.**

   Indiquez le port proxy SSL et les paramètres associés à l'aide de l'un des formats suivants :

   - **Indiquez PEM ou PKCS #12 comme format de clé.**

     ```
     # ksslcfg create -f key-format -i key-and-certificate-file \
     -p password-file -x proxy-port ssl-port
     ```

   - **Indiquez PKCS #11 comme format de clé.**

```
# ksslcfg create -f pkcs11 -T PKCS11-token -C certificate-label \
-p password-file -x proxy-port ssl-port
```

4. **Vérifiez que l'instance est en ligne.**

```
# svcs svc:/network/ssl/proxy
STATE          STIME    FMRI
online         02:22:22 svc:/network/ssl/proxy:default
```

5. **Configurez le serveur Web pour qu'il soit à l'écoute sur le port proxy SSL.**

   Pour plus d'instructions, reportez-vous à Oracle iPLANET WEB SERVER 7.0.15 (http://docs.oracle.com/cd/E18958_01/index.htm).

6. **Définissez une dépendance SMF pour le serveur Web.**

   Le service du serveur Web peut uniquement démarrer une fois que l'instance proxy SSL au niveau du noyau a démarré. Les commandes suivantes établissent cette dépendance, en supposant que le FMRI du service de serveur Web est `svc:/network/http:Webserver7` :

```
# svccfg -s svc:/network/http:webserver7
svc:/network/http:webserver7> addpg kssl dependency
...webserver7> setprop kssl/entities = fmri:svc:/network/ssl/proxy:kssl-INADDR_ANY-443
...webserver7> setprop kssl/grouping = astring: require_all
...webserver7> setprop kssl/restart_on = astring: refresh
...webserver7> setprop kssl/type = astring: service
...webserver7> end
```

7. **Activez le service du serveur Web.**

```
# svcadm enable svc:/network/http:webserver7
```

▼ Configuration du proxy SSL au niveau du noyau de manière à utiliser le protocole SSL d'Apache 2.2 comme solution de repli

Dans cette procédure, vous configurez de toutes pièces un serveur Web Apache 2.2 et configurez le proxy SSL au niveau du noyau en tant que mécanisme de traitement de session SSL principal. Lorsque l'ensemble de chiffrements SSL proposé par le client n'inclut aucun chiffrement offert par le proxy SSL au niveau du noyau, le serveur Web Apache 2.2 sert de solution de repli. Cette procédure met en oeuvre le scénario complexe illustré à la Figure 3-2, “Communications de serveur Web chiffrées par le noyau avec option de repli sur le chiffrement au niveau de l'utilisateur”.

Avant de commencer Vous devez prendre le rôle `root`. Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Sur le serveur Apache 2.2, créez un certificat de clé destiné à être utilisé par le proxy SSL au niveau du noyau du serveur.**

   a. **Génération d'une demande de signature de certificat (CSR).**

   La commande suivante génère une demande de signature de certificat et la clé privée qui lui est associée pour le proxy SSL au niveau du noyau :

```
# cd /root
# openssl req \
> -x509 -new \
> -subj "/C=CZ/ST=Prague region/L=Prague/CN=`hostname`" \
> -newkey rsa:2048 -keyout webkey.pem \
> -out webcert.pem
Generating a 2048 bit RSA private key
.+++
........+++
writing new private key to 'webkey.pem'
Enter PEM pass phrase: JohnnyCashIsCool
Verifying - Enter PEM pass phrase:  JohnnyCashIsCool
#
# chmod 440 /root/webcert.pem ; chown root:webservd /root/webcert.pem
```

Remarque - La longueur minimale de clé RSA est de 2 048 pour être conforme à FIPS 140. Pour plus d'informations, reportez-vous à " Using a FIPS 140 Enabled System in Oracle Solaris 11.2 ".

   Pour plus d'informations, reportez-vous à la page de manuel `openssl`(5).

   b. **Envoyez la demande de signature de certificat à votre autorité de certification (CA).**

   c. **Remplacez le fichier `Webcert.pem` par le certificat signé fourni par votre autorité de certification.**

2. **Configurez le proxy SSL au niveau du noyau avec une phrase de passe et le certificat de clé publique/privée.**

   a. **Créez, enregistrez et protégez la phrase de passe.**

```
# echo "RefrigeratorsAreCool" > /root/kssl.pass
# chmod 440 /root/kssl.pass; chown root:webservd /root/kssl.pass
```

Remarque - Le nom d'hôte ne peut pas contenir d'espaces.

b. Rassemblez le certificat de clé privée et le certificat de clé publique dans un seul fichier.

```
# cat /root/webcert.pem /root/webkey.pem > /root/webcombo.pem
```

c. Configurez le proxy SSL au niveau du noyau avec le certificat de clé publique/privée et la phrase de passe.

```
# ksslcfg create -f pem -i /root/webcombo.pem -x 8443 -p /root/kssl.pass 443
```

3. Configurer le serveur Web afin qu'il pour le processus d'écoute reçoive les demandes des communications non chiffrées sur le port 8443.

Modifiez la ligne `Listen` dans le fichier `/etc/apache2/2.2/httpd.conf`.

```
# pfedit /etc/apache2/2.2/httpd.conf
...
## Listen 80
Listen 8443
```

4. Ajoutez le modèle de module SSL, `ssl.conf`, au répertoire de configuration Apache.

```
# cp /etc/apache2/2.2/samples-conf.d/ssl.conf /etc/apache2/2.2/ssl.conf
```

Ce module ajoute l'écoute des connexions chiffrées sur le port 443.

5. Activez le serveur Web pour déchiffrer la phrase de passe dans le fichier `/root/kssl.pass`.

a. Créez un script shell qui lit le fichier `kssl.pass`.

```
# pfedit /root/put-passphrase.sh
#!/usr/bin/ksh -p
## Reads proxy SSL au niveau du noyau passphrase
/usr/bin/cat /root/kssl.pass
```

b. Définissez le script comme exécutable et protégez le fichier.

```
# chmod 500 /root/put-passphrase.sh
# chown webservd:webservd /root/put-passphrase.sh
```

c. Modifiez le paramètre `SSLPassPhraseDialog` dans le fichier `ssl.conf` de manière à appeler ce script shell.

```
# pfedit /etc/apache2/2.2/ssl.conf
```

```
...
## SSLPassPhraseDialog  builtin
SSLPassPhraseDialog exec:/root/put-passphrase.sh
```

6. Placez les certificats de clés publique et privée du serveur Web aux emplacements appropriés.

Les valeurs des paramètres `SSLCertificateFile` et `SSLCertificateKeyFile` dans le fichier `ssl.conf` indiquent les emplacements et les noms attendus. Vous pouvez copier les certificats aux emplacements corrects ou créer des liens vers ces emplacements.

```
# ln -s  /root/webcert.pem /etc/apache2/2.2/server.crt     SSLCertificateFile default location
# ln -s /root/webkey.pem /etc/apache2/2.2/server.key      SSLCertificateKeyFile default location
```

7. Activez le service Apache.

```
# svcadm enable apache22
```

8. (Facultatif) Vérifiez que les deux ports fonctionnent.

Utilisez les commandes `openssl s_client` et `kstat` pour afficher les paquets.

a. Utilisez un chiffrement qui est disponible au proxy SSL au niveau du noyau.

```
# openssl s_client -cipher RC4-SHA -connect web-server:443
```

Une augmentation de `1` du compteur `kstatkssl_full_handshakes` vérifie que la session SSL a été traitée par le proxy SSL au niveau du noyau.

```
# kstat -m kssl -s kssl_full_handshakes
```

b. Utilisez un chiffrement qui n'est pas disponible au proxy SSL au niveau du noyau.

```
# openssl s_client -cipher CAMELLIA256-SHA -connect web-server:443
```

Une augmentation de `1` du compteur `kstatkssl_fallback_connections` vérifie que le paquet est arrivé mais que la session SSL a été traitée par le serveur Web Apache.

```
# kstat -m kssl -s kssl_fallback_connections
```

Exemple 3-1 Configuration d'un serveur Web Apache 2.2 afin qu'il utilise le proxy SSL au niveau du noyau

La commande suivante permet de créer une instance de service pour le proxy SSL au niveau du noyau qui utilise le format de clé `pem` :

```
# ksslcfg create -f pem -i cert-and-key.pem -p kssl.pass -x 8443 443
```

▼ Utilisation du proxy SSL au niveau du noyau dans les zones

Le proxy SSL au niveau du noyau fonctionne dans les zones avec les restrictions suivantes :

- L'ensemble de la gestion SSL au niveau du noyau doit être réalisée dans la zone globale. L'administrateur de la zone globale doit pouvoir accéder aux fichiers de clés et de certificat de la zone locale. Vous pouvez démarrer le serveur Web dans la zone non globale une fois que vous avez configuré l'instance de service à l'aide de la commande dans la zone globale. `ksslcfg`
- Un nom d'hôte ou une adresse IP spécifique doit être spécifié(e) avec la commande `ksslcfg` lorsque vous configurez l'instance. En particulier, l'instance ne peut pas spécifier `INADDR_ANY` pour l'adresse IP.

Avant de commencer

Le service du serveur Web est configuré et activé dans la zone non globale.

Vous devez vous connecter en tant qu'administrateur disposant des profils de droits Network Security (sécurité réseau) et Zone Management (gestion de zone). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Dans la zone non globale, arrêtez le serveur Web.**

   Par exemple, pour arrêter un serveur Web Apache dans la zone `apache-zone`, exécutez la commande suivante :

   ```
   apache-zone # svcadm disable svc:/network/http:apache22
   ```

2. **Dans la zone globale, créez l'instance de service pour le proxy SSL au niveau du noyau dans la zone.**

   Pour créer une instance de service pour `apache-zone`, utilisez une commande semblable à ce qui suit :

   ```
   # ksslcfg create -f pem -i /zone/apache-zone/root/keypair.pem \
   -p /zone/apache-zone/root/skppass -x 8443 apache-zone 443
   ```

3. **Dans la zone non globale, activez l'instance de service Web.**

   Activez par exemple le service Web dans `apache-zone`.

   ```
   apache-zone # svcadm enable svc:/network/http:apache22
   ```

♦ ♦ ♦ CHAPITRE 4

A propos d'IP Filter dans Oracle Solaris

Ce chapitre fournit une présentation de la fonction IP Filter d'Oracle Solaris. Les tâches IP Filter sont décrites au Chapitre 5, Configuration d'IP Filter.

Le présent chapitre contient les informations suivantes :

Introduction à IP Filter

La fonction IP Filter d'Oracle Solaris est un pare-feu qui assure un filtrage de paquets avec état et la translation d'adresse réseau (NAT). IP Filter permet également le filtrage de paquets sans état, ainsi que la création et la gestion des pools d'adresses.

Le filtrage de paquets assure une protection de base contre les attaques potentielles via le réseau. IP Filter peut filtrer les paquets par adresse IP, port, protocole, interface réseau et direction du trafic. IP Filter peut également filtrer en fonction d'une adresse IP source individuelle, d'une adresse IP de destination, d'une plage d'adresses IP ou par pools d'adresses.

IP Filter est dérivé du logiciel Open Source IP Filter. Les conditions de licence, attribution et déclarations de copyright pour Open Source IP Filter sont accessibles via le chemin par défaut `/usr/lib/ipf/IPFILTER.LICENCE`. Si vous avez installé Oracle Solaris à un autre emplacement que celui par défaut, modifiez le chemin indiqué afin d'accéder au fichier à l'emplacement où il a été installé.

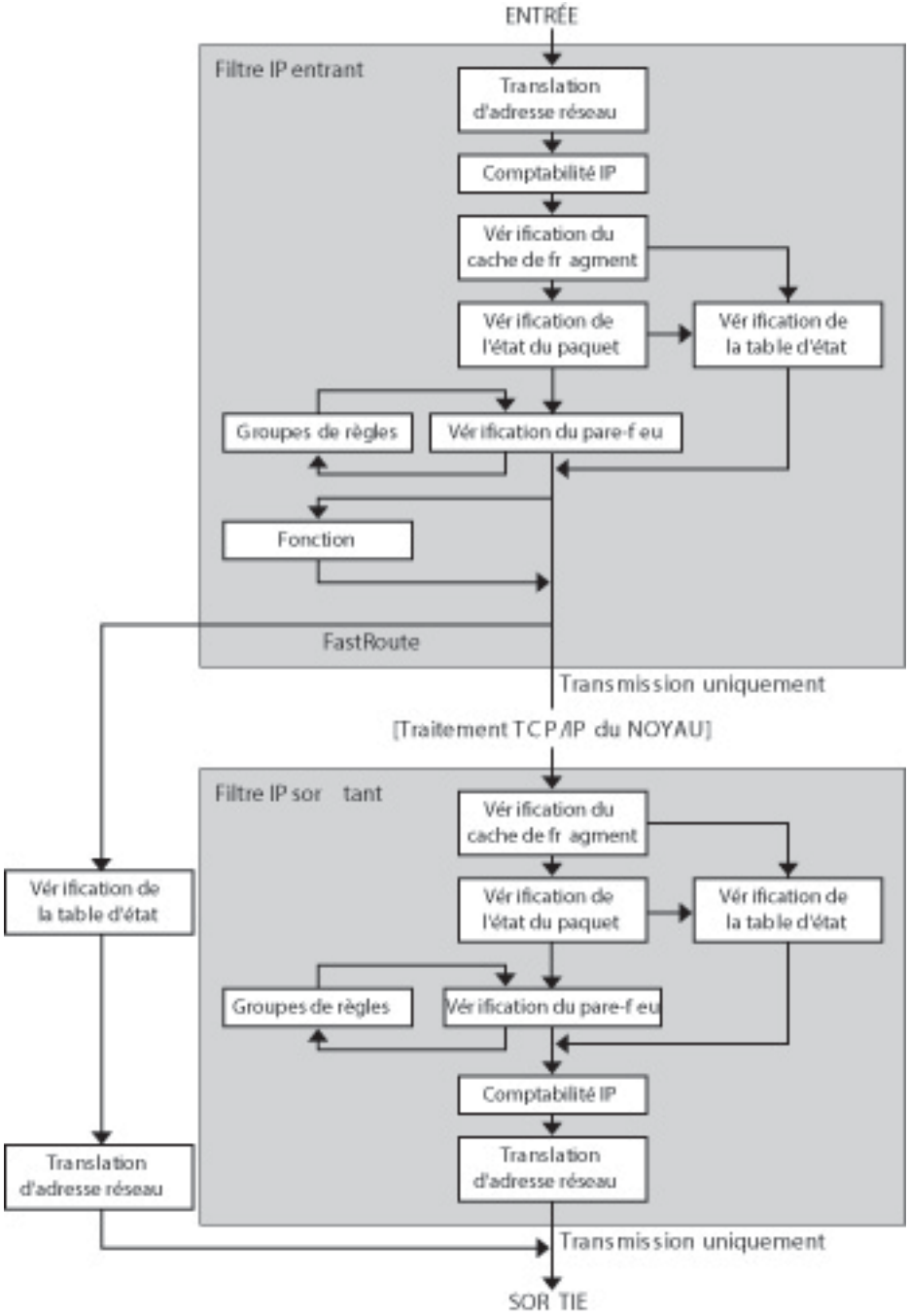
Sources d'informations pour Open Source IP Filter

La page d'accueil du logiciel Open Source IP Filter de Darren Reed se trouve à l'adresse http://coombs.anu.edu.au/~avalon/ip-filter.html. Ce site Web fournit des informations relatives au logiciel Open Source IP Filter, notamment un lien vers le didacticiel "IP Filter Based Firewalls HOWTO" (Brendan Conoboy et Erik Fichtner, 2002). Vous trouverez dans ce didacticiel les instructions de construction de pare-feux dans un environnement BSD UNIX, expliquées pas à pas. Bien qu'il soit destiné à un environnement BSD UNIX, ce didacticiel est également applicable à la configuration d'IP Filter sur Oracle Solaris.

Traitement des paquets avec IP Filter

Au cours du traitement d'un paquet, IP Filter exécute une séquence d'étapes. Le diagramme ci-dessous illustre les étapes du traitement d'un paquet et l'intégration du filtrage à la pile de protocole TCP/IP.

FIGURE 4-1 Séquence de traitement d'un paquet



Le traitement d'un paquet inclut les opérations suivantes :

- **Translation d'adresse réseau (NAT)**

  Translation d'une adresse IP privée vers une adresse publique, ou définition d'alias pour plusieurs adresses privées vers une adresse publique unique. NAT (Network Address Translation, translation d'adresse réseau) permet à une organisation de résoudre le problème d'épuisement des adresses IP lorsqu'elle utilise un réseau et requiert l'accès à Internet.
- **Comptabilisation IP**

  Les règles d'entrée et de sortie peuvent être configurées séparément, avec enregistrement du nombre d'octets transmis. En cas de correspondance d'une règle, le nombre d'octets du paquet est ajouté à la règle et des statistiques en cascade sont rassemblées.
- **Vérification du cache de fragments**

  Par défaut, les paquets fragmentés sont mis en mémoire cache. Lorsque tous les fragments d'un paquet particulier arrivent, les règles de filtrage sont appliquées et les fragments sont soit autorisés, soit bloqués. Si `set defrag off` figure dans le fichier de règles, les fragments ne sont pas mis en cache.
- **Vérification de l'état du paquet**

  Si la règle contient l'instruction `keep state`, tous les paquets d'une session spécifiée sont automatiquement transmis ou bloqués, selon la spécification de la règle : `pass` (transmettre) ou `block` (bloquer).
- **Vérification du pare-feu**

  Les règles d'entrée et de sortie peuvent être configurées séparément, afin d'autoriser ou non la transmission d'un paquet, via IP Filter, vers les routines TCP/IP du noyau ou vers le réseau.
- **Groupes**

  Les groupes permettent d'écrire des ensembles de règles selon une structure arborescente.
- **Fonction**

  Une fonction correspond à l'action à réaliser. Les fonctions sont, par exemple, `block` (bloquer), `pass` (transmettre), `literal` (littéral) et `send ICMP response` (envoyer une réponse ICMP).
- **Fast-route**

  FastRoute indique à IP Filter de ne pas transmettre le paquet vers la pile UNIX IP pour le routage, ce qui entraîne une réduction TTL.
- **Authentification IP**

  Les paquets authentifiés ne sont transmis via les boucles du pare-feu qu'une seule fois, afin d'éviter le traitement multiple de certains paquets.

Recommandations relatives à l'utilisation d'IP Filter

- IP Filter est géré par le service SMF `svc:/network/ipfilter`. Pour une présentation complète de SMF, reportez-vous au Chapitre 1, “ Introduction à l’utilitaire de gestion des services ” du manuel “ Gestion des services système dans Oracle Solaris 11.2 ”. Pour plus d'informations sur les procédures étape par étape associées à SMF, reportez-vous au Chapitre 3, “ Administration de services ” du manuel “ Gestion des services système dans Oracle Solaris 11.2 ”.
- IP Filter requiert la modification directe des fichiers de configuration.
- IP Filter est installé en tant que composant d'Oracle Solaris. Par défaut, le service IP Filter est activé lorsque votre système est configuré de façon à utiliser la mise en réseau automatique. Le profil réseau automatique, comme décrit dans les pages de manuel `nwam`(5) et `netadm`(1M), active ce pare-feu. Pour une configuration personnalisée sur un système mis en réseau automatiquement, le service IP Filter n'est pas activé. Pour obtenir la description des tâches associées à l'activation du service, reportez-vous à la section “Configuration du service IP Filter” à la page 59.
- Pour administrer IP Filter, vous devez prendre le rôle `root` ou disposer du profil de droits IP Filter Management (gestion IP Filter). Vous pouvez attribuer le profil de droits de gestion IP Filter à un utilisateur ou à un rôle créé par vous. Pour créer le rôle et assigner le rôle à un utilisateur, reportez-vous à dans “ Création d’un rôle ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.
- Le logiciel Oracle Solaris Cluster ne prend pas en charge le filtrage IP Filter pour les services évolutifs. En revanche, IP Filter est pris en charge pour les services de basculement. Pour connaître les recommandations et restrictions relatives à IP Filter dans un cluster, reportez-vous à la section "Restrictions concernant les fonctions du SE Oracle Solaris" dans le *Guide d'installation du logiciel Oracle Solaris Cluster*.
- Le filtrage interzones est pris en charge à condition que les règles IP Filter soient implémentées dans une zone qui fonctionne en tant que routeur virtuel pour les autres zones du système.

Utilisation des fichiers de configuration IP Filter

IP Filter peut assurer des services de pare-feu ou de translation d'adresse réseau (NAT, Network Address Translation). Aucune règle de pare-feu et de translation NAT n'est fournie par défaut. Vous devez créer des fichiers de configuration personnalisés et définir les chemins d'accès à ces fichiers en tant que valeurs de propriétés du service IP Filter. Une fois que le service est activé, ces fichiers sont chargés automatiquement lorsque le système est réinitialisé. Pour obtenir des exemples de fichiers de configuration, reportez-vous à la section “Exemples de fichiers de configuration IP Filter” à la page 85. Pour plus d'informations, reportez-vous à la page de manuel `svc.ipfd`(1M).

Utilisation d'ensembles de règles IP Filter

Pour gérer le pare-feu, vous devez spécifier des ensembles de règles à l'aide d'IP Filter, puis filtrer le trafic réseau en fonction de ces ensembles de règles. Les types d'ensembles de règles suivants sont disponibles :

- ensembles de règles de filtrage de paquets ;
- Ensembles de règles NAT (Network Address Translation, translation d'adresse réseau)

En outre, vous pouvez créer des pools d'adresses pour référencer les groupes d'adresses IP. Ensuite, ces pools peuvent être utilisés dans un ensemble de règles. Les pools d'adresses permet de retrouver plus rapidement le traitement des règles. En outre, ils facilitent la gestion des grands groupes d'adresses.

Utilisation de la fonctionnalité de filtrage de paquets d'IP Filter

Vous pouvez configurer le filtrage de paquets à l'aide des ensembles de règles de filtrage de paquets. La commande `ipf` permet d'utiliser les ensembles de règles de filtrage de paquets. Pour plus d'informations sur la commande `ipf`, reportez-vous à la page de manuel `ipf`(1M).

Vous pouvez créer des règles de filtrage de paquets via la ligne de commande, à l'aide de la commande `ipf` ou dans un fichier de configuration de filtrage de paquets. Pour charger le fichier de configuration, vous devez le créer, puis indiquer son chemin d'accès au service IP Filter.

Avec IP Filter, deux ensembles de règles de filtrage de paquets peuvent coexister : l'ensemble de règles actif et l'ensemble de règles inactif. Dans la plupart des cas, vous utiliserez l'ensemble de règles actif. Toutefois, la commande `ipf -I` vous permet d'appliquer l'action de la commande à la liste de règles inactives. La liste de règles inactives n'est pas employée par IP Filter, sauf si vous la sélectionnez. La liste de règles inactives constitue l'emplacement auquel vous pouvez enregistrer des règles sans affecter le filtrage de paquets actif.

IP Filter traite les règles de la liste de règles du début à la fin de la liste de règles configurée, puis transmet ou bloque le paquet. Un indicateur permet à IP Filter de déterminer si un paquet doit être transmis ou non. Il parcourt l'intégralité de l'ensemble de règles et détermine si le paquet doit être transmis ou bloqué, en fonction de la dernière règle correspondante.

Il existe deux exceptions à ce processus. D'une part, si le paquet correspondant à une règle contenant le mot-clé `quick`, Si une règle inclut le mot-clé `quick`, l'action associée à cette règle est exécutée et les règles suivantes sont ignorées. D'autre part, si le paquet correspond

à une règle contenant le mot-clé `group`, seules les règles portant l'indicateur de ce group sont vérifiées.

Configuration des règles de filtrage de paquets

Appliquez la syntaxe suivante pour créer des règles de filtrage de paquets :

action `[in|out]` *option keyword, keyword...*

1. Chaque règle commence par une action. IP Filter applique l'action au paquet si celui-ci correspond à la règle. Les actions habituellement appliquées aux paquets sont répertoriées ci-dessous.

`block`	Empêche le paquet de traverser le filtre.
`pass`	Permet au paquet de traverser le filtre.
`log`	Consigne le paquet sans déterminer s'il est bloqué ou transmis. Exécutez la commande `ipmon` pour afficher le journal.
`count`	Inclut le paquet dans les statistiques du filtre. Exécutez la commande `ipfstat` pour afficher les statistiques.
`skip` *number*	Entraîne le filtre à ignorer les règles de filtrage *number*.
`auth`	Le paquet est authentifié par un programme utilisateur qui valide les informations qu'il contient. Le programme détermine si le paquet est transmis ou bloqué.

2. Le mot suivant est `in` ou `out`. Votre choix détermine si la règle de filtrage de paquets est appliquée à un paquet entrant (in) ou à un paquet sortant (out).
3. Ensuite, vous pouvez insérer toute une liste d'options. Si vous en utilisez plusieurs, elles doivent être dans l'ordre indiqué ci-dessous.

`log`	Consigne le paquet si la règle constitue la dernière règle correspondante. Exécutez la commande `ipmon` pour afficher le journal.
`quick`	Exécute la règle contenant l'option `quick` si un paquet lui correspond. Toute vérification de règle subséquente est interrompue.
`on` *interface-name*	Applique la règle uniquement si le paquet entre ou sort via l'interface spécifiée.

`dup-to` *interface-name* Copie le paquet et envoie la copie sur *interface-name* vers une adresse IP éventuellement spécifiée.

Remarque - L'option `dup-to` dans une règle permet à l'administrateur du réseau de créer un *TAP réseau*. Bien que cette option soit toujours prise en charge dans Oracle Solaris, elle a perdu en importance. Les commutateurs sont disponibles moderne qui vous permettent de pour effectuer la configuration directement pressions les ports correspondants, qui annule le réseau est nécessaire de définir cette fonction dans une règle. Pour, reportez-vous à la documentation de votre commutateur réseau Configuration des ports toucher le.

`to` *interface-name* Déplace le paquet vers une file d'attente sortante sur *interface-name*.

4. Une fois les options spécifiées, vous avez le choix entre plusieurs mots-clés afin de déterminer si le paquet correspond à la règle. Les mots-clés doivent être utilisés dans l'ordre indiqué ci-dessous.

Remarque - Par défaut, le filtre autorise la transmission de tout paquet ne correspondant à aucune règle du fichier de configuration.

`tos` Filtre les paquets en fonction de leur type de service, exprimé sous forme d'entier décimal ou hexadécimal.

`ttl` Filtre les paquets en fonction de leur durée de vie. La durée de vie d'un paquet est une valeur enregistrée dans celui-ci qui indique la durée pendant laquelle le paquet peut se trouver sur le réseau avant d'être abandonné.

`proto` Les paquets correspondant à la règle sont déterminés en fonction d'un protocole spécifique. Vous pouvez employer l'un des noms de protocole spécifiés dans le fichier `/etc/protocols` ou un nombre décimal représentant le protocole. Le mot-clé `tcp/udp` peut être utilisé pour filtrer les paquets TCP ou UDP.

`from/to/all/any` Filtre les paquets en fonction des éléments suivants : l'adresse IP source, l'adresse IP de destination et le numéro de port. Le mot-clé `all` permet d'accepter tous les paquets, quelles que soient leur source et leur destination.

`with` Les paquets correspondant à la règle sont ceux qui sont associés à des attributs spécifiques. Insérez le mot `not` ou `no` devant le mot-clé pour indiquer qu'un paquet ne correspond à la règle qu'en l'absence de l'option.

`flags`	Employé pour TCP afin de filtrer les paquets en fonction des indicateurs TCP définis. Pour plus d'informations sur les indicateurs TCP, reportez-vous à la page de manuel `ipf`(4).
`icmp-type`	Filtre les paquets en fonction du type d'ICMP. Ce mot-clé n'est employé que si l'option `proto` est définie sur `icmp` et que l'option `flags` n'est pas utilisée.
`keep` *keep-options*	Détermine les informations conservées pour le paquet. Les *keep-options* disponibles incluent l'option `state`. L'option `state` conserve des informations sur la session et peut être appliquée aux paquets TCP, UDP et ICMP.
`head` *number*	Crée un groupe pour les règles de filtrage, désigné par le numéro *numéro*.
`group` *number*	Ajoute la règle au groupe de numéro *numéro* au lieu du groupe par défaut. Toutes les règles de filtrage sont placées dans le groupe 0 si aucun autre groupe n'est spécifié.

L'exemple suivant illustre la constitution d'une syntaxe de règle de filtrage de paquets pour créer une règle. Pour bloquer le trafic entrant à partir de l'adresse IP `192.168.0.0/16`, ajoutez la règle suivante à la liste de règles :

```
block in quick from 192.168.0.0/16 to any
```

Pour obtenir la syntaxe et la grammaire complètes utilisées pour écrire des règles de filtrage de paquets, reportez-vous à la page de manuel `ipf`(4). Pour une description des tâches associées au filtrage de paquets, reportez-vous à la section “Gestion des ensembles de règles de filtrage de paquets d'IP Filter” à la page 66. Pour une explication du schéma d'adressage IP (`192.168.0.0/16`) présenté dans l'exemple, reportez-vous au Chapitre 1, “ Planification du développement du réseau ” du manuel “ Planification du développement du réseau dans Oracle Solaris 11.2 ”.

Utilisation de la fonctionnalité NAT d'IP Filter

NAT configure des règles de mappage qui réalisent la translation des adresses IP source et de destination vers d'autres adresses Internet ou intranet. Ces règles modifient les adresses source et de destination des paquets IP entrants et sortants et envoient les paquets. Vous pouvez également utiliser NAT pour rediriger le trafic d'un port à un autre. NAT assure l'intégrité du paquet en cas de modification ou de redirection de celui-ci.

Vous pouvez créer des règles NAT à la ligne de commande, à l'aide de la commande `ipnat` ou dans un fichier de configuration NAT. Vous devez créer le fichier de configuration NAT et définir son chemin d'accès comme valeur de la propriété `config/ipnat_config_file` du service. La valeur par défaut est `/etc/ipf/ipnat.conf`. Pour plus d'informations, reportez-vous à la commande `ipnat`(1M).

Les règles NAT peuvent s'appliquer à la fois aux adresses IPv4 et IPv6. Au contraire, vous devez définir des règles distinctes pour chaque type d'adresse. Dans une règle NAT qui inclut les adresses IPv6, vous ne pouvez pas utiliser les commandes NAT `mapproxy` et `rdrproxy` simultanément.

Configuration des règles NAT

Appliquez la syntaxe ci-dessous pour créer des règles NAT :

command interface-name parameters

1. Toute règle commence par l'une des commandes ci-dessous :

`map`	Mappe une adresse IP ou un réseau IP vers une autre adresse IP ou un autre réseau IP selon un processus circulaire non contrôlé.
`rdr`	Redirige les paquets d'un couple port-adresse IP vers un autre couple port-adresse IP.
`bimap`	Etablit une translation d'adresse réseau bidirectionnelle entre une adresse IP externe et une adresse IP interne.
`map-block`	Etablit la translation basée sur les adresses IP statiques. Cette commande se base sur un algorithme qui force la translation des adresses vers une plage de destination.

2. Le mot suivant correspond au nom de l'interface, par exemple `bge0`.
3. Ensuite, vous avez le choix entre divers paramètres, afin de définir la configuration NAT. Les paramètres suivants sont disponibles :

`ipmask`	Désigne le masque réseau.
`dstipmask`	Désigne l'adresse cible de la translation de `ipmask`.
`mapport`	Désigne les protocoles `tcp`, `udp` ou `tcp/udp`, ainsi qu'une plage de numéros de port.

L'exemple ci-dessous indique comment construire une règle NAT. Pour réécrire un paquet sortant sur le périphérique `net2` avec l'adresse source `192.168.1.0/24` et pour afficher son

adresse source comme étant `10.1.0.0/16`, ajoutez la règle ci-dessous à l'ensemble de règles NAT :

```
map net2 192.168.1.0/24 -> 10.1.0.0/16
```

Les règles suivantes s'appliquent aux adresses IPv6 :

```
map net3 fec0:1::/64 -> 2000:1:2::/72 portmap tcp/udp 1025:65000
map-block net3 fe80:0:0:209::/64 -> 209:1:2::/72 ports auto
rdr net0 209::ffff:fe13:e43e port 80 -> fec0:1::e,fec0:1::f port 80 tcp round-robin
```

Pour obtenir la syntaxe et la grammaire complètes, reportez-vous à la page de manuel `ipnat(4)`.

Utilisation de la fonctionnalité de pools d'adresses d'IP Filter

Les pools d'adresses constituent une seule référence pour un groupe de paires adresse / masque de réseau. Les pools d'adresses permettent de trouver plus rapidement les adresses IP correspondant aux règles. En outre, ils facilitent la gestion des grands groupes d'adresses.

Les règles de configuration de pool d'adresses peuvent se trouver dans un fichier qui est chargé par le service IP Filter. Vous devez créer un fichier, puis définir son chemin d'accès comme valeur de la propriété `config/ippool_config_file` du service. La valeur par défaut est `/etc/ipf/ippool.conf`.

Configuration des pools d'adresses

Pour créer un pool d'adresses, appliquez la syntaxe suivante :

`table role =` *role-name* `type =` *storage-format* `number =` *reference-number*

`table` Définit la référence des adresses.

`role` Spécifie le rôle du pool dans IP Filter. Vous ne pouvez faire référence qu'au rôle `ipf`.

`type` Spécifie le format de stockage du pool.

`number` Spécifie le numéro de référence utilisé par la règle de filtrage.

Par exemple, pour faire référence aux groupes d'adresses `10.1.1.1 et 10.1.1.2` et au réseau `192.16.1.0` à l'aide du numéro de pool `13`, insérez la règle suivante dans le fichier de configuration de pool d'adresses :

```
table role = ipf type = tree number = 13
{ 10.1.1.1/32, 10.1.1.2/32, 192.168.1.0/24 };
```

Ensuite, pour faire référence au numéro de pool `13` dans une règle de filtrage, élaborez une règle similaire à la suivante :

```
pass in from pool/13 to any
```

Vous devez charger le fichier de pool avant de charger les règles contenant une référence au pool. Dans le cas contraire, le pool n'est pas défini, comme indiqué dans la sortie suivante :

```
# ipfstat -io
empty list for ipfilter(out)
block in from pool/13(!) to any
```

Si vous ajoutez le pool par la suite, l'ensemble de règle du noyau n'est pas mis à jour. Vous devez également recharger le fichier de règles faisant référence au pool.

Pour obtenir la syntaxe et la grammaire complètes, reportez-vous à la page de manuel `ippool`(4).

IPv6 pour IP Filter

Ce filtrage peut se baser sur l'adresse IPv6 source/de destination, sur les pools contenant des adresses IPv6 et sur les en-têtes d'extension IPv6.

De nombreux aspects d'IPv6 sont similaires à IPv4. Toutefois, les en-têtes et tailles des paquets ne sont pas identiques dans les deux versions d'IP, ce qui constitue une considération de poids pour IP Filter. Les paquets IPv6 appelés *jumbogrammes* contiennent un datagramme de longueur supérieure à 65 535 octets. IP Filter ne prend pas en charge les jumbogrammes IPv6.

Remarque - Pour plus d'informations sur les jumbograms, reportez-vous à IPv6 Jumbograms, RFC 2675 (http://www.ietf.org/rfc/rfc2675.txt).

Les tâches IP Filter associées à IPv6 ne sont pas très différentes d'IPv4. La différence la plus notable est l'emploi de l'option `-6` avec certaines commandes. Les commandes `ipf` et `ipfstat` incluent l'option `-6` à utiliser avec le filtrage de paquets IPv6. Appliquez l'option `-6` avec la commande `ipf` pour charger et vider les règles de filtrage de paquets IPv6. Pour afficher les statistiques IPv6, utilisez l'option `-6` avec la commande `ipfstat`. Les commandes `ipmon` et `ippool` prennent également en charge IPv6, même si aucune option n'est associée à la prise en charge d'IPv6. La commande `ipmon` a été optimisée pour autoriser la journalisation des paquets IPv6. La commande `ippool` prend en charge les pools avec les adresses IPv6. Vous pouvez créer des pools distincts pour les adresses IPv4 et IPv6, ou un pool contenant à la fois des adresses IPv4 et IPv6.

Pour créer des règles de filtrage de paquets IPv6 réutilisables, vous devez créer un fichier IPv6 spécifique. Puis vous définissez son chemin d'accès comme valeur de la propriété `config/ip6_config_file` du service d'IP Filter. La valeur par défaut est `/etc/ipf/ip6.conf`.

Pour obtenir une description des tâches associées à IP Filter, reportez-vous au Chapitre 5, Configuration d'IP Filter.

Pages de manuel IP Filter

Les pages de manuel suivantes traitent d'IP Filter.

`ipf`(1M)	Gère les règles IP Filter, affiche les paramètres réglables et effectue d'autres tâches.
`ipf`(4)	Contient la grammaire et la syntaxe de création des règles de filtrage de paquets IP Filter.
`ipfilter`(5)	Décrit le logiciel IP Filter.
`ipfs`(1M)	Enregistre et restaure les informations NAT et les informations de table d'état d'une réinitialisation à l'autre.
`ipfstat`(1M)	Extrait et affiche les statistiques relatives au traitement des paquets.
`ipmon`(1M)	Ouvre le périphérique de journalisation et affiche les paquets consignés pour le filtrage de paquets et la translation NAT.
`ipnat`(1M)	Gère les règles NAT et affiche les statistiques NAT.
`ipnat`(4)	Contient la grammaire et la syntaxe pour la création de règles NAT.
`ippool`(1M)	Crée et gère des pools d'adresses.
`ippool`(4)	Contient la grammaire et la syntaxe de création des pools d'adresses IP Filter.
`svc.ipfd`(1M)	Fournit des informations sur la configuration du service IP Filter.

♦ ♦ ♦ CHAPITRE 5

Configuration d'IP Filter

Ce chapitre fournit les instructions relatives à chaque étape des tâches pour IP Filter. Pour obtenir des informations générales, reportez-vous au Chapitre 4, A propos d'IP Filter dans Oracle Solaris.

Ce chapitre comprend les sections suivantes :

Configuration du service IP Filter

La liste des tâches ci-dessous répertorie les procédures permettant de créer des règles IP Filter ainsi que d'activer et de désactiver le service.

TABLEAU 5-1 Liste des tâches pour la configuration du service IP Filter

Tâche	Pour obtenir des instructions
Affichage des fichiers utilisés par IP Filter et du statut du service.	"Affichage des valeurs par défaut du service IP Filter" à la page 60
Personnalisation des ensembles de règles de filtrage de paquets pour le trafic réseau, de paquets sur un NAT et de pools d'adresses.	"Création de fichiers de configuration IP Filter" à la page 61
Activation, désactivation ou actualisation du service IP Filter.	"Activation et actualisation d'IP Filter" à la page 63
Modification de la valeur par défaut des paquets arrivant en fragments.	"Désactivation du réassemblage des paquets" à la page 63
Filtrage du trafic entre les zones sur votre système.	"Activation du filtrage de loopback" à la page 64
Arrêt de l'utilisation d'IP Filter.	"Désactivation du filtrage de paquets" à la page 65

▼ Affichage des valeurs par défaut du service IP Filter

Avant de commencer

Pour exécuter la commande `ipfstat`, vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Affichez les noms de fichiers de configuration et les emplacements pour le service IP Filter.**

```
$ svccfg -s ipfilter:default listprop   | grep file
config/ipf6_config_file                     astring     /etc/ipf/ipf6.conf
config/ipnat_config_file                    astring     /etc/ipf/ipnat.conf
config/ippool_config_file                   astring     /etc/ipf/ippool.conf
firewall_config_default/custom_policy_file  astring     none
```

Les trois premières propriétés de fichiers ont des emplacements par défaut. Ces fichiers n'existent pas tant que vous ne les avez pas créés. Si vous déplacez un fichier de configuration, vous devez modifier sa valeur de propriété. Pour plus d'informations sur cette procédure, reportez-vous à la section "Création de fichiers de configuration IP Filter" à la page 61.

Vous modifiez la quatrième propriété de fichier lorsque vous personnalisez vos propres règles de filtrage de paquets. Reportez-vous à l'Étape 1 et à l'Étape 2 de la section "Création de fichiers de configuration IP Filter" à la page 61.

2. **Déterminez si le service IP Filter est activé.**

   - Sur un système en réseau manuel, IP Filter n'est pas activé par défaut.

```
$ svcs -x ipfilter:default
svc:/network/ipfilter:default (IP Filter)
 State: disabled since Mon Sep 10 10:10:50 2012
Reason: Disabled by an administrator.
   See: http://oracle.com/msg/SMF-8000-05
   See: ipfilter(5)
Impact: This service is not running.
```

   - Sur un système en réseau automatique sur un réseau IPv4, exécutez la commande suivante pour afficher la stratégie IP Filter :

```
# ipfstat -io
```

     - Pour afficher le fichier qui a créé la stratégie, lisez `/etc/nwam/loc/NoNet/ipf.conf`. Ce fichier est en lecture seule.
     - Pour modifier la stratégie, reportez-vous à la section "Création de fichiers de configuration IP Filter" à la page 61.

Remarque - Pour afficher la stratégie IP Filter sur un réseau IPv6, ajoutez l'option -6, comme dans : `ipfstat -6io`. Pour plus d'informations, reportez-vous à la page de manuel `ipfstat`(1M).

Création de fichiers de configuration IP Filter

Pour modifier la stratégie IP Filter pour une configuration réseau configurée de manière automatique ou pour utiliser IP Filter dans un réseau configuré manuellement, créez des fichiers de configuration, informez le service de l'existence de ces fichiers, puis activez le service.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Spécifiez l'emplacement du fichier de stratégie pour le service IP Filter.**

   Ce fichier contient l'ensemble des règles de filtrage de paquets.

   a. **Définissez tout d'abord le fichier de stratégie sur `custom`.**

   ```
   # svccfg -s ipfilter:default setprop firewall_config_default/policy = astring:
    "custom"
   ```

   b. **Indiquez ensuite l'emplacement.**

   Par exemple, placez votre ensemble de règles de filtrage de paquets à l'emplacement `/etc/ipf/myorg.ipf.conf`.

   ```
   # svccfg -s ipfilter:default \
   setprop firewall_config_default/custom_policy_file = astring: "/etc/ipf/
   myorg.ipf.conf"
   ```

2. **Créez votre ensemble de règles de filtrage de paquets.**

   Pour plus d'informations sur le filtrage de paquets, reportez-vous à la section "Utilisation de la fonctionnalité de filtrage de paquets d'IP Filter" à la page 50. Pour consulter des exemples de fichiers de configuration, reportez-vous à la section "Exemples de fichiers de configuration IP Filter" à la page 85 et au fichier `/etc/nwam/loc/NoNet/ipf.conf`.

Remarque - Si le fichier de stratégie spécifié est vide, aucun filtrage n'est effectué. Un fichier de filtrage de paquets vide correspond à un ensemble de règles défini comme suit :

```
pass in all
pass out all
```

3. **(Facultatif) Créez un fichier de configuration de translation d'adresse réseau (NAT, Network Address Translation) pour IP Filter.**

   Pour filtrer des paquets par le biais d'une NAT, créez un fichier destiné à contenir les règles NAT avec le nom par déaut, `/etc/ipf/ipnat.conf`. Si vous utilisez un nom différent, il faut remplacer la valeur de la propriété de service `config/ipnat_config_file`, comme illustré ci-dessous :

   ```
   # svccfg -s ipfilter:default \
   setprop config/ipnat_config_file = astring: "/etc/ipf/myorg.ipnat.conf"
   ```

   Pour plus d'informations sur NAT, reportez-vous à la section “Utilisation de la fonctionnalité NAT d'IP Filter” à la page 53.

4. **(Facultatif) Créez un fichier de configuration de pool d'adresses.**

   Pour pouvoir faire référence à un groupe d'adresses par le biais d'un pool d'adresses unique, créez un fichier contenant le pool avec le nom par défaut, `/etc/ipf/ippool.conf`. Si vous utilisez un nom différent, il faut remplacer la valeur de la propriété de service `config/ippool_config_file`, comme illustré ci-dessous :

   ```
   # svccfg -s ipfilter:default \
   setprop config/ippool_config_file = astring: "/etc/ipf/myorg.ippool.conf"
   ```

   Un pool d'adresses peut contenir un ensemble quelconque d'adresses IPv4 et IPv6. Pour plus d'informations sur les pools d'adresses, reportez-vous à la section “Utilisation de la fonctionnalité de pools d'adresses d'IP Filter” à la page 55.

5. **(Facultatif) Activez le filtrage du trafic en loopback.**

   Pour filtrer le trafic entre les zones configurées sur le système, le cas échéant, activez le filtrage de loopback. Reportez-vous à la section “Activation du filtrage de loopback” à la page 64. Vous devez également définir des ensembles de règles applicables aux zones.

6. **(Facultatif) Désactivez le réassemblage des paquets fragmentés.**

   Par défaut, les fragments sont réassemblés dans IP Filter. Pour modifier cette valeur par défaut, reportez-vous à la section “Désactivation du réassemblage des paquets” à la page 63.

Activation et actualisation d'IP Filter

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

Vous avez effectué les étapes de la section "Création de fichiers de configuration IP Filter" à la page 61.

1. **Activez IP Filter.**
   Pour activer IP Filter pour la première fois, saisissez la commande suivante :

   ```
   # svcadm enable network/ipfilter
   ```

2. **Si vous modifiez des fichiers de configuration d'IP Filter alors que le service est en cours d'exécution, actualisez le service.**

   ```
   # svcadm refresh network/ipfilter
   ```

   Remarque - La commande `refresh` désactive brièvement le pare-feu. Pour ne pas désactiver le pare-feu, ajoutez des règles ou un nouveau fichier de configuration. Pour consulter des procédures illustrées à l'aide d'exemples, reportez-vous à la section "Utilisation des ensembles de règles IP Filter" à la page 66.

Désactivation du réassemblage des paquets

Par défaut, les fragments sont réassemblés dans IP Filter. Pour désactiver le réassemblage, insérez une règle au début du fichier de stratégie.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter) ainsi que de l'autorisation `solaris.admin.edit/`*path-to-IPFilter-policy-file*. Le rôle `root` dispose de tous ces droits. Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Désactivez IP Filter.**

   ```
   # svcadm disable network/ipfilter
   ```

2. **Ajoutez la règle suivante au début du fichier de stratégie IP Filter.**

   ```
   set defrag off;
   ```

   Utilisez la commande `pfedit`, comme illustré ci-dessous :

```
# pfedit /etc/ipf/myorg.ipf.conf
```

Cette règle doit précéder toutes les règles `block` et `pass` du fichier. Toutefois, vous pouvez insérer des commentaires avant la ligne, comme dans l'exemple ci-dessous :

```
# Disable fragment reassembly
#
set defrag off;
# Define policy
#
block in all
block out all
other rules
```

3. **Activez IP Filter.**

```
# svcadm enable network/ipfilter
```

4. **Assurez-vous que les paquets ne sont pas réassemblés.**

```
# ipf -T defrag
defrag  min 0   max 0x1 current 0
```

Si la valeur associée à `current` est `0`, les fragments ne sont pas réassemblés. Si la valeur associée à `current` est `1`, les fragments sont réassemblés.

▼ Activation du filtrage de loopback

Avant de commencer

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter) ainsi que de l'autorisation `solaris.admin.edit/`*path-to-IPFilter-policy-file*. Le rôle `root` dispose de tous ces droits. Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

1. **Le cas échéant, arrêtez IP Filter.**

```
# svcadm disable network/ipfilter
```

2. **Ajoutez la règle suivante au début du fichier de stratégie IP Filter.**

```
set intercept_loopback true;
```

Utilisez la commande `pfedit`, comme illustré ci-dessous :

```
# pfedit /etc/ipf/myorg.ipf.conf
```

Cette ligne doit précéder toutes les règles `block` et `pass` définies dans le fichier. Toutefois, vous pouvez insérer des commentaires avant la ligne, comme dans l'exemple ci-dessous :

```
...
#set defrag off;
#
# Enable loopback filtering to filter between zones
#
set intercept_loopback true;
#
# Define policy
#
block in all
block out all
other rules
```

3. **Activez IP Filter.**

```
# svcadm enable network/ipfilter
```

4. **Pour vérifier le statut du filtrage de loopback, exécutez la commande ci-dessous :**

```
# ipf -T ipf_loopback
ipf_loopback    min 0   max 0x1 current 1
#
```

Si la valeur associée à `current` est `0`, le filtrage du loopback est désactivé. Si la valeur associée à `current` est `1`, le filtrage de loopback est activé.

▼ Désactivation du filtrage de paquets

Cette procédure supprime toutes les règles à partir du noyau et désactive le service. Si vous avez recours à cette procédure, vous devez activer IP Filter avec les fichiers de configuration appropriés pour redémarrer le filtrage de paquets et la translation NAT. Pour plus d'informations, reportez-vous à la section “Activation et actualisation d'IP Filter” à la page 63.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

- **Pour désactiver le service, utilisez la commande `svcadm`.**

```
# svcadm disable network/ipfilter
```

Pour tester ou déboguer le service, vous pouvez supprimer des ensembles de règles pendant que le service est en cours d'exécution. Pour plus d'informations, reportez-vous à la section “Utilisation des ensembles de règles IP Filter” à la page 66.

Utilisation des ensembles de règles IP Filter

Vous pouvez modifier ou désactiver le filtrage des paquets et les règles NAT dans les conditions suivantes :

- En vue de réaliser des tests
- En vue de résoudre des problèmes système qui semblent être causés par IP Filter

La liste des tâches ci-dessous répertorie les procédures associées aux ensembles de règles IP Filter.

TABLEAU 5-2 Liste de tâches pour l'utilisation des ensembles de règles IP Filter

Tâche	Pour obtenir des instructions
Affichez l'ensemble actif de règles de filtrage de paquets.	“Affichage de l'ensemble actif de règles de filtrage de paquets” à la page 67
Affichez un ensemble de règles inactif pour le filtrage de paquets.	“Affichage de l'ensemble inactif de règles de filtrage de paquets” à la page 67
Activez un nouvel ensemble de règles actif.	“Activation d'un nouvel ensemble de règles de filtrage de paquets ou d'un ensemble mis à jour” à la page 67
Supprimez un ensemble de règles.	“Suppression d'un ensemble de règles de filtrage de paquets” à la page 69
Ajoutez des règles aux ensembles de règles.	“Ajout de règles à l'ensemble actif de règles de filtrage de paquets” à la page 69 “Ajout de règles à l'ensemble inactif de règles de filtrage de paquets” à la page 71
Basculez entre les ensembles de règles actif et inactif.	“Basculement entre les ensembles actif et inactif de règles de filtrage de paquets” à la page 71
Supprimez du noyau un ensemble de règles inactif.	“Suppression d'un ensemble inactif de règles de filtrage de paquets du noyau” à la page 73
Affichez les règles NAT actives.	“Affichage des règles NAT actives dans IP Filter” à la page 73
Suppression des règles NAT.	“Désactivation des règles NAT dans IP Filter” à la page 74
Ajout de règles à des règles NAT actives.	“Ajout de règles aux règles de filtrage de paquets NAT” à la page 74
Affichez les pools d'adresses actifs.	“Affichage des pools d'adresses actifs” à la page 76
Supprimez un pool d'adresses.	“Suppression d'un pool d'adresses” à la page 76
Ajout de règles à un pool d'adresses.	“Ajout de règles à un pool d'adresses” à la page 76

Gestion des ensembles de règles de filtrage de paquets d'IP Filter

IP Filter autorise l'hébergement dans le noyau d'un ensemble de règles de filtrage de paquets actif et d'un ensemble de règles inactif. L'ensemble de règles actif détermine le filtrage appliqué aux paquets entrants et aux paquets sortants. L'ensemble de règles inactif contient également des règles. Ces règles ne sont pas appliquées, sauf si vous définissez l'ensemble de règles inactif

comme l'ensemble de règles actif. Vous pouvez gérer, afficher et modifier les ensembles actif et inactif de règles de filtrage de paquets.

Remarque - Les procédures suivantes fournissent des exemples pour des réseaux IPv4. Pour les paquets IPv6, utilisez l'option `-6` tel que décrit à l'Étape 2 de la section “Affichage des valeurs par défaut du service IP Filter” à la page 60.

▼ Affichage de l'ensemble actif de règles de filtrage de paquets

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

- **Affichez l'ensemble actif de règles de filtrage de paquets.**
  L'exemple ci-dessous présente la sortie de l'ensemble actif de règles de filtrage de paquets chargé dans le noyau.

```
# ipfstat -io
empty list for ipfilter(out)
pass in quick on net1 from 192.168.1.0/24 to any
pass in all
block in on net1 from 192.168.1.10/32 to any
```

▼ Affichage de l'ensemble inactif de règles de filtrage de paquets

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

- **Affichez l'ensemble inactif de règles de filtrage de paquets.**
  L'exemple ci-dessous présente la sortie d'un ensemble inactif de règles de filtrage de paquets.

```
# ipfstat -I -io
pass out quick on net1 all
pass in quick on net1 all
```

▼ Activation d'un nouvel ensemble de règles de filtrage de paquets ou d'un ensemble mis à jour

Effectuez la procédure ci-dessous pour exécuter l'une ou l'autre des tâches suivantes :

- activation d'un ensemble de règles de filtrage de paquets différent de celui que IP Filter utilise actuellement ;
- Rechargement du même ensemble de règles de filtrage mis à jour.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Procédez de l'une des façons suivantes :**

   - Si vous souhaitez activer un ensemble de règles complètement différent, créez un nouvel ensemble de règles dans un fichier distinct.
   - Mettez à jour l'ensemble de règles actuel dans votre fichier de configuration.

2. **Supprimez l'ensemble de règles actuel et chargez le nouvel ensemble de règles.**

   ```
   # ipf -Fa -f filename
   ```

   Les règles présentes dans *filename* remplacent l'ensemble de règles actif.

Remarque - N'utilisez pas de commandes telles que `ipf -D` ou `svcadm restart` pour charger l'ensemble de règles mis à jour. Ces commandes affectent la sécurité du réseau, car elles désactivent le pare-feu avant de charger le nouvel ensemble de règles.

Exemple 5-1 Activation d'un nouvel ensemble de règles de filtrage de paquets

L'exemple suivant illustre le remplacement d'un ensemble de règles de filtrage de paquets par un autre ensemble de règles.

```
# ipfstat -io
empty list for ipfilter(out)
pass in quick on net0 all
# ipf -Fa -f /etc/ipf/ipfnew.conf
# ipfstat -io
empty list for ipfilter(out)
block in log quick from 10.0.0.0/8 to any
```

Exemple 5-2 Rechargement d'un ensemble de règles de filtrage de paquets mis à jour

Dans l'exemple ci-dessous, un ensemble de règles de filtrage de paquets actuellement actif est rechargé suite à sa mise à jour.

Liste de l'environnement d'initialisation actif de règles dans l'état (facultatif).

```
# ipfstat -io
empty list for ipfilter (out)
```

```
block in log quick from 10.0.0.0/8 to any
```

Ensuite, modifiez le fichier de configuration `/etc/ipf/myorg.ipf.conf`, actualisez le service et répertoriez à nouveau l'ensemble de règles actif.

```
# svcadm refresh network/ipfilter
# ipfstat -io
empty list for ipfilter (out)
block in log quick from 10.0.0.0/8 to any
block in quick on net1 from 192.168.0.0/12 to any
```

▼ Suppression d'un ensemble de règles de filtrage de paquets

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

- **Supprimez l'ensemble de règles.**

```
# ipf -F [a|i|o]
```

`-a`	Supprime toutes les règles de filtrage de l'ensemble de règles.
`-i`	Supprime les règles de filtrage pour les paquets entrants.
`-o`	Supprime les règles de filtrage pour les paquets sortants.

Exemple 5-3 Suppression d'un ensemble de règles de filtrage de paquets

Dans l'exemple ci-dessous, toutes les règles de filtrage sont supprimées de l'ensemble de règles de filtrage actif.

```
# ipfstat -io
block out log on net0 all
block in log quick from 10.0.0.0/8 to any
# ipf -Fa
# ipfstat -io
empty list for ipfilter(out)
empty list for ipfilter(in)
```

▼ Ajout de règles à l'ensemble actif de règles de filtrage de paquets

L'ajout de règles à un ensemble de règles existant peut être utile lors de la réalisation de tests ou lors du dépannage. Le service IP Filter reste activé lorsque les règles sont ajoutées. Toutefois,

lorsque le service est actualisé, redémarré ou activé, les règles sont perdues, sauf si elles existent dans des fichiers constituant des propriétés du service IP Filter.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

- **Appliquez l'une des méthodes ci-dessous pour ajouter des règles à l'ensemble de règles actif :**

  - Pour ajouter des règles à l'ensemble de règles via la ligne de commande, exécutez la commande `ipf -f -`.

    ```
    # echo "block in on net1 proto tcp from 10.1.1.1/32 to any" | ipf -f -
    ```

    Ces règles ajoutées ne font pas partie de la configuration d'IP Filter lorsque le service est actualisé, redémarré ou activé.
  - Exécutez les commandes ci-dessous :

    1. Créez un ensemble de règles dans le fichier de votre choix.
    2. Ajoutez les règles que vous avez créées à l'ensemble de règles actif.

       ```
       # ipf -f filename
       ```

       Les règles présentes dans le fichier *filename* sont ajoutées à la fin de l'ensemble de règles actif. IP Filter utilise un algorithme de type "dernière règle correspondante", de sorte que les nouvelles règles déterminent les priorités de filtrage, sauf si l'utilisateur ajoute le mot-clé `quick`. Si le paquet correspond à une règle contenant le mot-clé `quick`, l'action associée à cette règle est exécutée et les règles suivantes sont ignorées.

  Si *filename* est la valeur de l'une des propriétés de fichiers de configuration IP Filter, les règles sont rechargées lorsque le service est activé, redémarré ou actualisé. Dans le cas contraire, les règles ajoutées forment un ensemble de règles temporaire.

Exemple 5-4 Ajout de règles à l'ensemble actif de règles de filtrage de paquets

Dans l'exemple ci-dessous, une règle est ajoutée à l'ensemble actif de règles de filtrage de paquets à partir de la ligne de commande.

```
# ipfstat -io
empty list for ipfilter(out)
block in log quick from 10.0.0.0/8 to any
# echo "block in on net1 proto tcp from 10.1.1.1/32 to any" | ipf -f -
# ipfstat -io
empty list for ipfilter(out)
block in log quick from 10.0.0.0/8 to any
block in on net1 proto tcp from 10.1.1.1/32 to any
```

▼ Ajout de règles à l'ensemble inactif de règles de filtrage de paquets

La création d'un ensemble de règles inactif dans le noyau peut être utile pour la réalisation de tests ou le dépannage. Cet ensemble de règles peut être substitué à l'ensemble de règles actif sans qu'il soit nécessaire d'arrêter le service IP Filter. Toutefois, lorsque le service est actualisé, redémarré ou activé, l'ensemble de règles inactif doit être ajouté.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

1. **Créez un ensemble de règles dans le fichier de votre choix.**

2. **Ajoutez les règles que vous avez créées à l'ensemble de règles inactif.**

   `# ipf -I -f` *filename*

   Les règles présentes dans le fichier *filename* sont ajoutées à la fin de l'ensemble de règles inactif. IP Filter utilise un algorithme de type "dernière règle correspondante", de sorte que les nouvelles règles déterminent les priorités de filtrage, sauf si l'utilisateur ajoute le mot-clé `quick`. Si le paquet correspond à une règle contenant le mot-clé `quick`, l'action associée à cette règle est exécutée et les règles suivantes sont ignorées.

Exemple 5-5 Ajout de règles à l'ensemble de règles inactif

Dans l'exemple ci-dessous, une règle est ajoutée à l'ensemble de règles inactif à partir d'un fichier.

```
# ipfstat -I -io
pass out quick on net1 all
pass in quick on net1 all
# ipf -I -f /etc/ipf/ipftrial.conf
# ipfstat -I -io
pass out quick on net1 all
pass in quick on net1 all
block in log quick from 10.0.0.0/8 to any
```

▼ Basculement entre les ensembles actif et inactif de règles de filtrage de paquets

Le passage à un autre ensemble de règles dans le noyau peut être utile lors de la réalisation de tests ou du dépannage. L'ensemble de règles peut être rendu actif sans qu'il soit nécessaire d'arrêter le service IP Filter.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

- **Basculez entre les ensembles de règles actif et inactif.**

```
# ipf -s
```

Cette commande permet de basculer entre les ensembles de règles actif et inactif dans le noyau. Si l'ensemble de règles inactif est vide, aucun filtrage de paquets n'est effectué.

Remarque - Lorsque le service IP Filter est actualisé, redémarré ou activé, les règles enregistrées dans des fichiers constituant des propriétés du service IP Filter sont restaurées. L'ensemble de règles inactif n'est pas restauré.

Exemple 5-6 Basculement entre les ensembles actif et inactif de règles de filtrage de paquets

L'exemple ci-dessous illustre l'utilisation de la commande `ipf -s`, laquelle a pour effet de rendre actif l'ensemble de règles inactif et de rendre inactif l'ensemble de règles actif.

- Avant l'exécution de la commande `ipf -s`, la sortie de la commande `ipfstat -I -io` met en évidence les règles de l'ensemble de règles inactif. La sortie de la commande `ipfstat -io` affiche les règles dans l'ensemble de règles actif.

```
# ipfstat -io
empty list for ipfilter(out)
block in log quick from 10.0.0.0/8 to any
block in on net1 proto tcp from 10.1.1.1/32 to any
# ipfstat -I -io
pass out quick on net1 all
pass in quick on net1 all
block in log quick from 10.0.0.0/8 to any
```

- Après l'exécution de la commande `ipf -s`, les sorties des commandes `ipfstat -I -io` et `ipfstat -io` indiquent que le contenu des deux ensembles de règles a été échangé.

```
# ipf -s
Set 1 now inactive
# ipfstat -io
pass out quick on net1 all
pass in quick on net1 all
block in log quick from 10.0.0.0/8 to any
# ipfstat -I -io
empty list for inactive ipfilter(out)
block in log quick from 10.0.0.0/8 to any
block in on net1 proto tcp from 10.1.1.1/32 to any
```

▼ Suppression d'un ensemble inactif de règles de filtrage de paquets du noyau

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

- **Spécifiez l'ensemble de règles inactif via la commande "flush all".**

```
# ipf -I -Fa
```

Remarque - Si vous exécutez ensuite la commande `ipf -s`, l'ensemble de règles inactif vide devient l'ensemble de règles actif. Si l'ensemble de règles actif est vide, *aucun* filtrage n'est effectué.

Exemple 5-7 Suppression d'un ensemble inactif de règles de filtrage de paquets du noyau

Dans l'exemple ci-dessous, l'ensemble inactif de règles de filtrage de paquets est vidé afin de supprimer toutes les règles.

```
# ipfstat -I -io
empty list for inactive ipfilter(out)
block in log quick from 10.0.0.0/8 to any
block in on net1 proto tcp from 10.1.1.1/32 to any
# ipf -I -Fa
# ipfstat -I -io
empty list for inactive ipfilter(out)
empty list for inactive ipfilter(in)
```

Gestion des règles NAT d'IP Filter

Les procédures suivantes gèrent, affichent et modifient les règles NAT pour IP Filter.

▼ Affichage des règles NAT actives dans IP Filter

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

- **Affichez les règles NAT actives.**

L'exemple ci-dessous présente la sortie de l'ensemble de règles NAT actif.

```
# ipnat -l
List of active MAP/Redirect filters:
map net0 192.168.1.0/24 -> 20.20.20.1/32

List of active sessions:
```

▼ Désactivation des règles NAT dans IP Filter

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

- **Supprimez les règles NAT du noyau.**

```
# ipnat -FC
```

L'option `-C` permet de supprimer toutes les entrées de la liste de règles NAT actuelle. L'option `-F` permet de supprimer toutes les entrées actives de la table de translation NAT qui indique les mappages NAT actifs.

Exemple 5-8 Suppression des règles NAT

Dans l'exemple ci-dessous, les entrées des règles NAT actuelles sont supprimées.

```
# ipnat -l
List of active MAP/Redirect filters:
map net0 192.168.1.0/24 -> 20.20.20.1/32

List of active sessions:
# ipnat -C
1 entries flushed from NAT list
# ipnat -l
List of active MAP/Redirect filters:

List of active sessions:
```

▼ Ajout de règles aux règles de filtrage de paquets NAT

L'ajout de règles à un ensemble de règles existant peut être utile lors de la réalisation de tests ou lors du dépannage. Le service IP Filter reste activé lorsque les règles sont ajoutées. Toutefois, lorsque le service est actualisé, redémarré ou activé, les règles NAT sont perdues, sauf si elles existent dans un fichier constituant une propriété du service IP Filter.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

- **Appliquez l'une des méthodes ci-dessous pour ajouter des règles à l'ensemble de règles actif :**

  - Pour ajouter des règles à l'ensemble de règles NAT via la ligne de commande, exécutez la commande `ipnat -f -`.

    ```
    # echo "map net0 192.168.1.0/24 -> 20.20.20.1/32" | ipnat -f -
    ```

    Ces règles ajoutées ne font pas partie de la configuration d'IP Filter lorsque le service est actualisé, redémarré ou activé.
  - Exécutez les commandes ci-dessous :

    1. Créez des règles NAT supplémentaires dans le fichier de votre choix.
    2. Ajoutez les règles que vous avez créées aux règles NAT actives.

       ```
       # ipnat -f filename
       ```

       Les règles présentes dans le fichier *filename* sont ajoutées à la fin des règles NAT.

       Si *filename* est la valeur de l'une des propriétés de fichiers de configuration IP Filter, les règles sont rechargées lorsque le service est activé, redémarré ou actualisé. Dans le cas contraire, les règles ajoutées forment un ensemble de règles temporaire.

Exemple 5-9 Ajout de règles à l'ensemble de règles NAT

Dans l'exemple ci-dessous, une règle est ajoutée à l'ensemble de règles NAT via la ligne de commande.

```
# ipnat -l
List of active MAP/Redirect filters:

List of active sessions:
# echo "map net0 192.168.1.0/24 -> 20.20.20.1/32" | ipnat -f -
# ipnat -l
List of active MAP/Redirect filters:
map net0 192.168.1.0/24 -> 20.20.20.1/32

List of active sessions:
```

Gestion des pools d'adresses d'IP Filter

Les procédures ci-dessous pour gèrent, affichent et modifient les pools d'adresses.

▼ Affichage des pools d'adresses actifs

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

- **Affichez le pool d'adresses actif.**

  Dans l'exemple ci-dessous, le contenu du pool d'adresses actif est affiché.

```
# ippool -l
table role = ipf type = tree number = 13
        { 10.1.1.1/32, 10.1.1.2/32, 192.168.1.0/24; };
```

▼ Suppression d'un pool d'adresses

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

- **Supprimez les entrées du pool d'adresses actuel.**

```
# ippool -F
```

Exemple 5-10 Suppression d'un pool d'adresses

Dans l'exemple ci-dessous, un pool d'adresses est supprimé.

```
# ippool -l
table role = ipf type = tree number = 13
        { 10.1.1.1/32, 10.1.1.2/32, 192.168.1.0/24; };
# ippool -F
1 object flushed
# ippool -l
```

▼ Ajout de règles à un pool d'adresses

L'ajout de règles à un ensemble de règles existant peut être utile lors de la réalisation de tests ou lors du dépannage. Le service IP Filter reste activé lorsque les règles sont ajoutées. Toutefois, lorsque le service est actualisé, redémarré ou activé, les règles de pools d'adresses sont perdues, sauf si elles existent dans un fichier constituant une propriété du service IP Filter.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide

de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

1. **Appliquez l'une des méthodes ci-dessous pour ajouter des règles à l'ensemble de règles actif :**

   - Pour ajouter des règles à l'ensemble de règles via la ligne de commande, exécutez la commande `ippool -f -`.

     ```
     # echo "table role = ipf type = tree number = 13
     {10.1.1.1/32, 10.1.1.2/32, 192.168.1.0/24};" | ippool -f -
     ```

     Ces règles ajoutées ne font pas partie de la configuration d'IP Filter lorsque le service est actualisé, redémarré ou activé.
   - Exécutez les commandes ci-dessous :

     1. Créez des pools d'adresses supplémentaires dans le fichier de votre choix.
     2. Ajoutez les règles que vous avez créées au pool d'adresses actif.

        ```
        # ippool -f filename
        ```

        Les règles présentes dans le fichier *filename* sont ajoutées à la fin du pool d'adresses actif.

2. **Si les règles contiennent des pools qui ne figurent pas dans l'ensemble de règles d'origine, procédez comme suit :**

   a. **Ajoutez les pools à une nouvelle règle de filtrage de paquets.**

   b. **Ajoutez la nouvelle règle de filtrage de paquets à l'ensemble de règles actuel.**

      Suivez les instructions de la section “Ajout de règles à l'ensemble actif de règles de filtrage de paquets” à la page 69.

Remarque - N'actualisez ou ne redémarrez pas le service IP Filter. Vous risquez de perdre vos règles de pools d'adresses ajoutées.

Exemple 5-11 Ajout de règles à un pool d'adresses

Dans l'exemple ci-dessous, un pool d'adresses est ajouté à l'ensemble de règles de pool d'adresses via la ligne de commande.

```
# ippool -l
table role = ipf type = tree number = 13
        { 10.1.1.1/32, 10.1.1.2/32, 192.168.1.0/24; };
# echo "table role = ipf type = tree number = 100
 {10.0.0.0/32, 172.16.1.2/32, 192.168.1.0/24};" | ippool -f -
```

```
# ippool -l
table role = ipf type = tree number = 100
        { 10.0.0.0/32, 172.16.1.2/32, 192.168.1.0/24; };
table role = ipf type = tree number = 13
        { 10.1.1.1/32, 10.1.1.2/32, 192.168.1.0/24; };
```

Affichage des statistiques et des informations relatives à IP Filter

TABLEAU 5-3 Liste des tâches pour l'affichage des statistiques et informations IP Filter

Tâche	Pour obtenir des instructions
Affichage des tables d'état.	“Affichage des tables d'état d'IP Filter” à la page 78
Affichage des statistiques sur l'état des paquets.	“Affichage des statistiques d'état d'IP Filter” à la page 79
Liste des paramètres réglables IP Filter.	“Affichage des paramètres réglables IP Filter” à la page 80
Affichez les statistiques NAT.	“Affichage des statistiques NAT d'IP Filter” à la page 80
Affichez les statistiques de pool d'adresses.	“Affichage des statistiques de pool d'adresses d'IP Filter” à la page 81

▼ Affichage des tables d'état d'IP Filter

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

- **Affichez la table d'état.**

```
# ipfstat
```

Remarque - L'option `-t` permet d'afficher la table d'état au format de l'utilitaire `top` UNIX.

Exemple 5-12 Affichage des tables d'état d'IP Filter

L'exemple suivant illustre la sortie de la table d'état.

```
# ipfstat
bad packets:            in 0    out 0
 IPv6 packets:          in 56286 out 63298
 input packets:         blocked 160 passed 11 nomatch 1 counted 0 short 0
output packets:         blocked 0 passed 13681 nomatch 6844 counted 0 short 0
```

```
 input packets logged:  blocked 0 passed 0
output packets logged:  blocked 0 passed 0
 packets logged:        input 0 output 0
 log failures:          input 0 output 0
fragment state(in):     kept 0  lost 0  not fragmented 0
fragment reassembly(in):bad v6 hdr 0     bad v6 ehdr 0  failed reassembly 0
fragment state(out):    kept 0  lost 0  not fragmented 0
packet state(in):       kept 0  lost 0
packet state(out):      kept 0  lost 0
ICMP replies:   0       TCP RSTs sent:  0
Invalid source(in):     0
Result cache hits(in):  152     (out):  6837
IN Pullups succeeded:   0       failed: 0
OUT Pullups succeeded:  0       failed: 0
Fastroute successes:    0       failures:       0
TCP cksum fails(in):    0       (out):  0
IPF Ticks:      14341469
Packet log flags set: (0)
        none
```

▼ Affichage des statistiques d'état d'IP Filter

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

- **Affichez les statistiques d'état.**

```
# ipfstat -s
```

Exemple 5-13 Affichage des statistiques d'état d'IP Filter

L'exemple suivant illustre la sortie des statistiques d'état.

```
# ipfstat -s
IP states added:
        0 TCP
        0 UDP
        0 ICMP
        0 hits
        0 misses
        0 maximum
        0 no memory
        0 max bucket
        0 active
        0 expired
        0 closed
State logging enabled
```

```
State table bucket statistics:
        0 in use
        0.00% bucket usage
        0 minimal length
        0 maximal length
        0.000 average length
```

▼ Affichage des paramètres réglables IP Filter

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

- **Affichez les paramètres réglables du noyau pour IP Filter.**
  La sortie suivante est tronquée.

```
# ipf -T list
fr_flags min 0 max 0xffffffff current 0
fr_active min 0 max 0 current 0
...
ipstate_logging min 0 max 0x1 current 1
...
fr_authq_ttl min 0x1 max 0x7fffffff current sz = 0
fr_enable_rcache min 0 max 0x1 current 0
```

▼ Affichage des statistiques NAT d'IP Filter

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

- **Affichez les statistiques NAT.**

```
# ipnat -s
```

Exemple 5-14 Affichage des statistiques NAT d'IP Filter

L'exemple ci-dessous illustre les statistiques NAT.

```
# ipnat -s
mapped  in      0       out     0
added   0       expired 0
no memory       0       bad nat 0
inuse   0
```

```
rules   1
wilds   0
```

▼ Affichage des statistiques de pool d'adresses d'IP Filter

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

- **Affichez les statistiques de pool d'adresses.**

```
# ippool -s
```

Exemple 5-15 Affichage des statistiques de pool d'adresses d'IP Filter

L'exemple suivant illustre les statistiques de pool d'adresses.

```
# ippool -s
Pools:  3
Hash Tables:    0
Nodes:  0
```

Utilisation des fichiers journaux IP Filter

TABLEAU 5-4 Liste des tâches pour l'utilisation des fichiers journaux IP Filter

Tâche	Pour obtenir des instructions
Créez un fichier journal IP Filter distinct.	"Configuration d'un fichier journal d'IP Filter" à la page 81
Affichage des journaux d'état, NAT et normaux.	"Affichage des fichiers journaux IP Filter" à la page 83
Vidage du tampon de journalisation des paquets	"Vidage du tampon du journal de paquets" à la page 84
Les paquets consignés peuvent être enregistrés dans un fichier afin d'être consultés par la suite.	"Enregistrement dans un fichier des paquets consignés" à la page 84

▼ Configuration d'un fichier journal d'IP Filter

Par défaut, toutes les informations de journal IP Filter sont enregistrées par `syslog`. Il est recommandé de créer un fichier journal distinct dans lequel enregistrer les informations de

trafic IP Filter afin de séparer ces données des autres données susceptibles d'être consignées dans le fichier journal `syslog`.

Avant de commencer Vous devez prendre le rôle `root`.

1. **Déterminez quelle instance du service `system-log` est activée.**

```
% svcs system-log
STATE          STIME    FMRI
disabled       13:11:55 svc:/system/system-log:rsyslog
online         13:13:27 svc:/system/system-log:default
```

Remarque - Si l'instance de service `rsyslog` est en ligne, modifiez le fichier `rsyslog.conf`.

2. **Ajoutez les lignes suivantes au fichier `/etc/syslog.conf` :**

```
# Save IP Filter log output to its own file
local0.debug             /var/log/log-name
```

Remarque - Dans votre entrée, utilisez la touche de tabulation (et non la barre d'espacement) pour séparer `local0.debug` de `/var/log/`*log-name*. Pour plus d'informations, reportez-vous aux pages de manuel `syslog.conf`(4) et `syslogd`(1M).

3. **Créez le fichier journal.**

```
# touch /var/log/log-name
```

4. **Actualisez les informations de configuration du service `system-log`.**

```
# svcadm refresh system-log:default
```

Remarque - Actualisez l'instance du service `system-log:rsyslog` si le service `rsyslog` est activé.

Exemple 5-16 Création d'un journal IP Filter

L'exemple suivant crée le fichier `ipmon.log` pour archiver les informations IP Filter.

Modifiez `syslog.conf`.

```
pfedit /etc/syslog.conf
## Save IP Filter log output to its own file
local0.debug<Tab>/var/log/ipmon.log
```

Puis, dans la ligne de commande, créez le fichier et redémarrez le service.

```
# touch /var/log/ipmon.log
# svcadm restart system-log
```

▼ Affichage des fichiers journaux IP Filter

Avant de commencer

Vous avez effectué les étapes de la section “Configuration d'un fichier journal d'IP Filter” à la page 81.

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

- **Affichez le fichier journal normal, le fichier journal NAT ou le fichier journal d'état.**

  Pour afficher un fichier journal, tapez la commande ci-dessous, conjointement avec l'option adéquate :

  ```
  # ipmon -o [S|N|I] filename
  ```

`S`	Affiche le fichier journal d'état.
`N`	Affiche le fichier journal NAT.
`I`	Affiche le fichier journal IP normal.

  - **Pour afficher le fichier journal normal et les fichiers journaux d'état et NAT, appliquez les options :**

    ```
    # ipmon -o SNI filename
    ```

  - **Après avoir arrêté le démon `ipmon`, vous pouvez utiliser la commande `ipmon` pour afficher les fichiers journaux d'état, NAT et IP Filter :**

    ```
    # pkill ipmon
    # ipmon -a filename
    ```

Remarque - N'utilisez pas la syntaxe `ipmon -a` si le démon `ipmon` est en cours d'exécution. Normalement, le démon démarre automatiquement à l'initialisation du système. L'exécution de la commande `ipmon -a` ouvre une seconde instance d'`ipmon`. Dans ce cas, les deux copies lisent les mêmes informations de journal, mais tout message du journal n'est reçu que par l'une d'elles.

Pour plus d'informations sur l'affichage des fichiers journaux, reportez-vous à la page de manuel `ipmon`(1M).

Exemple 5-17 Affichage des fichiers journaux IP Filter

L'exemple ci-dessous présente la sortie du fichier `/var/ipmon.log`.

```
# ipmon -o SNI /var/ipmon.log
02/09/2012 15:27:20.606626 net0 @0:1 p 129.146.157.149 ->
129.146.157.145 PR icmp len 20 84 icmp echo/0 IN
```

ou

```
# pkill ipmon
# ipmon -aD /var/ipmon.log
02/09/2012 15:27:20.606626 net0 @0:1 p 129.146.157.149 ->
129.146.157.145 PR icmp len 20 84 icmp echo/0 IN
```

Vidage du tampon du journal de paquets

Cette procédure efface le tampon et affiche la sortie à l'écran.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits IP Filter Management (gestion IP Filter). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

- **Vidage du tampon de journalisation des paquets**

  ```
  # ipmon -F
  ```

Exemple 5-18 Vidage du tampon du journal de paquets

L'exemple ci-dessous présente la sortie obtenue en cas de suppression d'un fichier journal. Le système génère un rapport même si le fichier journal est vide, comme dans cet exemple.

```
# ipmon -F
0 bytes flushed from log buffer
0 bytes flushed from log buffer
0 bytes flushed from log buffer
```

Enregistrement dans un fichier des paquets consignés

Vous pouvez enregistrer des paquets dans un fichier lors du dépannage ou si vous souhaitez effectuer un audit manuel du trafic.

Avant de commencer Vous devez prendre le rôle `root`.

- **Enregistrez dans un fichier les paquets consignés.**

  ```
  # cat /dev/ipl > filename
  ```

Continuez la journalisation des paquets dans le fichier *filename* et interrompez la procédure en tapant `Ctrl-C` pour afficher de nouveau l'invite de ligne de commande.

Exemple 5-19 Enregistrement dans un fichier des paquets consignés

L'exemple ci-dessous présente les résultats obtenus lorsque les paquets consignés sont enregistrés dans un fichier.

```
# cat /dev/ipl > /tmp/logfile
^C#

# ipmon -f /tmp/logfile
02/09/2012 15:30:28.708294 net0 @0:1 p 129.146.157.149,33923 ->
  129.146.157.145,23 PR tcp len 20 52 -S IN
02/09/2012 15:30:28.708708 net0 @0:1 p 129.146.157.149,33923 ->
  129.146.157.145,23 PR tcp len 20 40 -A IN
02/09/2012 15:30:28.792611 net0 @0:1 p 129.146.157.149,33923 ->
  129.146.157.145,23 PR tcp len 20 70 -AP IN
02/09/2012 15:30:28.872000 net0 @0:1 p 129.146.157.149,33923 ->
 129.146.157.145,23 PR tcp len 20 40 -A IN
02/09/2012 15:30:28.872142 net0 @0:1 p 129.146.157.149,33923 ->
  129.146.157.145,23 PR tcp len 20 43 -AP IN
02/09/2012 15:30:28.872808 net0 @0:1 p 129.146.157.149,33923 ->
  129.146.157.145,23 PR tcp len 20 40 -A IN
02/09/2012 15:30:28.872951 net0 @0:1 p 129.146.157.149,33923 ->
  129.146.157.145,23 PR tcp len 20 47 -AP IN
02/09/2012 15:30:28.926792 net0 @0:1 p 129.146.157.149,33923 ->
  129.146.157.145,23 PR tcp len 20 40 -A IN
.
.
(output truncated)
```

Exemples de fichiers de configuration IP Filter

Les exemples suivants illustrent les règles de filtrage de paquets qui s'appliquent à un hôte unique, un serveur et un routeur.

Les fichiers de configuration suivent les règles de syntaxe UNIX standard :

- Le signe dièse (#) indique qu'une ligne contient des commentaires.
- Une ligne peut contenir à la fois des commentaires et des règles.
- Vous pouvez ajouter des espaces supplémentaires afin de faciliter la lecture des règles.
- La définition d'une règle peut s'étaler sur plusieurs lignes. Un backslash (\) à la fin d'une ligne indique que la règle continue sur la ligne suivante.

Pour plus d'informations sur la syntaxe, reportez-vous à la section “Configuration des règles de filtrage de paquets” à la page 51.

EXEMPLE 5-20 Configuration d'un hôte IP Filter

Cet exemple présente une configuration sur un système hôte avec une interface réseau `net0`.

```
# pass and log everything by default
pass in log on net0 all
pass out log on net0 all

# block, but don't log, incoming packets from other reserved addresses
block in quick on net0 from 10.0.0.0/8 to any
block in quick on net0 from 172.16.0.0/12 to any

# block and log untrusted internal IPs. 0/32 is notation that replaces
# address of the machine running IP Filter.
block in log quick from 192.168.1.15 to <thishost>
block in log quick from 192.168.1.43 to <thishost>

# block and log X11 (port 6000) and remote procedure call
# and portmapper (port 111) attempts
block in log quick on net0 proto tcp from any to net0/32 port = 6000 keep state
block in log quick on net0 proto tcp/udp from any to net0/32 port = 111 keep state
```

Cet ensemble de règles commence par deux règles sans restrictions qui permettent à tous les types de données d'entrer et de sortir via l'interface `net0`. Le deuxième ensemble de règles empêche tout paquet entrant issu des espaces d'adressage privées `10.0.0.0` et `172.16.0.0` de traverser le pare-feu. L'ensemble de règles suivant bloque des adresses internes spécifiques du système hôte. Enfin, le dernier ensemble de règles empêche l'entrée des paquets via les ports 6000 et 111.

EXEMPLE 5-21 Configuration d'un serveur IP Filter

Cet exemple présente la configuration d'un système hôte qui peut être utilisé en tant que serveur Web. Ce système a une interface réseau `net0`.

```
# web server with an net0 interface
# block and log everything by default;
# then allow specific services
# group 100 - inbound rules
# group 200 - outbound rules
# (0/32) resolves to our IP address)
*** FTP proxy ***

# block short packets which are packets
# fragmented too short to be real.
block in log quick all with short

# block and log inbound and outbound by default,
# group by destination
block in log on net0 from any to any head 100
block out log on net0 from any to any head 200
```

```
# web rules that get hit most often
pass in quick on net0 proto tcp from any \
to net0/32 port = http flags S keep state group 100
pass in quick on net0 proto tcp from any \
to net0/32 port = https flags S keep state group 100


# inbound traffic - ssh, auth
pass in quick on net0 proto tcp from any \
to net0/32 port = 22 flags S keep state group 100
pass in log quick on net0 proto tcp from any \
to net0/32 port = 113 flags S keep state group 100
pass in log quick on net0 proto tcp from any port = 113 \
to net0/32 flags S keep state group 100

# outbound traffic - DNS, auth, NTP, ssh, WWW, smtp
pass out quick on net0 proto tcp/udp from net0/32 \
to any port = domain flags S keep state group 200
pass in quick on net0 proto udp from any \
port = domain to net0/32 group 100

pass out quick on net0 proto tcp from net0/32 \
to any port = 113 flags S keep state group 200
pass out quick on net0 proto tcp from net0/32 port = 113 \
to any flags S keep state group 200

pass out quick on net0 proto udp from net0/32 to any \
port = ntp group 200
pass in quick on net0 proto udp from any \
port = ntp to net0/32 port = ntp group 100

pass out quick on net0 proto tcp from net0/32 \
to any port = ssh flags S keep state group 200

pass out quick on net0 proto tcp from net0/32 \
to any port = http flags S keep state group 200
pass out quick on net0 proto tcp from net0/32 \
to any port = https flags S keep state group 200

pass out quick on net0 proto tcp from net0/32 \
to any port = smtp flags S keep state group 200


# pass icmp packets in and out
pass in quick on net0 proto icmp from any to net0/32  keep state group 100
pass out quick on net0 proto icmp from net0/32 to any keep state group 200

# block and ignore NETBIOS packets
block in quick on net0 proto tcp from any \
to any port = 135 flags S keep state group 100

block in quick on net0 proto tcp from any port = 137 \
to any flags S keep state group 100
block in quick on net0 proto udp from any to any port = 137 group 100
```

```
block in quick on net0 proto udp from any port = 137 to any group 100

block in quick on net0 proto tcp from any port = 138 \
to any flags S keep state group 100
block in quick on net0 proto udp from any port = 138 to any group 100

block in quick on net0 proto tcp from any port = 139 to any flags S keep state
group 100
block in quick on net0 proto udp from any port = 139 to any group 100
```

EXEMPLE 5-22 Configuration d'un routeur IP Filter

Cet exemple présente la configuration d'un routeur possédant une interface interne, `net0`, et une interface externe, `net1`.

```
# internal interface is net0 at 192.168.1.1
# external interface is net1 IP obtained via DHCP
# block all packets and allow specific services
*** NAT ***
*** POOLS ***

# Short packets which are fragmented too short to be real.
block in log quick all with short

# By default, block and log everything.
block in log on net0 all
block in log on net1 all
block out log on net0 all
block out log on net1 all


# Packets going in/out of network interfaces that are not on the
# loopback interface should not exist.
block in log quick on net0 from 127.0.0.0/8 to any
block in log quick on net0 from any to 127.0.0.0/8
block in log quick on net1 from 127.0.0.0/8 to any
block in log quick on net1 from any to 127.0.0.0/8


# Deny reserved addresses.
block in quick on net1 from 10.0.0.0/8 to any
block in quick on net1 from 172.16.0.0/12 to any
block in log quick on net1 from 192.168.1.0/24 to any
block in quick on net1 from 192.168.0.0/16 to any


# Allow internal traffic
pass in quick on net0 from 192.168.1.0/24 to 192.168.1.0/24
pass out quick on net0 from 192.168.1.0/24 to 192.168.1.0/24


# Allow outgoing DNS requests from our servers on .1, .2, and .3
pass out quick on net1 proto tcp/udp from net1/32 to any port = domain keep state
```

```
pass in quick on net0 proto tcp/udp from 192.168.1.2 to any port = domain keep state
pass in quick on net0 proto tcp/udp from 192.168.1.3 to any port = domain keep state


# Allow NTP from any internal hosts to any external NTP server.
pass in quick on net0 proto udp from 192.168.1.0/24 to any port = 123 keep state
pass out quick on net1 proto udp from any to any port = 123 keep state


# Allow incoming mail
pass in quick on net1 proto tcp from any to net1/32 port = smtp keep state
pass in quick on net1 proto tcp from any to net1/32 port = smtp keep state
pass out quick on net1 proto tcp from 192.168.1.0/24 to any port = smtp keep state


# Allow outgoing connections: SSH, WWW, NNTP, mail, whois
pass in quick on net0 proto tcp from 192.168.1.0/24 to any port = 22 keep state
pass out quick on net1 proto tcp from 192.168.1.0/24 to any port = 22 keep state

pass in quick on net0 proto tcp from 192.168.1.0/24 to any port = 80 keep state
pass out quick on net1 proto tcp from 192.168.1.0/24 to any port = 80 keep state
pass in quick on net0 proto tcp from 192.168.1.0/24 to any port = 443 keep state
pass out quick on net1 proto tcp from 192.168.1.0/24 to any port = 443 keep state

pass in quick on net0 proto tcp from 192.168.1.0/24 to any port = nntp keep state
block in quick on net1 proto tcp from any to any port = nntp keep state
pass out quick on net1 proto tcp from 192.168.1.0/24 to any port = nntp keep state

pass in quick on net0 proto tcp from 192.168.1.0/24 to any port = smtp keep state

pass in quick on net0 proto tcp from 192.168.1.0/24 to any port = whois keep state
pass out quick on net1 proto tcp from any to any port = whois keep state


# Allow ssh from offsite
pass in quick on net1 proto tcp from any to net1/32 port = 22 keep state


# Allow ping out
pass in quick on net0 proto icmp all keep state
pass out quick on net1 proto icmp all keep state


# allow auth out
pass out quick on net1 proto tcp from net1/32 to any port = 113 keep state
pass out quick on net1 proto tcp from net1/32 port = 113 to any keep state


# return rst for incoming auth
block return-rst in quick on net1 proto tcp from any to any port = 113 flags S/SA


# log and return reset for any TCP packets with S/SA
block return-rst in log on net1 proto tcp from any to any flags S/SA
```

```
# return ICMP error packets for invalid UDP packets
block return-icmp(net-unr) in proto udp all
```

♦ ♦ ♦ CHAPITRE 6

A propos de l'architecture de sécurité IP

L'architecture IPsec (IP security) offre la protection cryptographique des paquets IP sur les réseaux IPv4 et IPv6.

Ce chapitre comprend les sections suivantes :

Pour implémenter IPsec sur votre réseau, reportez-vous au Chapitre 7, Configuration d'IPsec. Pour obtenir des informations de référence, reportez-vous au Chapitre 12, IPsec et référence de gestion des clés

Introduction à IPsec

Le contenu de IPsec Pour protéger les paquets en utilisant les mécanismes de chiffrement et IP fournit des contrôles d'intégrité authentifiant le contenu du paquet. Car IPsec s'exécute au niveau de la couche réseau, une application réseau peut tirer profit d'IPsec sans pour autant avoir à modifier sa configuration. Une utilisation à bon escient d'IPsec en fait un outil efficace de sécurisation du trafic réseau.

IPsec emploie les termes suivants :

- **Protocoles de sécurité –** La protection appliquée à un paquet IP. L'en-tête d'authentification (AH, Authentication Header) protège un paquet IP en y ajoutant un vecteur de contrôle

d'intégrité (ICV, Integrity Check Vector), qui est un hachage du paquet entier y compris les en-têtes IP. Le destinataire est sûr que le paquet n'a pas été modifié. Il ne garantit pas la confidentialité avec chiffrement.

protocole ESP protège la charge utile d'un paquet IP. Les données traitées (payload) d'un paquet peut être crypté pour fournir peut garantir l'intégrité des données relatives à la confidentialité et à l'aide d'un ICV.

- **Associations de sécurité (AS) –** Les paramètres cryptographiques, les clés, le protocole de sécurité IP, les adresses IP, le protocole IP, les numéros de port et d'autres paramètres utilisés pour faire correspondre une certaine AS à un flux de trafic en particulier.
- **Base de données des associations de sécurité (SADB) –** La base de données contenant les associations de sécurité. Les AS sont référencées par l'index du paramètre de sécurité, le protocole de sécurité et l'adresse IP de destination. Ces trois éléments identifient AS SA IPsec de manière unique. IP lorsqu'un système reçoit qui comporte des données d'un en-tête de paquet ESP ou AH IPsec, () pour le système effectue une recherche dans le SA correspondante dans un SADB. Si aucun des documents est trouvée, celle-ci est SA permet l'utilisation de décryptage et de vérification de la IPsec paquet. Pas de correspondance de en cas d'échec de la vérification ou SA est trouvée, le paquet est rejeté.
- **Gestion des clés –** Génération et distribution sécurisée des clés qui utilisées par les algorithmes cryptographiques, ainsi que des AS où elles seront enregistrées.
- **Base de données des stratégies de sécurité (SPD) –** La base de données qui spécifie la stratégie de sécurité à appliquer au trafic IP. La base de données SPD filtre le trafic et identifie le mode de traitement des paquets. Un paquet peut être ignoré ou passé au clair. Ou protégé à l'aide d'IPsec, c'est-à-dire la stratégie de sécurité est appliquée.

  En ce qui concerne les paquets sortants, la stratégie IPsec détermine si IPSec doit être appliqué à un paquet IP. Si IPsec est appliquée, le module IP SA permet de lancer des recherches dans la mise en correspondance SADB utilise pour l'EdC à élément de contrat ainsi que mettre en oeuvre la règle.

  Pour les paquets entrants, la stratégie IPsec permet de s'assurer que le niveau de protection d'un paquet reçu est appropriée. Si la stratégie IP requiert les paquets à partir d'une certaine IPsec adresse à protéger à l'aide de non protégé, le système supprime tous les paquets. Si un paquet sortant est protégé par IPsec, la recherche dans la IP SA module EdC et SADB s'applique pour l'élément de contrat correspondant au paquet.

Les applications peuvent appeler IPsec pour appliquer les mécanismes aux paquets IP au niveau de chaque socket. Lorsque la stratégie IPsec est appliquée à un port sur lequel un socket est déjà connecté, le trafic qui utilise ce socket ne bénéficie pas de la protection IPsec. Bien sûr, les sockets ouverts sur un port *après* l'application de la stratégie IPsec en bénéficient aussi.

Flux de paquets IPsec

La Figure 6-1, “Application d'IPsec au processus de paquet sortant” montre le parcours d'un paquet IP lorsqu'IPsec est appelé sur un paquet sortant. Le diagramme du flux indique l'endroit

auquel les en-têtes d'authentification AH et les associations de sécurité ESP sont susceptibles d'être appliqués au paquet. Les sections suivantes expliquent les méthodes d'application de ces entités et de sélection des algorithmes.

La Figure 6-2, “Application d'IPsec au processus de paquet entrant” illustre le processus entrant IPsec.

FIGURE 6-1 Application d'IPsec au processus de paquet sortant

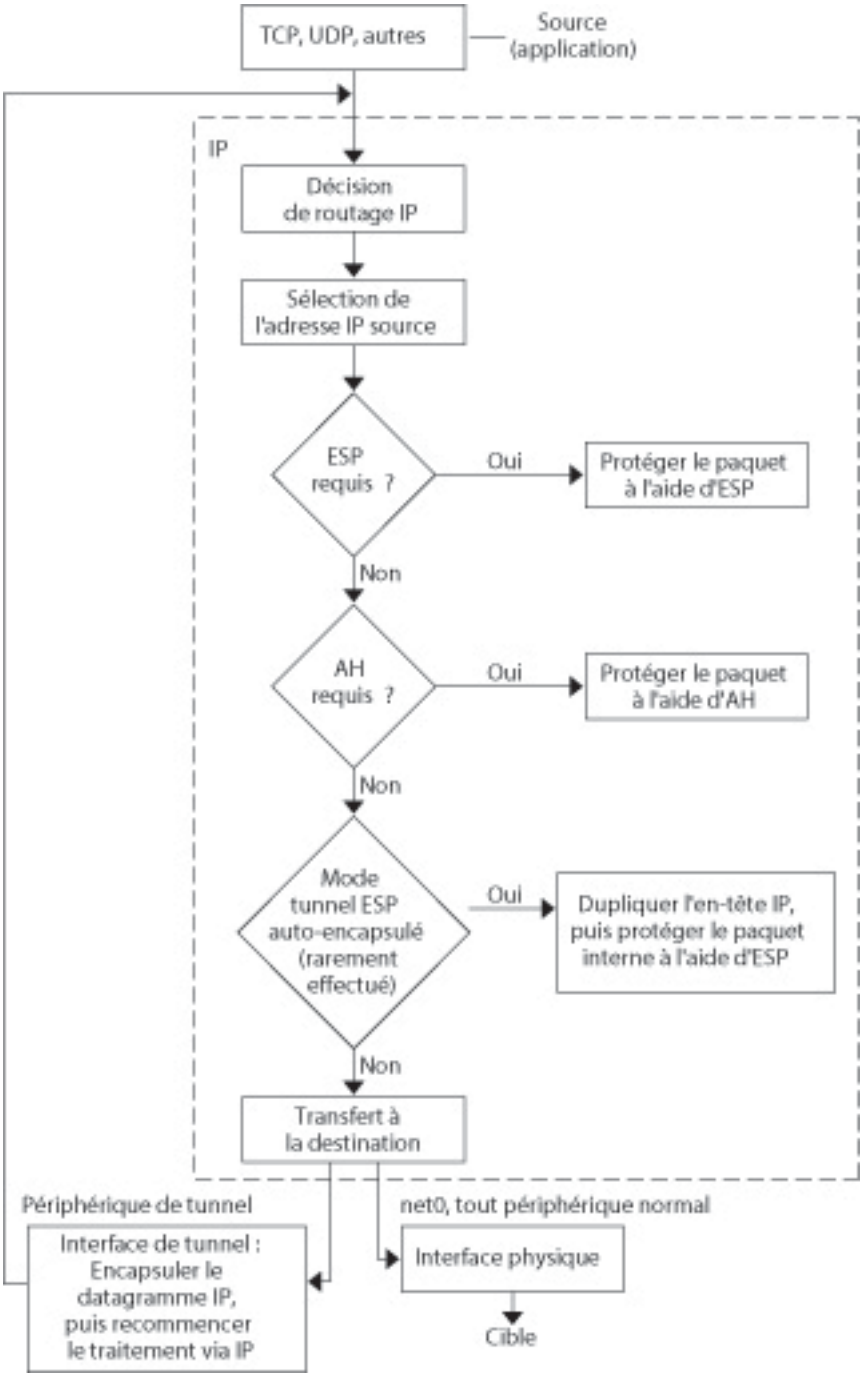
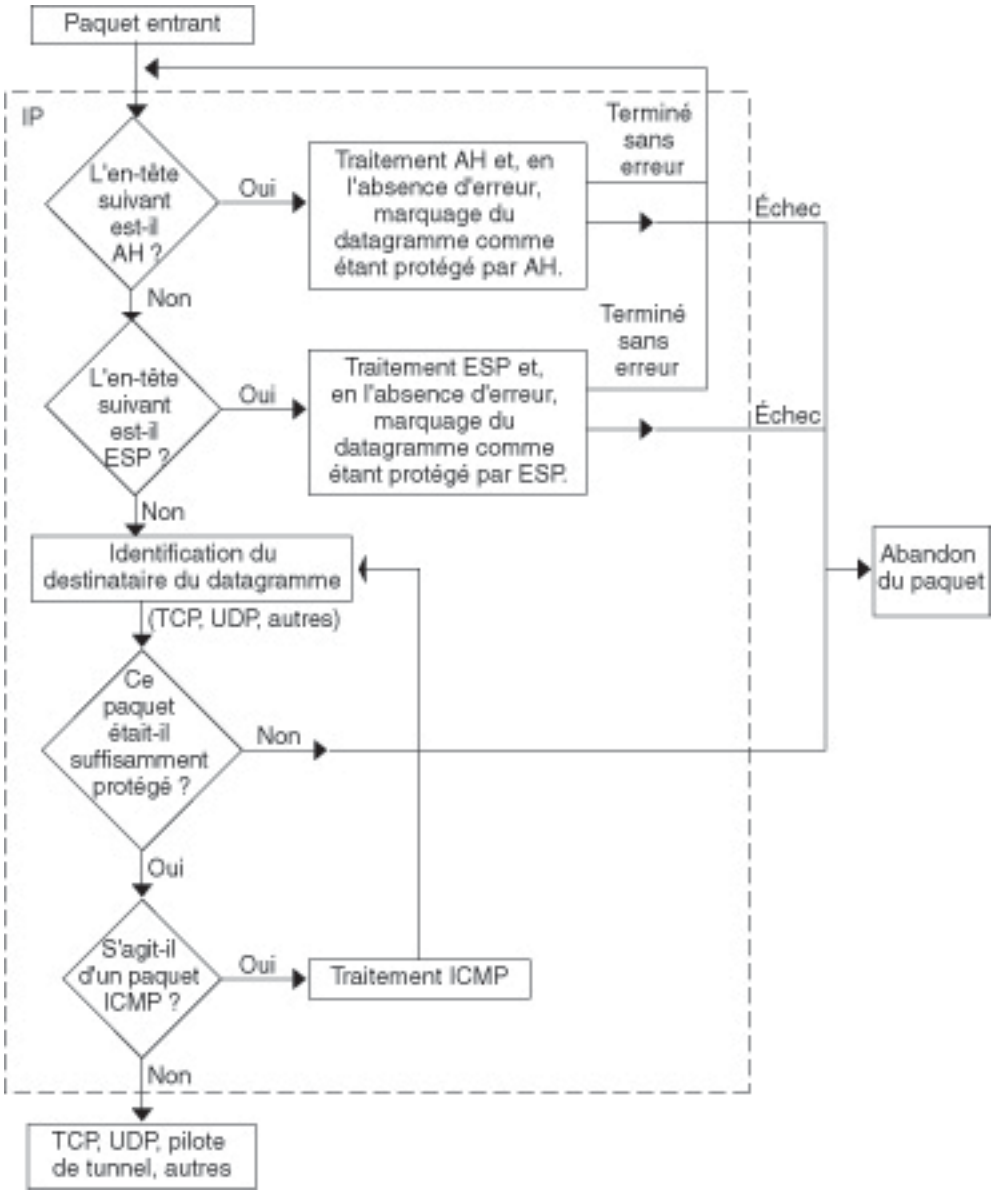


FIGURE 6-2 Application d'IPsec au processus de paquet entrant



Associations de sécurité IPsec

Une *association de sécurité* (AS) IPsec définit les propriétés de sécurité qui seront appliquées à un paquet IP dont les paramètres correspondent à ceux enregistrés dans l'AS. Chaque SA est unidirectionnel. Sont bidirectionnelles. parce que la plupart des deux SA les communications sont obligatoires pour une seule connexion.

Les trois éléments suivants réunis identifient une AS IPsec de manière unique :

- le protocole de sécurité (AH ou ESP) ;
- l'adresse IP de destination ;
- L'index du paramètre de sécurité

Le SPI du SA fournit une protection supplémentaire et est transmise dans l'en-tête AH ou ESP d'un packet protégé par IPsec. Les pages de manuel `ipsecah`(7P) et `ipsecesp`(7P) expliquent l'étendue de la protection AH et ESP. Une somme de contrôle d'intégrité permet d'authentifier un paquet. En cas d'échec de l'authentification, le paquet est rejeté.

Les associations de sécurité sont stockées dans une *base de données d'associations de sécurité* (SADB). Une interface d'administration socket, l'interface `PF_KEY`, autorise les applications privilégiées à gérer programmatiquement la base de données. Par exemple, le démon IKE et la commande `ipseckeys` font appel à l'interface socket `PF_KEY`.

Pour une description détaillée de la SADB IPsec, reportez-vous à la section “Base de données des associations de sécurité IPsec” à la page 228.

Pour plus d'informations sur la gestion de la SADB reportez-vous aux pages de manuel `pf_key`(7P) et `ipseckey`(1M).

Clé de gestion pour les associations de sécurité IPsec

Les associations de sécurité (SA) requièrent des numéros de clé pour l'authentification et le chiffrement. La gestion de ces numéros de clés s'appelle *gestion des clés*. Oracle Solaris fournit deux méthodes de gestion des clés pour les AS IPsec : IKE et la méthode manuelle.

Génération pour IPsec SA IKE

Le protocole IKE (Internet Key Exchange) assure la gestion des clés de manière automatique. Oracle Solaris 11.2 prend en charge la version 2 (IKEv2) et la version 1 (IKEv1) du protocole IKE.

L'utilisation d'IKE pour gérer les SA IPsec est recommandée. Ces protocoles de gestion des clés offrent les avantages suivants :

- Configuration simple
- L'authentification de pairs permettent la mise en place de méthodes
- Générer automatiquement des EdC avec une clé aléatoire source d'un niveau de qualité élevé
- N'est pas nécessaire pour générer les nouveaux EdC une intervention administrative.

Pour plus d'informations, reportez-vous à la section “Fonctionnement d'IKE” à la page 134.

Pour configurer IKE, reportez-vous au Chapitre 9, Configuration d'IKEv2. Si vous communiquer avec un système qui ne prend pas en charge le protocole IKEv2, suivez les instructions du Chapitre 10, Configuration d'IKEv1.

Génération de clés pour IPsec SA manuelle

L'utilisation de clés manuelles et est plus complexe que est potentiellement dangereuse IKE. Un fichier système, `/etc/inet/secret/ipseckeys`, contient les clés de chiffrement. Si elles ne leur est compromise, enregistrée peut être utilisée pour décrypter le trafic réseau. Les clés IKE fréquemment car les modifications de la situation vis - à - vis, la fenêtre est moins important de tels un compromis. L'utilisation du fichier `ipseckeys` ou de son interface de commande, `ipseckey`, convient uniquement aux systèmes qui ne prennent pas en charge IKE.

Bien qu'elle présente un nombre limité d'options générales, la commande `ipseckey` prend en charge un langage de commande enrichi. Si vous le souhaitez, une interface de programmation de génération manuelle de clés peut transmettre les demandes. Pour plus d'informations, reportez-vous aux pages de manuel `ipseckey`(1M) et `pf_key`(7P).

En règle générale, les SA sont générées manuellement lorsqu'IKE n'est pas disponible pour une raison quelconque. Cependant, si les valeurs SPI sont uniques, la génération manuelle des SA et IKE peuvent être utilisés en même temps.

Protection des protocoles IPsec

IPsec offre deux protocoles de sécurité dans le cadre de la protection des données :

- AH (Authentication Header, en-tête d'authentification)
- ESP (Encapsulating Security Payload, association de sécurité)

L'intégrité des données AH à l'aide fournit un algorithme d'authentification. Il ne chiffre pas le paquet.

ESP protège le paquet comportant en règle générale, un algorithme de chiffrement et fournit l'intégrité des données grâce à un algorithme d'authentification. Fournir un nom utilisateur et une partie du chiffrement et les algorithmes de chiffrement GCM l'authentification, tel que AES.

Le protocole ne peut pas être utilisé avec AH la translation d'adresse réseau (NAT).

En-tête d'authentification

L'en-tête d'authentification offre l'authentification des données, un niveau élevé d'intégrité et la protection de rediffusion des paquets IP. AH protège la majeure partie du paquet IP. Comme l'illustre la figure suivante, AH est inséré entre l'en-tête IP et l'en-tête de transport.

L'en-tête de transport peut être TCP, UDP, SCTP ou ICMP. Dans le cas de l'utilisation d'un tunnel, l'en-tête de transport peut être un autre en-tête IP.

ESP (Encapsulating Security Payload, association de sécurité)

Le module protocole ESP assure la confidentialité des encapsulations ESP. ESP propose également les services AH. Toutefois, externe ESP IP ne protège pas l'en-tête . ESP fournit des services d'authentification afin d'assurer l'intégrité du paquet protégé. Du fait qu'ESP utilise une technologie de chiffrement, un système fournissant ESP peut être soumis à des lois sur le contrôle des importations et exportations.

L'en-tête et le bloc de fin ESP encapsulent la charge utile IP. Est utilisé avec ESP lorsque le chiffrement, il est appliqué les données traitées (payload) uniquement sur les IP, comme illustré ci-dessous .

Le paquet dans un en-tête TCP est authentifié, et qu'il ESP encapsule les en-tête TCP et ses données. S'il s'agit d'un paquet IP-in-IP, ESP protège le paquet IP interne. La stratégie par socket permet l'*auto-encapsulation*. Ainsi, ESP peut encapsuler les options IP, le cas échéant.

L'auto-encapsulation peut être utilisé en écrivant un programme utilisant la fonction `setsockopt()`. Lorsque l'auto-encapsulation est définie, l'en-tête IP est copié afin de créer un paquet IP-in-IP. Par exemple, lorsque l'auto-encapsulation n'est pas définie sur un socket TCP, le paquet est envoyé au format suivant :

`[ IP(a -> b)` *options* `+ TCP + data ]`

Lorsque l'auto-encapsulation est définie sur ce socket TCP, le paquet est envoyé au format suivant :

`[ IP(a -> b) + ESP [ IP(a -> b)` *options* `+ TCP + data ] ]`

Pour plus d'informations, reportez-vous à la section “Modes Transport et Tunnel dans IPsec” à la page 101.

Considérations de sécurité lors de l'utilisation de AH et ESP

Le tableau ci-dessous compare les protections fournies par AH et ESP sont fournis par.

TABLEAU 6-1 Protections assurées par AH et ESP dans IPsec

Protocole	Paquets protégés	Protection	Attaques contrées
AH	Protection des paquets de l'en-tête IP à la fin de la transporter les données	Intégrité élevée, authentification des données : ■ réception garantie des données exactes envoyées par l'expéditeur ■ attaques par rejeu possibles lorsqu'un AH n'active pas la protection par rejeu	Rejeu, couper-coller
ESP	Protection des paquets de l'ESP à la fin de l'en-tête de données de transport	Chiffrement de la charge utile IP à l'aide de l'option de chiffrement Confidentialité garantie	Ecoute électronique
		Protection identique à la protection AH à l'aide de l'option d'authentification	Rejeu, couper-coller
		Intégrité élevée, authentification des données et confidentialité à l'aide des deux options	Rediffusion, couper-coller, écoute électronique

Authentification et chiffrement dans IPsec

La sécurité IPsec utilise deux types d'algorithmes : les algorithmes d'authentification et les algorithmes de chiffrement. Le protocole AH recourt à des algorithmes d'authentification. Le protocole ESP peut utiliser aussi bien les algorithmes d'authentification que les algorithmes de chiffrement. La commande `ipsecalgs` affiche la liste des algorithmes présents sur le

système, ainsi que leurs propriétés. Pour plus d'informations, reportez-vous à la page de manuel `ipsecalgs`(1M) Vous pouvez aussi utiliser les fonctions décrites dans la page de manuel `getipsecalgbyname`(3NSL)pour obtenir les propriétés des algorithmes.

IPsec utilise la structure cryptographique pour effectuer le chiffrement et l'authentification. La structure cryptographique permet à IPsec de tirer profit des prend en charge l'accélération matérielle auquel cas le matériel.

Pour plus d'informations, reportez-vous aux références suivantes :

- Chapitre 1, “ Structure cryptographique ” du manuel “ Gestion du chiffrement et des certificats dans Oracle Solaris 11.2 ”
- Chapitre 8, “ Introduction to the Oracle Solaris Cryptographic Framework ” du manuel “ Developer’s Guide to Oracle Solaris 11 Security ”

Stratégies de protection IPsec

Vous pouvez appliquer les stratégies de protection IPsec aux niveaux suivants :

- Niveau à l'échelle du système
- Niveau par socket

IPsec applique stratégie à l'échelle du système aux paquets sortants et entrants qui correspondent à une règle de stratégie IPsec. La règle peut indiquer un algorithme spécifique ou autoriser une l'une des différentes algorithmes. Vous pouvez appliquer d'autres règles aux paquets sortants, en raison des données supplémentaires connues du système.

Les paquets entrants sont acceptés ou rejetés. La décision d'accepter ou de rejeter un paquet sortant est fonction de plusieurs critères. Si les critères se chevauchent ou entrent en conflit, la règle analysée en premier est utilisée.

Vous pouvez spécifier des exceptions à une stratégie IPsec qui s'applique autrement à la majorité des paquets. C'est-à-dire que vous avez également la possibilité de *contourner* une stratégie IPsec. Le système ne tiennent pas compte du applicable à l'ensemble du système ou peut être par socket.

Au niveau du trafic sur un système-y compris les zones IP partagées, les stratégies sont mises en oeuvre mais adresse les mécanismes de sécurité à proprement parler ne sont pas appliqués. En revanche, la stratégie sortante sur un paquet interne au système se traduit par un paquet sortant auquel ces mécanismes ont été appliqués. Pour les zones en mode IP exclusif, la stratégie a été mise en application et les mécanismes de sécurité à proprement parler sont appliqués.

Utilisez le fichier `ipsecinit.conf` et la commande `ipsecconf` pour configurer les stratégies IPsec. Pour obtenir des détails et des exemples, reportez-vous à la page de manuel `ipsecconf`(1M) et au Chapitre 7, Configuration d'IPsec.

Modes Transport et Tunnel dans IPsec

Les normes IPsec définissent deux modes de fonctionnement distincts pour IPsec : le *mode transport* et le *mode tunnel*. La principale différence entre le mode Tunnel et le mode Transport est l'emplacement d'application de la stratégie. En mode Tunnel, le packet d'origine est encapsulé dans une autre en-tête IP. Les adresses d'une autre en-tête peuvent être différentes.

Les paquets sont protégés par AH, ESP ou ces deux protocoles dans chaque mode. Les modes diffèrent dans l'application de la stratégie, comme suit :

- En mode Transport, les adresses IP de l'en-tête externe sont utilisées pour déterminer la stratégie IPsec qui sera appliquée au paquet.
- En mode Tunnel, deux en-têtes IP sont envoyés. Le paquet IP interne détermine la stratégie IPsec qui protège son contenu.

  Le mode Tunnel peut être appliqué à toute combinaison de systèmes en fin de chaîne et de systèmes intermédiaires, telle que les passerelles de sécurité.

En mode transport, l'en-tête IP, l'en-tête suivant et tous les ports pris en charge par l'en-tête suivant peuvent être utilisés pour déterminer la stratégie IPsec. En fait, IPsec peut mettre en oeuvre différentes stratégies en mode Transport entre deux adresses IP au niveau d'un seul port. Par exemple, si l'en-tête suivant est un en-tête TCP, qui prend en charge les ports, la stratégie IPsec peut alors être définie pour un port TCP de l'adresse IP externe.

Le mode Tunnel ne fonctionne que pour les paquets IP-in-IP. En mode Tunnel, la stratégie IPsec est mise en oeuvre sur le contenu du paquet IP interne. Différentes stratégies IPsec peuvent être mises en oeuvre pour différentes adresses IP internes. En d'autres termes, l'en-tête IP interne, ainsi que son en-tête suivant et les ports que ce dernier prend en charge, peuvent mettre en oeuvre une stratégie. Contrairement au mode transport, le mode tunnel ne permet pas à l'en-tête IP extérieur de dicter la stratégie de son paquet IP interne.

Par conséquent, en mode Tunnel, la stratégie IPsec peut être spécifiée pour les sous-réseaux d'un LAN derrière un routeur et pour les ports de ces sous-réseaux. La stratégie IPsec peut également être spécifiée pour des adresses IP données (des hôtes) sur ces sous-réseaux. Les ports de ces hôtes peuvent aussi avoir une stratégie IPsec spécifique. Toutefois, si un protocole de routage dynamique est exécuté sur un tunnel, veillez à ne pas utiliser de sélection de sous-réseau ou d'adresse, car la vue de la topologie réseau sur le réseau homologue pourrait être modifiée. Les modifications annuleraient la stratégie IPsec statique. La section “Protection d'un VPN à l'aide d'IPsec” à la page 117 contient des exemples de mises en tunnel comprenant la configuration de routes statiques.

Dans Oracle Solaris, le mode tunnel peut être mise en oeuvre uniquement sur une interface réseau à mise en tunnel IP. Pour plus d'informations sur les interfaces de mise en tunnel reportez-vous au Chapitre 4, “ Administration de tunnels sur IP ” du manuel “ Administration des réseaux TCP/IP, d’IPMP et des tunnels IP dans Oracle Solaris 11.2 ”. La stratégie IPsec fournit un mot-clé `tunnel` pour sélectionner une interface réseau de mise en tunnel IP. Lorsque le mot-clé `tunnel` figure dans une règle, tous les sélecteurs spécifiés dans cette règle s'appliquent au paquet interne.

La figure suivante illustre un en-tête IP avec un paquet TCP non protégé.

FIGURE 6-3 Paquet IP non protégé transportant des informations TCP

En mode Transport, ESP protège les données, comme illustré ci-dessous. La zone ombrée indique la partie chiffrée du paquet.

FIGURE 6-4 Paquet IP protégé transportant des informations TCP

En mode tunnel, l'ensemble du paquet est *à l'intérieur* de l'en-tête ESP. Le paquet de la Figure 6-3, “Paquet IP non protégé transportant des informations TCP” est protégé en mode tunnel par un en-tête IPsec externe, ESP dans ce cas, comme indiqué sur l'illustration suivante.

FIGURE 6-5 Paquet IPsec protégé en mode Tunnel

La stratégie IPsec fournit des mots-clés pour le monde Tunel et Transport. Pour plus d'informations, reportez-vous aux références suivantes :

- Pour plus d'informations sur la stratégie par socket, reportez-vous à la page de manuel `ipsec`(7P).
- Pour obtenir un exemple de la stratégie par socket, reportez-vous à la section “Utilisation d'IPsec pour protéger Web Server Communication avec d'autres serveurs” à la page 115.
- Pour plus d'informations sur les tunnels, reportez-vous à la page de manuel `ipsecconf`(1M).

- Pour un exemple de configuration de tunnel, reportez-vous à la section “Protection de la connexion entre deux réseaux locaux à l'aide d'IPsec en mode Tunnel” à la page 120.

Réseaux privés virtuels et IPsec

Le terme VPN est souvent utilisé pour décrire un réseau point à point privé et sécurité construit par-dessus un réseau public, par exemple Internet. Si le point à point, ou VPN réseau peut être utilisé pour connecter des systèmes sur les réseaux privés ou les réseaux de systèmes sur les réseaux privés ensemble.

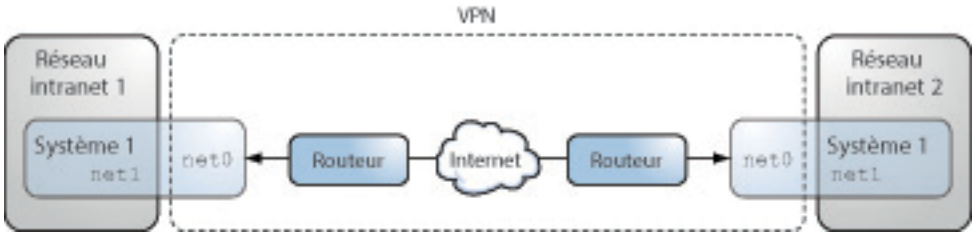
Un tunnel configuré est une interface point à point. Le tunnel permet l'encapsulation d'un paquet IP dans un autre paquet IP. Un tunnel correctement configuré requiert une source et une destination. Pour plus d'informations, reportez-vous à la section “ Création et configuration d’un tunnel IP ” du manuel “ Administration des réseaux TCP/IP, d’IPMP et des tunnels IP dans Oracle Solaris 11.2 ”.

Un tunnel crée une interface physique apparente avec IP. Le trafic qui est positionné au-dessus IP interface de tunnel IP protégé à l'aide d'IPsec.

L'interface de tunnel d'Oracle Solaris peut être utilisée pour encapsuler, ou *mettre en tunnel*, un paquet IP d'un système à un autre. Ajoute une mis en tunnel de paquets IP l'en-tête IP sur le devant de l'en-tête d'origine. Utilise des adresses de sorte que les nouvelles qui sont à envoyer en-tête sur le réseau public. Les adresses sont représentées par les interfaces `net0` dans le diagramme ci-après.

La figure suivante illustre comment deux des sites peuvent utiliser VPN IPsec, séparés les uns des autres pour créer un. Le trafic entre `Intranet 1` et `Intranet 2` est mis en tunnel via Internet au moyen de l'encapsulation IP-in-ESP. Dans ce cas, les adresses `net0` sont utilisées dans les en-têtes IP extérieurs, les adresses IP internes étant celles des paquets mis en tunnel provenant des réseaux intranet. Les adresses IP car qui délimitent la période au-delà , ils sont couverts par ESP avec le trafic sont protégés depuis le contrôle est répartie sur Internet.

FIGURE 6-6 Réseau privé virtuel



Pour voir un exemple détaillé de la procédure de configuration, reportez-vous à la section “Protection de la connexion entre deux réseaux locaux à l'aide d'IPsec en mode Tunnel” à la page 120.

IPsec et FIPS 140

Vous pouvez aisément configurer IPsec pour qu'il soit conforme aux exigences FIPS 140 sur un système sur lequel FIPS 140 est activé. Vous êtes chargé de ne choisir que des algorithmes validés FIPS 140 pour créer les clés et les certificats. Les procédures et exemples de ce guide utilisent des algorithmes approuvés FIPS 140, sauf lorsque l'algorithme `any` est spécifié.

Remarque - Si l'exigence stricte est de n'utiliser qu'une cryptographie validée FIPS 140-2, vous devez exécuter la version Oracle Solaris 11.1 SRU 5.5 ou la version Oracle Solaris 11.1 SRU 3. Oracle a obtenu la validation FIPS 140-2 relative à la structure cryptographique dans ces deux versions spécifiques. Oracle Solaris 11.2 s'appuie sur cette base validée pour apporter des améliorations logicielles concernant la performance, les fonctionnalités et la fiabilité. Configurez Oracle Solaris 11.2 chaque fois que possible en mode FIPS 140-2 pour bénéficier de ces améliorations.

Vous trouverez ci-dessous les mécanismes disponibles pour IPsec et approuvés pour une utilisation dans Oracle Solaris en Mode FIPS 140 :

- AES dans les modes CBC, CCM, GCM et GMAC dans des longueurs de clés 128 bits à 256 bits.
- 3DES
- SHA1
- SHA2 dans les longueurs de clés 256 bits et 512 bits.

Pour la liste définitive des algorithmes validés FIPS 140 pour Oracle Solaris, reportez-vous à http://www.oracle.com/technetwork/topics/security/140sp2061-2082028.pdf. Pour une discussion plus complète sur le sujet, reportez-vous à “ Using a FIPS 140 Enabled System in Oracle Solaris 11.2 ”.

Passage de la translation d'adresses et IPsec

IKE peut négocier des SA IPsec dans une zone NAT Cela permet aux systèmes de se connecter en toute sécurité à partir d'un réseau distant, même lorsqu'ils résident derrière un périphérique NAT. Par exemple, les employés travaillant à domicile ou se connectant depuis un site de conférence peuvent protéger leur trafic à l'aide d'IPsec.

Un routeur NAT permet d'associer une adresse interne privée à une adresse Internet unique. Les routeurs NAT équipent de nombreux points d'accès publics à Internet, comme ceux qu'on trouve dans les hôtels.

L'utilisation d'IKE lorsqu'un routeur NAT figure entre les systèmes de communication correspond au NAT-T (NAT traversal). NAT-T présente les restrictions suivantes :

- Le protocole AH dépend d'un en-tête IP permanent, ce qui empêche son fonctionnement avec NAT-T. Le protocole ESP s'utilise avec NAT-T.
- Le routeur NAT n'applique pas de règles de traitement particulières. Un routeur NAT obéissant à des règles de traitement IPsec pourrait intervenir dans l'implémentation de NAT-T.
- NAT-T fonctionne uniquement lorsque l'initiateur IKE est le système derrière le routeur NAT. Un répondeur IKE ne peut pas se trouver derrière un routeur NAT, à moins que celui-ci ne soit programmé pour transférer des paquets IKE au système adéquat figurant derrière lui.

Les RFC suivantes décrivent la fonctionnalité de NAT et les limites de NAT-T. Des copies des RFC sont disponibles à l'adresse http://www.rfc-editor.org.

- RFC 3022, "Traditional IP Network Address Translator (Traditional NAT)", janvier 2001
- RFC 3715, "IPsec-Network Address Translation (NAT) Compatibility Requirements", mars 2004
- RFC 3947, "Negotiation of NAT-Traversal in the IKE", janvier 2005
- RFC 3948, "UDP Encapsulation of IPsec Packets", janvier 2005

IPsec et SCTP

Oracle Solaris prend en charge le protocole SCTP (Streams Control Transmission Protocol). Bien que prise en charge, l'utilisation du protocole SCTP et du numéro de port SCTP dans le cadre de la spécification de la stratégie IPsec n'est pas stable. Les extensions IPsec pour SCTP spécifiées dans le document RFC 3554 ne sont pas encore implémentées. Ces restrictions peuvent être source de complications lors de la création de la stratégie IPsec pour SCPT.

SCTP peut avoir recours à plusieurs adresses source et cible dans le cadre d'une association SCTP unique. Lorsque la stratégie IPsec est appliquée à une seule adresse source ou cible, la communication peut échouer si SCTP change l'adresse source ou cible de cette association. La stratégie IPsec reconnaît uniquement l'adresse d'origine. Pour plus d'informations sur SCTP, reportez-vous à la RFC Stream Control Transmission Protocol (SCTP).

IPsec et les zones Oracle Solaris

IPsec est pris en charge dans les zones. Chaque zone peut posséder la stratégie IPsec et IKE sa propre configuration. Une zone peut être traité comme un hôte distinct.

L'exception concernant des zones, ce qui IP partagé ne possèdent pas leur propre IP pile. Les zones en mode IP partagé, pour la stratégie IPsec et IKE de configuration sont effectuées dans la zone globale. La stratégie IPsec IP partagées des règles de zones pour utiliser l'adresse IP affectée à cette zone.

Pour plus d'informations, reportez-vous au Chapitre 1, “ Présentation d’Oracle Solaris Zones ” du manuel “ Présentation d’Oracle Solaris Zones ”.

IPsec et des machines virtuelles

Le protocole IPsec fonctionne avec des machines virtuelles (VM). Servez-vous du serveur Oracle VM Server pour créer des machines virtuelles sur les systèmes SPARC. Sur les systèmes x86, vous pouvez utiliser Oracle VM VirtualBox. Pour plus d'informations sur la configuration, reportez-vous au guide d'administration de la version de votre MV Oracle.

Commandes et fichiers de configuration d'IPsec

Le Tableau 6-2, “Commandes et fichiers de configuration d'IPsec sélectionnés” décrit les fichiers, commandes et identificateurs de service utilisés pour configurer et gérer IPsec. Exhaustif, ce tableau inclut les commandes et fichiers de gestion des clés.

Pour plus d'informations sur les identificateurs de service, reportez-vous au Chapitre 1, “ Introduction à l’utilitaire de gestion des services ” du manuel “ Gestion des services système dans Oracle Solaris 11.2 ”.

Pour obtenir des instructions sur la mise en oeuvre d'IPsec sur votre réseau, reportez-vous à la section “Protection du trafic réseau à l'aide d'IPsec” à la page 109.

Pour obtenir des informations de référence sur IPsec, reportez-vous au Chapitre 12, IPsec et référence de gestion des clés.

TABLEAU 6-2 Commandes et fichiers de configuration d'IPsec sélectionnés

IPsec Commande, fichier ou service	Description	Page de manuel
`svc:/network/ipsec/ipsecalgs`	Service SMF qui gère les algorithmes IPsec.	`ipsecalgs`(1M)

IPsec Commande, fichier ou service	Description	Page de manuel
`svc:/network/ipsec/manual-key`	Service SMF qui gère les SA IPsec dont les clés sont spécifiées manuellement.	`ipseckey`(1M)
`svc:/network/ipsec/policy`	Le service SMF qui gère la stratégie IPsec.	`smf`(5), `ipsecconf`(1M)
`svc:/network/ipsec/ike:ikev2`, `svc:/network/ipsec/ike:default`	Les instances de services SMF pour la gestion automatique des AS IPsec à l'aide d'IKE.	`smf`(5), `in.ikev2d`(1M), `in.iked`(1M)
Fichier `/etc/inet/ipsecinit.conf`	Fichier de stratégie IPsec. Le service `policy` de SMF s'en sert pour configurer la stratégie IPsec lors de l'initialisation du système.	`ipsecconf`(1M)
Commande `ipsecconf`	Commande de stratégie IPsec. Utile pour afficher et modifier la stratégie IPsec actuelle, ainsi que pour effectuer des tests. Le service `policy` de SMF s'en sert pour configurer la stratégie IPsec lors de l'initialisation du système.	`ipsecconf`(1M)
Interface socket `PF_KEY`	Interface pour la base de données des associations de sécurité (SADB). Responsable de la gestion manuelle et automatique des clés.	`pf_key`(7P)
Commande `ipseckey`	IPsec SAs keying command. `ipseckey` is a command-line front end to the `PF_KEY` interface. `ipseckey` peut créer, supprimer ou modifier des AS.	`ipseckey`(1M)
Fichier `/etc/inet/secret/ipseckeys`	Contient des AS dotées de clés attribuées manuellement. Le service `manual-key` de SMF s'en sert pour configurer les AS lors de l'initialisation du système.	
Commande `ipsecalgs`	Commande d'algorithmes IPsec. Utile pour l'affichage et la modification de la liste d'algorithmes IPsec et de leurs propriétés. Le service `ipsecalgs` de SMF s'en sert pour synchroniser les algorithmes IPsec connus avec le noyau lors de l'initialisation du système.	`ipsecalgs`(1M)
Fichier `/etc/inet/ipsecalgs`	Contient les définitions d'algorithme et les mécanismes IPsec configurés. Ce fichier est géré par la commande `ipsecalgs` et ne doit jamais être modifié manuellement.	
fichier `/etc/inet/ike/ikev2.config`	Fichier de configuration et de stratégie IKEv2. La gestion des clés se base sur des règles et des paramètres généraux à partir de ce fichier. Reportez-vous à la section “Utilitaires et fichiers d'IKEv2” à la page 229.	`ikev2.config`(4)
Fichier `/etc/inet/ike/config`	Fichier de configuration et de stratégie IKEv1. Par défaut, ce fichier n'existe pas. La gestion des clés se base sur des règles et des paramètres généraux à partir de ce fichier. Reportez-vous à la section “Utilitaires et fichiers d'IKEv1” à la page 233. Si ce fichier existe, le service `svc:/network/ipsec/ike:default` démarre le démon IKEv1, `in.iked`.	`ike.config`(4)

♦ ♦ ♦ CHAPITRE 7

Configuration d'IPsec

Ce chapitre fournit les procédures d'implémentation d'IPsec sur votre réseau. Ces procédures sont décrites dans les sections suivantes :

Vous trouverez une présentation d'IPsec au Chapitre 6, A propos de l'architecture de sécurité IP. Pour obtenir des informations de référence sur IPsec, reportez-vous au Chapitre 12, IPsec et référence de gestion des clés.

Remarque - Les tâches suivantes partent du principe que les systèmes reçoivent des adresses IP statiques et que vous exécutez le profil de configuration réseau `DefaultFixed`. Si la commande `netadm list` renvoie `Automatic`, reportez-vous à la page de manuel `netcfg`(1M) pour plus d'informations.

Protection du trafic réseau à l'aide d'IPsec

Les procédures de cette section permettent de sécuriser le trafic entre deux systèmes et de sécuriser un serveur Web. Pour protéger un VPN (Virtual Private Network, réseau privé virtuel), reportez-vous à la section "Protection d'un VPN à l'aide d'IPsec" à la page 117. Pour obtenir des informations sur les procédures supplémentaires de gestion d'IPsec et sur l'utilisation des commandes SMF avec IPsec et IKE, reportez-vous à la section "Tâches IPsec supplémentaires" à la page 125.

Les informations ci-dessous s'appliquent à toutes les tâches de configuration IPsec :

- **IPsec et zones –** Chaque système est une zone globale ou une zone en mode IP exclusif. Pour plus d'informations, reportez-vous à la section "IPsec et les zones Oracle Solaris" à la page 106.
- **IPsec et Mode FIPS 140 –** En tant qu'administrateur IPsec, vous êtes en charge de choisir des algorithmes validés FIPS 140 pour Oracle Solaris. Les procédures et exemples de ce

chapitre utilisent des algorithmes approuvés FIPS 140, sauf lorsque l'algorithme `any` est spécifié.

- **IPsec et RBAC –** Pour l'utilisation des rôles pour administrer IPsec, reportez-vous au Chapitre 3, “ Affectation de droits dans OracleÂ Solaris ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”. La section “Configuration d'un rôle pour la sécurité réseau” à la page 128 présente un exemple.
- **IPsec et SCTP –** Vous pouvez utiliser IPsec pour protéger les associations SCTP (Streams Control Transmission Protocol, protocole de transmission de contrôle de flux), mais avec prudence. Pour plus d'informations, reportez-vous à la section “IPsec et SCTP” à la page 105.
- **Etiquettes IPsec et Trusted Extensions –** Sur les systèmes configurés avec la fonctionnalité Trusted Extensions d'Oracle Solaris, il est possible d'ajouter des étiquettes aux paquets IPsec. Pour plus d'informations, reportez-vous à la section “ Administration d’IPsec avec étiquettes ” du manuel “ Configuration et administration de Trusted Extensions ”.
- **Adresses IPv4 et IPv6 :** les exemples IPsec dans ce guide utilisent des adresses IPv4. Oracle Solaris prend également en charge les adresses IPv6. Pour configurer IPsec pour un réseau IPv6, remplacez les adresses des exemples par des adresses IPv6. Lorsque vous protégez des tunnels avec IPsec, vous pouvez utiliser des adresses IPv4 et IPv6 en guise d'adresses internes et externes. Ce type de configuration vous permet d'acheminer IPv6 via tunnel sur un réseau IPv4, par exemple.

La liste des tâches ci-dessous répertorie les procédures de configuration d'IPsec sur un ou plusieurs systèmes. Les pages de manuel `ipsecconf`(1M), `ipseckey`(1M) et `ipadm`(1M) man décrivent également des procédures utiles dans leurs sections d'exemples respectives.

TABLEAU 7-1 Liste de tâches pour la protection du trafic à l'aide d'IPsec

Tâche	Description	Pour obtenir des instructions
Sécurisation du trafic entre deux systèmes	Protège les paquets transmis d'un système à un autre.	“Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111
Sécurisation d'un serveur Web à l'aide de la stratégie IPsec	Requiert un trafic non-Web pour utiliser IPsec. Les clients Web sont identifiés par des ports particuliers ; les vérifications IPsec sont ignorées.	“Utilisation d'IPsec pour protéger Web Server Communication avec d'autres serveurs” à la page 115
Utilisation d'IKE pour créer automatiquement des numéros de clés pour les SA IPsec.	Méthode recommandée pour la création des SA IPsec.	“Configuration d'IKEv2” à la page 143 et “Configuration d'IKEv1” à la page 171
Configuration d'un réseau privé virtuel (VPN, Virtual Private Network) sécurisé	Définit IPsec entre deux systèmes sur Internet.	“Protection d'un VPN à l'aide d'IPsec” à la page 117
Configuration de la gestion manuelle des clés.	Fournit les données brutes sans utiliser IKE pour les SA IPsec.	“Création manuelle de clés IPsec” à la page 125

▼ Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec

Cette procédure correspond à la configuration suivante :

- Les systèmes reçoivent des adresses IP statiques et exécutent le profil de configuration réseau `DefaultFixed`. Si la commande `netadm list` renvoie `Automatic`, reportez-vous à la page de manuel `netcfg`(1M) pour plus d'informations.
- Les systèmes s'appellent `enigma` et `partym`.
- Chaque système dispose d'une adresse IP. Il peut d'agir d'une adresse IPv4, IPv6 ou les deux. Cette procédure utilise des adresses IPv4.
- Chaque système est une zone globale ou une zone en mode IP exclusif. Pour plus d'informations, reportez-vous à la section “IPsec et les zones Oracle Solaris” à la page 106.
- Chaque système chiffre le trafic avec l'algorithme AES et l'authentifie avec SHA-2.

Remarque - L'algorithme SHA-2 peut être requis pour certains sites.

- Chaque système utilise des associations de sécurité partagées (SA, Security Associations).

  Avec les SA partagées, une seule paire de SA est suffisante pour protéger les deux systèmes.

Remarque - Pour utiliser IPsec avec des étiquettes sur un système Trusted Extensions, reportez-vous aux étapes supplémentaires indiquées à la section “ Application des protections IPsec dans un réseau TrustedÂ Extensions multiniveau ” du manuel “ Configuration et administration de Trusted Extensions ”.

Avant de commencer

Un utilisateur disposant des droits spécifiques peut exécuter les commandes suivantes sans être utilisateur `root` :

- Pour exécuter les commandes de configuration, vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau).
- Dans ce rôle administratif, vous pouvez modifier les fichiers système IPsec et créer des clés à l'aide de la commande `pfedit`.
- Pour modifier le fichier `hosts`, vous devez être dans le rôle `root` ou avoir une autorisation explicite pour modifier le fichier. Reportez-vous à l'Exemple 7-7, “Octroi à un utilisateur de confiance de l'autorisation de configurer et gérer IPsec”.

Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, “Configuration de la stratégie IPsec à distance via une connexion `ssh`” et à “ Administration à distance de ZFS avec le shell sécurisé ” du manuel “ Gestion de l’accès au shell sécurisé dans Oracle Solaris 11.2 ” pour des instructions sur la connexion à distance sécurisée.

1. **Sur chaque système, ajoutez des entrées pour l'hôte au fichier `/etc/inet/hosts`.**

   Cette étape permet au service de noms local de résoudre les noms de service des adresses IP sans dépendre d'un service de noms en réseau.

   a. **Sur un système appelé `partym`, saisissez les lignes suivantes dans le fichier `hosts` :**

   ```
   ## Secure communication with enigma
   192.168.116.16 enigma
   ```

   b. **Sur un système appelé `enigma`, saisissez les lignes suivantes dans le fichier `hosts` :**

   ```
   ## Secure communication with partym
   192.168.13.213 partym
   ```

2. **Sur chaque système, créez le fichier de stratégie IPsec.**

   Le nom de fichier est `/etc/inet/ipsecinit.conf`. Vous en trouverez un exemple dans le fichier `/etc/inet/ipsecinit.sample`.

   ```
   # pfedit /etc/inet/ipsecinit.conf
   ```

3. **Ajoutez une entrée de stratégie IPsec au fichier `ipsecinit.conf`.**

   La syntaxe des entrées de stratégie IPsec est décrite dans la page de manuel `ipsecconf`(1M).

   a. **Sur le système `enigma`, ajoutez la stratégie ci-dessous :**

   ```
   {laddr enigma raddr partym} ipsec {encr_algs aes encr_auth_algs sha512 sa shared}
   ```

   Etant donné que le mot-clé `dir` n'est pas utilisé, la stratégie s'applique aux paquets entrants et sortants.

   b. **Sur le système `partym`, ajoutez la même stratégie :**

   ```
   {laddr partym raddr enigma} ipsec {encr_algs aes encr_auth_algs sha512 sa shared}
   ```

4. **Dans chaque système, configurez IKE pour gérer les AS IPsec.**

   Utilisez l'une des procédures décrites à la section “Configuration d'IKEv2” à la page 143. La syntaxe du fichier de configuration IKE est décrite à la page de manuel `ikev2.config`(4). Si vous communiquez avec un système qui ne prend en charge que le protocole IKEv1,

reportez-vous à la section “Configuration d'IKEv1” à la page 171 et à la page de manuel `ike.config`(4).

Remarque - Si vous devez générer et maintenir vos clés manuellement, reportez-vous à la section “Création manuelle de clés IPsec” à la page 125.

---

5. **Vérifiez la syntaxe du fichier de stratégie IPsec.**

```
% pfbash
# /usr/sbin/ipsecconf -c /etc/inet/ipsecinit.conf
```

Corrigez les éventuelles erreurs, vérifiez la syntaxe du fichier, puis continuez.

6. **Actualisez la stratégie IPsec.**

```
# svcadm refresh ipsec/policy:default
```

La stratégie IPsec est activée par défaut. *Actualisez*-la. Si vous avez désactivé la stratégie IPsec, activez-la.

```
# svcadm enable ipsec/policy:default
```

7. **Activez les clés pour IPsec.**

- **Si le service `ike` n'est pas activé, activez-le.**

Remarque - Si vous communiquez avec un système qui ne prend en charge que le protocole IKEv1, spécifiez l'instance `ike:default`.

```
# svcadm enable ipsec/ike:ikev2
```

- **Si le service `ike` est activé, redémarrez-le.**

```
# svcadm restart ike:ikev2
```

Si vous avez configuré les clés manuellement à l'Étape 4, suivez la procédure de la section “Création manuelle de clés IPsec” à la page 125 pour activer les clés.

8. **Assurez-vous que les paquets sont protégés.**

La procédure est décrite à la section “Vérification de la protection des paquets par IPsec” à la page 131.

Exemple 7-1 Configuration de la stratégie IPsec à distance via une connexion `ssh`

Dans cet exemple, l'administrateur ayant le rôle `root` configure la stratégie IPsec et des clés sur deux systèmes en utilisant la commande `ssh` pour atteindre le second système. L'administrateur

est défini à l'identique sur les deux systèmes. Pour plus d'informations, reportez-vous à la page de manuel ssh(1).

1. L'administrateur configure le premier système en effectuant les Étape 1 à Étape 5 de la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.
2. Dans une autre fenêtre de terminal, l'administrateur utilise le nom et l'ID utilisateur définis de façon identique pour se connecter à distance avec la commande ssh.

```
local-system % ssh -l jdoe other-system
other-system # su - root
Enter password: xxxxxxxx
other-system #
```

3. Dans la fenêtre de terminal de la session ssh, l'administrateur configure la stratégie IPsec et les clés du second système en effectuant les étapes Étape 1 à Étape 7.
4. L'administrateur met fin à la session ssh.

```
other-system # exit
local-system
# exit
```

5. L'administrateur active la stratégie IPsec sur le premier système en suivant les procédures des étapesÉtape 6 et Étape 7.

La prochaine fois que les deux systèmes communiquent, y compris par le biais d'une connexion ssh, la communication est protégée par IPsec.

Exemple 7-2 Configuration de la stratégie Ipsec pour exécution en mode FIPS 140

Dans cet exemple, l'administrateur configure la stratégie IPsec sur un système sur lequel FIPS 140 est activé pour qu'elle suive une stratégie de sécurité de site qui exige des algorithmes symétriques dont la longueur de clé est d'au moins 192 bits.

L'administrateur indique deux stratégies IPsec possibles. La première stratégie indique AES en mode CCM pour le chiffrement et l'authentification, et la seconde indique AES avec longueur de clé de 192 bits et 256 bits pour le chiffrement, et SHA384 pour l'authentification.

```
{laddr machine1 raddr machine2} ipsec {encr_algs aes-ccm(192...) sa shared} or ipsec
{laddr machine1 raddr machine2} ipsec {encr_algs aes(192...) encr_auth_algs sha2(384) sa
shared}
```

Utilisation d'IPsec pour protéger Web Server Communication avec d'autres serveurs

Sur un système qui s'exécute sur un serveur Web, vous pouvez utiliser l'ensemble du trafic IPsec à l'exception Web pour protéger les demandes des clients. En règle générale, le trafic réseau protégée entre le serveur Web et d'autres serveurs backend.

En plus d'autoriser des clients Web à contourner IPSec, la stratégie IPsec de cette procédure permet au serveur DNS d'effectuer des demandes client DNS. Tout autre trafic est protégé par IPsec.

Avant de commencer

Cette procédure part du principe que les étapes de la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111 qui permettent de configurer IPsec sur les deux serveurs ont été accomplies, de sorte que les conditions suivantes sont en vigueur :

- Chaque système est une zone globale ou une zone avec une exclusif Le multipathing sur réseau IP adresses fixe IP. Pour plus d'informations, reportez-vous à la section “IPsec et les zones Oracle Solaris” à la page 106.
- La communication avec le serveur Web est déjà protégée par IPsec.
- Les numéros de clés sont générés par IKE.
- Vous avez vérifié que les paquets sont protégés.

Un utilisateur disposant des droits spécifiques peut exécuter les commandes sans être utilisateur `root`.

- Pour exécuter les commandes de configuration, vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau).
- Pour modifier les fichiers système ayant trait à IPsec et créer des clés, vous utilisez la commande `pfedit`.
- Pour modifier le fichier `hosts`, vous devez être dans le rôle `root` ou avoir une autorisation explicite pour modifier le fichier.

Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, “Configuration de la stratégie IPsec à distance via une connexion `ssh`” et à “ Administration à distance de ZFS avec le shell sécurisé ” du manuel “ Gestion de l’accès au shell sécurisé dans Oracle Solaris 11.2 ” pour des instructions sur la connexion à distance sécurisée.

1. **Déterminez les services qui doivent ignorer les vérifications de stratégie IPsec.**

Pour un serveur Web, ces services incluent les ports TCP 80 (HTTP) et 443 (HTTP sécurisé). Si le serveur Web assure la recherche de noms DNS, le serveur doit peut-être inclure également le port 53 pour TCP et UDP.

2. **Ajoutez la stratégie relative au serveur Web au fichier de stratégie IPsec.**

   Ajoutez les lignes suivantes au fichier `ipsecinit.conf` :

```
# pfedit /etc/inet/ipsecinit.conf
...
# Web traffic that web server should bypass.
{lport  80 ulp tcp dir both} bypass {}
{lport 443 ulp tcp dir both} bypass {}

# Outbound DNS lookups should also be bypassed.
{rport 53 dir both} bypass {}

# Require all other traffic to use ESP with AES and SHA-2.
# Use a unique SA for outbound traffic from the port
{} ipsec {encr_algs aes encr_auth_algs sha512 sa shared}
```

   Cette configuration permet uniquement au trafic sécurisé d'accéder au système, avec les exceptions de contournement décrites à l'Étape 1.

3. **Vérifiez la syntaxe du fichier de stratégie IPsec.**

```
# ipsecconf -c /etc/inet/ipsecinit.conf
```

4. **Actualisez la stratégie IPsec.**

```
# svcadm refresh ipsec/policy
```

5. **Actualisez les clés pour IPsec.**

   Redémarrez le service `ike`.

```
# svcadm restart ike:ikev2
```

   Remarque - Si vous communiquez avec un système qui ne prend en charge que le protocole IKEv1, spécifiez l'instance `ike:default`.

   Si vous avez configuré les clés manuellement, suivez les instructions de la section “Création manuelle de clés IPsec” à la page 125.

   Votre installation est terminée.

6. **(Facultatif) Autorisez un système distant à communiquer avec le serveur Web pour le trafic non-Web.**

   Ajoutez les lignes suivantes dans un fichier `/etc/inet/ipsecinit.conf` stocké sur le système distant :

```
## Communicate with web server about nonweb stuff
##
{raddr webserver} ipsec {encr_algs aes encr_auth_algs sha512 sa shared}
```

Vérifiez la syntaxe, puis actualisez la statégie IPsec pour l'activer.

remote-system **# ipsecconf -c /etc/inet/ipsecinit.conf**
remote-system **# svcadm refresh ipsec/policy**

Un système distant peut communiquer de manière sécurisée avec le serveur Web pour le trafic non-Web uniquement lorsque les stratégies IPsec des systèmes sont identiques.

7. **(Facultatif) Affichez les entrées de stratégie IPsec, y compris les entrées définies par tunnel, dans l'ordre dans lequel les correspondances sont repérées.**

```
# ipsecconf -L -n
```

Protection d'un VPN à l'aide d'IPsec

Vous pouvez utiliser pour protéger un VPN IPsec. Pour des informations de fond, reportez-vous à “Modes Transport et Tunnel dans IPsec” à la page 101. Les exemples et procédures de cette section utilisent des adresses IPv4, mais les exemples et procédures s'appliquent également aux réseaux privés virtuels IPv6. Pour une courte explication, reportez-vous à la section “Protection du trafic réseau à l'aide d'IPsec” à la page 109.

Des exemples de stratégies IPsec en mode Tunnel sont présentés à la section “Protection d'un VPN à l'aide d'IPsec en mode Tunnel (exemples)” à la page 117.

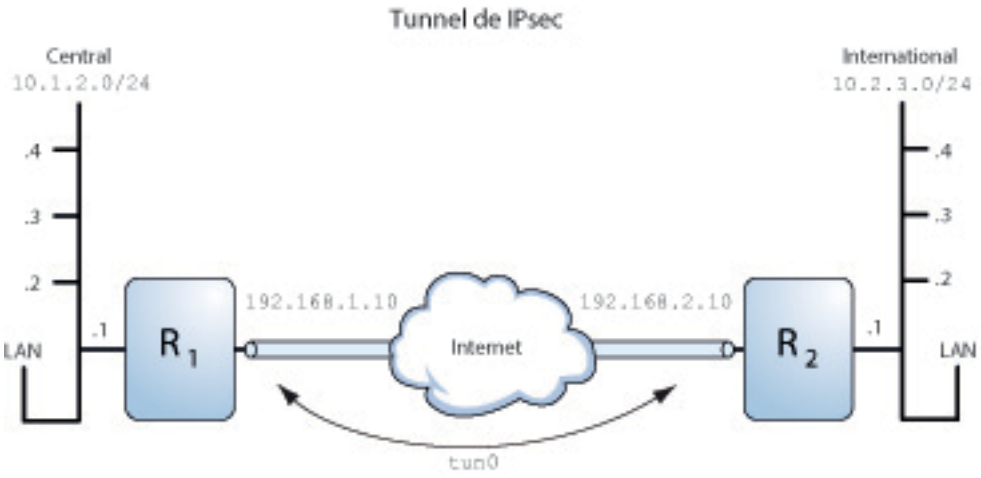
Protection d'un VPN à l'aide d'IPsec en mode Tunnel (exemples)

Le tunnel dans l'illustration suivante est configuré pour tous les sous-réseaux des LAN à l'aide des options suivantes :

```
## Tunnel configuration for ##
# Tunnel name is tun0
# Intranet point for the source is 10.1.2.1
# Intranet point for the destination is 10.2.3.1
# Tunnel source is 192.168.1.10
# Tunnel destination is 192.168.2.10

# Tunnel name address object is tun0/to-central
# Tunnel name address object is tun0/to-overseas
```

FIGURE 7-1 Tunnel protégé par IPsec



Les exemples suivants sont basés sur la figure ci-après .

EXEMPLE 7-3 Création d'un tunnel utilisable par tous les sous-réseaux

Dans cet exemple, la totalité du trafic des LAN locaux du LAN central de la Figure 7-1, “Tunnel protégé par IPsec” peut être mise en tunnel du routeur 1 au routeur 2, puis fournie à tous les LAN locaux du LAN Overseas. Le trafic est chiffré à l'aide d'AES.

```
## IPsec policy ##
{tunnel tun0 negotiate tunnel}
 ipsec {encr_algs aes encr_auth_algs sha512 sa shared}
```

EXEMPLE 7-4 Création d'un tunnel connectant deux sous-réseaux

Dans cet exemple, seul le trafic entre le sous-réseau `10.1.2.0/24` du LAN Central et le sous-réseau `10.2.3.0/24` du LAN Overseas est mis en tunnel et chiffré. En l'absence d'autres stratégies IPsec pour Central, si le LAN Central tente de transmettre des données pour d'autres LAN via ce tunnel, le trafic est abandonné au niveau du routeur 1.

```
## IPsec policy ##
{tunnel tun0 negotiate tunnel laddr 10.1.2.0/24 raddr 10.2.3.0/24}
 ipsec {encr_algs aes encr_auth_algs sha512 shared}
```

Description de la topologie réseau requise par les tâches IPsec afin de protéger un VPN

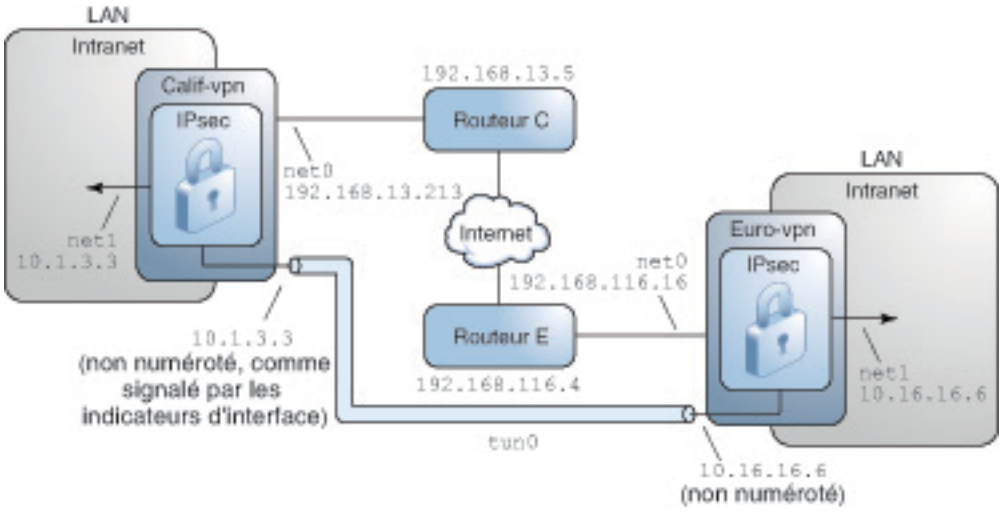
Les procédures de cette section partent de la configuration suivante. Le réseau est illustré dans la Figure 7-2, “Exemple de VPN entre plusieurs sites connectés à Internet”.

- Chaque système utilise un espace d'adressage IPv4.

  Ces procédures fonctionnent également avec les adresses IPv6 ou une combinaison d'adresses IPv4 et IPv6.
- Chaque système possède deux interfaces. L'interface `net0` se connecte à Internet. Dans cet exemple, les adresses IP Internet commencent par `192.168`. L'interface `net1` se connecte au LAN de la société, son intranet. Dans cet exemple, les adresses IP intranet commencent par le numéro `10`.
- Chaque système nécessite le chiffrement ESP avec l'algorithme AES. L'algorithme AES utilise une clé de 128 ou 256 bits.
- Chaque système nécessite l'authentification ESP avec l'algorithme SHA-2. Dans cet exemple, l'algorithme SHA-2 utilise une clé de 512 bits.
- Chaque système peut se connecter à un routeur bénéficiant d'un accès direct à Internet.
- Chaque système utilise des associations de sécurité partagées (SA, Security Associations).

L'illustration suivante montre les paramètres de configuration utilisées dans les procédures.

FIGURE 7-2 Exemple de VPN entre plusieurs sites connectés à Internet



Les paramètres de configuration sont répertoriées dans le tableau suivant.

Paramètres	Europe	Californie
Nom du système	`euro-vpn`	`calif-vpn`
Interface intranet du système	`net1`	`net1`
Adresse intranet du réseau, la route par défaut à l'autre	`10.16.16.6`	`10.1.3.3`
Objet d'adresse intranet du système	`net1/inside`	`net1/inside`
Interface Internet du système	`net0`	`net0`
Adresse Internet du système	`192.168.116.16`	`192.168.13.213`
Nom du routeur Internet	`router-E`	`router-C`
Adresse du routeur Internet	`192.168.116.4`	`192.168.13.5`
Nom du tunnel	`tun0`	`tun0`
Objet d'adresse de nom de tunnel	`tun0/v4tunaddr`	`tun0/v4tunaddr`

Pour plus d'informations sur les noms de tunnels, reportez-vous à la section " Administration des tunnels IP " du manuel " Administration des réseaux TCP/IP, d'IPMP et des tunnels IP dans Oracle Solaris 11.2 ". Pour plus d'informations sur les objets d'adresse, reportez-vous à la section " Configuration d'une interface IPv4 " du manuel " Configuration et administration des composants réseau dans Oracle Solaris 11.2 " et à la page de manuel `ipadm`(1M).

▼ Protection de la connexion entre deux réseaux locaux à l'aide d'IPsec en mode Tunnel

En mode Tunnel, le paquet IP interne détermine la stratégie IPsec qui protège son contenu.

Cette procédure est une extension de la procédure "Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec" à la page 111. La configuration est décrite à la section "Description de la topologie réseau requise par les tâches IPsec afin de protéger un VPN" à la page 119.

Pour une description plus détaillée des raisons pour lesquelles certaines commandes doivent être exécutées, reportez-vous aux étapes correspondantes à la section "Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec" à la page 111.

Remarque - Effectuez cette procédure sur les deux systèmes.

Outre la connexion de deux systèmes, vous connectez deux intranets qui leur sont connectés. Les systèmes de cette procédure fonctionnent comme des passerelles.

Remarque - Pour utiliser IPsec en mode Tunnel avec des étiquettes sur un système Trusted Extensions, reportez-vous aux étapes supplémentaires indiquées à la section " Configuration d'un tunnel au sein d'un réseau non autorisé " du manuel " Configuration et administration de Trusted Extensions ".

Avant de commencer

Chaque système est une zone globale ou une zone en mode IP exclusif. Pour plus d'informations, reportez-vous à la section “IPsec et les zones Oracle Solaris” à la page 106.

Un utilisateur disposant des droits spécifiques peut exécuter les commandes sans être utilisateur `root`.

- Pour exécuter les commandes de configuration, vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau).
- Pour modifier les fichiers système ayant trait à IPsec et créer des clés, vous utilisez la commande `pfedit`.
- Pour modifier le fichier `hosts`, vous devez être dans le rôle `root` ou avoir une autorisation explicite pour modifier le fichier.

Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, “Configuration de la stratégie IPsec à distance via une connexion `ssh`” et à “ Administration à distance de ZFS avec le shell sécurisé ” du manuel “ Gestion de l’accès au shell sécurisé dans Oracle Solaris 11.2 ” pour des instructions sur la connexion à distance sécurisée.

1. Contrôlez le flux de paquets avant de configurer IPsec.

a. Désactivez le transfert IP et le routage IP dynamique.

```
# routeadm -d ipv4-routing
# ipadm set-prop -p forwarding=off ipv4
# routeadm -u
```

La désactivation du transfert IP empêche le transfert de paquets d'un réseau vers un autre par l'intermédiaire de ce système. La commande `routeadm` est décrite à la page de manuel `routeadm`(1M).

b. Activez le multihébergement IP strict.

```
# ipadm set-prop -p hostmodel=strong ipv4
```

L'activation du multihébergement IP strict requiert que les paquets de l'une des adresses de destination du système arrivent à l'adresse de destination adéquate.

Lorsque le paramètre `hostmodel` est défini sur `strong`, les paquets arrivant sur une interface particulière doivent être adressés à l'une des adresses IP locales de cette interface. Tous les autres paquets sont abandonnés, même les paquets envoyés vers d'autres adresses locales du système.

c. Assurez-vous que la plupart des services réseau sont désactivés.

Assurez-vous que le service `ssh` est en cours d'exécution.

```
% svcs | grep network
…
online         Aug_09   svc:/network/ssh:default
```

2. **Ajoutez la stratégie IPsec pour le VPN dans le fichier `/etc/inet/ipsecinit.conf`.**

   Pour obtenir des exemples supplémentaires, reportez-vous à la section “Protection d'un VPN à l'aide d'IPsec en mode Tunnel (exemples)” à la page 117.

   Dans cette stratégie, la protection IPsec n'est pas requise entre les systèmes du réseau local et l'adresse IP interne de la passerelle, d'où l'ajout d'une déclaration `bypass`.

   a. **Sur le système `euro-vpn`, ajoutez l'entrée suivante au fichier `ipsecinit.conf` :**

   ```
   # LAN traffic to and from this host can bypass IPsec.
   {laddr 10.16.16.6 dir both} bypass {}

   # WAN traffic uses ESP with AES and SHA-2.
   {tunnel tun0 negotiate tunnel}
    ipsec {encr_algs aes encr_auth_algs sha512 sa shared}
   ```

   b. **Sur le système `calif-vpn`, ajoutez l'entrée suivante au fichier `ipsecinit.conf` :**

   ```
   # LAN traffic to and from this host can bypass IPsec.
   {laddr 10.1.3.3 dir both} bypass {}

   # WAN traffic uses ESP with AES and SHA-2.
   {tunnel tun0 negotiate tunnel}
    ipsec {encr_algs aes encr_auth_algs sha512 sa shared}
   ```

3. **Sur chaque système, configurez IKE afin d'ajouter une paire d'AS IPsec entre les deux systèmes.**

   Configurez IKE en suivant l'une des procédures de configuration décrites à la section “Configuration d'IKEv2” à la page 143. La syntaxe du fichier de configuration IKE est décrite à la page de manuel `ikev2.config`(4). Si vous communiquez avec un système qui ne prend en charge que le protocole IKEv1, reportez-vous à la section “Configuration d'IKEv1” à la page 171 et à la page de manuel `ike.config`(4).

   Remarque - Si vous devez générer et maintenir vos clés manuellement, reportez-vous à la section “Création manuelle de clés IPsec” à la page 125.

4. **Vérifiez la syntaxe du fichier de stratégie IPsec.**

   ```
   # ipsecconf -c /etc/inet/ipsecinit.conf
   ```

   Corrigez les éventuelles erreurs, vérifiez la syntaxe du fichier, puis continuez.

5. **Atualisez la stratégie IPsec.**

```
# svcadm refresh ipsec/policy
```

La stratégie IPsec est activée par défaut. *Actualisez*-la. Si vous avez désactivé la stratégie IPsec, activez-la.

```
# svcadm enable ipsec/policy
```

6. **Créez et configurez le tunnel, `tun0`.**

Les commandes suivantes configurent les interfaces internes et externes, créent le tunnel `tun0` et attribuent des adresses IP au tunnel.

a. **Sur le système `calif-vpn`, créez le tunnel et configurez-le.**

```
# ipadm create-ip net1
# ipadm create-addr -T static -a local=10.1.3.3 net1/inside
# dladm create-iptun -T ipv4 -a local=192.168.13.213,remote=192.168.116.16 tun0
# ipadm create-ip tun0
# ipadm create-addr -T static \
-a local=10.1.3.3,remote=10.16.16.6 tun0/v4tunaddr
```

La première commande crée l'interface IP `net1`. La seconde commande ajoute des adresses à `net1`. La troisième commande crée l'interface IP `tun0`. La quatrième commande ajoute les adresses IP sont encapsulés dans la liaison de tunnel. Pour plus d'informations, reportez-vous aux pages de manuel `dladm`(1M) et `ipadm`(1M).

b. **Sur le système `euro-vpn`, créez le tunnel et configurez-le.**

```
# ipadm create-ip net1
# ipadm create-addr -T static -a local=10.16.16.6 net1/inside
# dladm create-iptun -T ipv4 -a local=192.168.116.16,remote=192.168.13.213 tun0
# ipadm create-ip tun0
# ipadm create-addr -T static \
-a local=10.16.16.6,remote=10.1.3.3 tun0/v4tunaddr
```

Remarque - L'option `-T` de la commande `ipadm` spécifie le type d'adresse à créer. L'option `-T` de la commande `dladm` spécifie le tunnel.

Pour plus d'informations sur ces commandes, reportez-vous aux pages de manuel `dladm`(1M) et `ipadm`(1M) ainsi qu'à la section " Configuration d'une interface IPv4 " du manuel " Configuration et administration des composants réseau dans Oracle Solaris 11.2 ". Pour plus d'informations sur les noms personnalisés, reportez-vous à la section " Résolution de noms de périphériques réseau et des liaisons de données dans Oracle Solaris " du manuel " Configuration et administration des composants réseau dans Oracle Solaris 11.2 ".

7. **Sur chaque système, configurez le transfert.**

```
# ipadm set-ifprop -m ipv4 -p forwarding=on net1
# ipadm set-ifprop -m ipv4 -p forwarding=on tun0
# ipadm set-ifprop -m ipv4 -p forwarding=off net0
```

Le transfert IP signifie que les paquets arrivant peuvent être transférés. Le transfert IP signifie également que les paquets quittant l'interface peuvent provenir d'un autre emplacement. Pour que le transfert de paquet s'effectue sans erreur, vous devez activer le transfert IP à la fois sur l'interface réceptrice et sur l'interface émettrice.

Etant donné que l'interface `net1` se trouve *dans* l'intranet, le transfert IP doit être activé pour `net1`. Comme `tun0` connecte les deux systèmes via Internet, le transfert IP doit être activé pour `tun0`. Le transfert IP de l'interface `net0` est désactivé afin d'éviter toute injection de paquets par un concurrent *externe* sur Internet dans l'intranet protégé.

8. **Sur chaque système, empêchez la publication de l'interface privée.**

```
# ipadm set-addrprop -p private=on net0
```

Même si le transfert IP de `net0` est désactivé, l'implémentation d'un protocole de routage peut permettre d'annoncer l'interface. Par exemple, le protocole `in.routed` peut encore annoncer que `net0` est disponible pour transférer des paquets à ses homologues dans l'intranet. Pour éviter ces annonces, définissez l'indicateur `private` de l'interface.

9. **Redémarrez les services réseau.**

```
# svcadm restart svc:/network/initial:default
```

10. **Ajoutez manuellement une route par défaut sur l'interface `net0`.**

La route par défaut doit correspondre à un routeur bénéficiant d'un accès direct à Internet.

a. **Sur le système `calif-vpn`, ajoutez la route suivante :**

```
# route -p add net default 192.168.13.5
```

b. **Sur le système `euro-vpn`, ajoutez la route suivante :**

```
# route -p add net default  192.168.116.4
```

Même si l'interface `net0` ne fait pas partie de l'intranet, `net0` n'a pas besoin de passer par Internet pour atteindre le système homologue. Pour trouver son homologue, `net0` requiert des informations sur le routage Internet. Pour le reste d'Internet, le système VPN apparaît comme étant un hôte, non un routeur. Par conséquent, vous pouvez utiliser un routeur par défaut ou exécuter le protocole de recherche de routeur pour rechercher le système. Pour plus d'informations, reportez-vous aux pages de manuel `route`(1M) et `in.routed`(1M).

Tâches IPsec supplémentaires

La liste des tâches suivantes répertorie des tâches utiles dans le cadre de la gestion d'IPsec.

TABLEAU 7-2 Liste des tâches IPsec supplémentaires

Tâche	Description	Pour obtenir des instructions
Les SA IPsec de création ou de remplacement manuellement.	Fournit les données brutes pour les SA IPsec :	“Création manuelle de clés IPsec” à la page 125
Créez un rôle de sécurité réseau.	Crée un rôle pouvant configurer un réseau sécurisé, mais possédant moins de permissions que le rôle `root`.	“Configuration d'un rôle pour la sécurité réseau” à la page 128
Créez un profil de droits qui peut gérer tous les tâches de gestion réseau.	Crée un rôle pouvant effectuer mais possédant moins de permissions de gestion réseau que le rôle `root`.	Exemple 7-7, “Octroi à un utilisateur de confiance de l'autorisation de configurer et gérer IPsec”
Vérification de la protection des paquets par IPsec	Recherche des en-têtes spécifiques indiquant la méthode de protection des paquets IP dans la sortie de commande `snoop`.	“Vérification de la protection des paquets par IPsec” à la page 131
Gestion d'IPsec et des numéros de clés en tant qu'ensemble de services SMF	Désactive, permet d'exécuter et redémarre des services, le serveur Oracle Beehive Server actualise. Modifie également les valeurs de propriété des services.	“Affichage des propriétés d'IPsec et du service de clés manuel” à la page 214

▼ Création manuelle de clés IPsec

La procédure ci-dessous présente les clés IPsec pour le moment où vous n'utilisez pas uniquement pour la gestion des clés IKE.

Les AS IPsec qui sont ajoutées par le biais de la commande `ipseckey` ne sont pas conservées après la réinitialisation du système. Pour les AS IPsec persistantes, ajoutez des entrées au fichier `/etc/inet/secret/ipseckeys`.

Attention - Si vous devez opter pour une génération manuelle de clés prudence afin de garantir, prenez les précautions suivantes avant que les clés que vous générez sont sécurisés. Il s'agit des clés réelles utilisé pour sécuriser les données.

Avant de commencer

La gestion manuelle des numéros de clé dans une zone IP partagée s'effectue dans la zone globale. Zone pour IP exclusif, vous configurez les informations relatives à la génération de zone qui IP exclusif Le multipathing sur réseau IP dans.

Vous devez prendre le rôle `root`. Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

1. **Générez les clés pour les SA IPsec.**

Les clés doivent prendre en charge une stratégie spécifique dans le fichier `ipsecinit.conf`. Par exemple, vous pouvez utiliser la stratégie de la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111 :

```
{laddr enigma raddr partym} ipsec {encr_algs aes encr_auth_algs sha512 sa shared}
```

Cette stratégie utilise les algorithmes AES et SHA-2

a. Déterminez les clés nécessaires.

Vous devez générer les clés pour `aes`, `sha512` et l'index de paramètres de sécurité index du paramètre de sécurité de l'AS :

- Deux numéros aléatoires hexadécimaux comme valeur du SPI. un numéro pour le trafic sortant et un numéro pour le trafic entrant. Chaque numéro peut comporter huit caractères maximum.
- Deux numéros aléatoires hexadécimaux pour l'algorithme d'authentification SHA-2. Chacun des numéros doit comporter 512 caractères. L'un d'eux est dédié à `dst enigma`, l'autre à `dst partym`.
- Deux numéros aléatoires hexadécimaux pour l'algorithme de chiffrement AES. Chacun des numéros doit comporter 128 caractères. L'un d'eux est dédié à `dst enigma`, l'autre à `dst partym`.

Remarque - La commande `ipsecalgs -l` affiche les tailles de clé des algorithmes. Suivez la procédure ci-dessous lors de l'utilisation de clés manuelles, c'est-à-dire les algorithmes AES SHA512 et utilisez la. N'utilisez pas de type "Weak Identity Map" algorithmes des algorithmes, les ou le mode combiné des algorithmes pour clés manuelles GMAC.

---

b. Générez les clés requises.

- Si un générateur de nombres aléatoires est disponible sur votre site, utilisez-le.
- Utilisez la commande `pktool` comme indiqué dans la section “ Génération d’une clé symétrique à l’aide de la commande pktool ” du manuel “ Gestion du chiffrement et des certificats dans Oracle Solaris 11.2 ” et l'exemple IPsec de cette section.

2. Ajoutez les clés au fichier de clés manuelles pour IPsec.

a. Modifiez le fichier `/etc/inet/secret/ipseckeys` sur le système `enigma` de manière qu'il se présente comme suit :

```
## ipseckeys - This file takes the file format documented in
##  ipseckey(1m).
#   Note that naming services might not be available when this file
#   loads, just like ipsecinit.conf.
#
#   Backslashes indicate command continuation.
#
```

```
# for outbound packets on enigma
add esp spi 0x8bcd1407 \
   src 192.168.116.16 dst 192.168.13.213  \
   encr_alg aes \
   auth_alg sha512  \
   encrkey  d41fb74470271826a8e7a80d343cc5aa... \
   authkey  e896f8df7f78d6cab36c94ccf293f031...
#
# for inbound packets
add esp spi 0x122a43e4 \
   src 192.168.13.213 dst 192.168.116.16 \
   encr_alg aes \
   auth_alg sha512  \
   encrkey dd325c5c137fb4739a55c9b3a1747baa... \
   authkey ad9ced7ad5f255c9a8605fba5eb4d2fd...
```

b. Protégez le fichier à l'aide d'autorisations de lecture seule.

```
# chmod 400 /etc/inet/secret/ipseckeys
```

Si vous avez utilisé la commande `pfedit -s` pour créer le fichier `ipseckeys`, les autorisations sont définies correctement. Pour plus d'informations, reportez-vous à la page de manuel `pfedit`(1M).

c. Vérifiez la syntaxe du fichier.

```
# ipseckey -c /etc/inet/secret/ipseckeys
```

Remarque - Les clés utilisées sur les deux systèmes *doivent* être identiques.

3. Activez les clés pour IPsec.

- **Si le service `manual-key` n'est pas activé, activez-le.**

```
% svcs manual-key
STATE          STIME    FMRI
disabled       Apr_10   svc:/network/ipsec/manual-key:default
# svcadm enable ipsec/manual-key
```

- **Si le service `manual-key` est activé, actualisez-le.**

```
# svcadm refresh ipsec/manual-key
```

Étapes suivantes Si vous n'avez pas terminé d'établir la stratégie IPsec, effectuez de nouveau la procédure IPsec pour activer ou actualiser la stratégie IPsec. Pour obtenir des exemples de la stratégie IPsec de protection de VPN, reportez-vous à “Protection d'un VPN à l'aide d'IPsec” à la page 117. Pour plus d'exemples de la stratégie IPsec, reportez-vous à la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

▼ Configuration d'un rôle pour la sécurité réseau

Si vous administrez vos systèmes à l'aide des fonctionnalités de droits d'Oracle Solaris, suivez cette procédure pour générer un rôle de gestion ou de sécurité du réseau.

Avant de commencer Vous devez prendre le rôle `root` pour créer et affecter un rôle. Les utilisateurs standard peuvent répertorier et afficher le contenu des profils de droits disponibles.

1. Répertoriez les profils de droits disponibles relatifs au réseau.

```
% getent prof_attr | grep Network | more
...
Network Management:RO::Manage the host and network configuration...
Network Security:RO::Manage network and host security...:profiles=Network Wifi
Security,Network Link Security,Network IPsec Management...
Network Wifi Management:RO::Manage wifi network configuration...
Network Wifi Security:RO::Manage wifi network security...
Network Link Security:RO::Manage network link security...
Network IPsec Management:RO::Manage IPsec and IKE...
System Administrator:RO::Can perform most non-security administrative tasks:
profiles=...Network Management...
Information Security:RO::Maintains MAC and DAC security policies:
profiles=...Network Security...
```

Le profil de gestion du réseau est un profil supplémentaire inclus dans le profil d'administrateur système. Si vous avez attribué le profil de droits d'administrateur système à un rôle, alors ce dernier permet d'exécuter les commandes définies dans le profil de gestion du réseau.

2. Répertoriez les commandes dans le profil de droits Network Management.

```
% profiles -p "Network Management" info
...
cmd=/usr/sbin/dladm
cmd=/usr/sbin/dlstat
...
cmd=/usr/sbin/svcadm
cmd=/usr/sbin/svccfg
cmd=/usr/sbin/dumpcap
```

3. Choisissez l'étendue des rôles de sécurité réseau sur votre site.

Basez votre choix sur les profils de droits définis au cours de l'Étape 1.

- Pour créer un rôle qui gère l'ensemble de la sécurité du réseau, utilisez le profil de droits Network Security.
- Pour créer un rôle qui gère IPsec et IKE uniquement, utilisez le profil de droits Network IPsec Management.
- Pour créer un rôle qui gère la gestion du réseau et la sécurité, utilisez les Network Security ou Network IPsec Management Profil de droits, en plus du profil de gestion réseau.

4. **Créez le rôle et assignez-le à un ou plusieurs utilisateurs.**

   Pour connaître la procédure à suivre, reportez-vous à la section " Création d'un rôle " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 " and Exemple 7-7, "Octroi à un utilisateur de confiance de l'autorisation de configurer et gérer IPsec".

Exemple 7-5 Création et attribution d'un rôle Network Management and Security

Dans cet exemple, l'administrateur affecte à un rôle, Network Management deux et profils de droits Network Security. Ensuite, l'administrateur attribue le rôle à un utilisateur de confiance.

```
# roleadd -c "Network Mgt and Security" \
-S ldap -K profiles="Network Management Plus" netmgtsec
# passwd netmgtsec
New Password: xxxxxxxx
Confirm password: xxxxxxxx
# usermod -R netmgtsec jdoe
```

L'utilisateur `jdoe` a accès aux droits des profils une fois qu'il a pris le rôle `netmgtsec`.

```
% su - netsecmgt
Password: xxxxxxxx
#
```

Exemple 7-6 Répartition des responsabilités de sécurité réseau entre les rôles

Dans cet exemple, l'administrateur répartit les responsabilités de sécurité réseau entre deux rôles. Un rôle peut administrer la sécurité des connexions Wi-Fi et des liens et un autre rôle administrer IPsec et IKE. Chaque rôle est assigné à trois personnes, une personne par période de travail.

Ces rôles sont créés par l'administrateur comme suit :

1. L'administrateur nomme le premier rôle LinkWifi.
2. L'administrateur attribue au rôle les profils de droits Network Wifi, Network Link Security et Network Management.
3. L'administrateur attribue le rôle LinkWifi aux utilisateurs appropriés.
4. L'administrateur nomme le deuxième rôle Administrateur IPsec.
5. L'administrateur attribue au rôle les profils de droits Network IPsec Management et Network Management.
6. L'administrateur attribue le rôle d'administrateur IPsec aux utilisateurs appropriés.

Exemple 7-7 Octroi à un utilisateur de confiance de l'autorisation de configurer et gérer IPsec

Dans cet exemple, l'administrateur attribue à un utilisateur la responsabilité de configurer et de gérer IPsec.

En plus des profils de droits Network Management (gestion du réseau) et IPsec Network Management (gestion du réseau IPsec), l'administrateur donne à l'utilisateur la possibilité de modifier le fichier `hosts` et d'afficher les journaux.

1. L'administrateur crée deux profils de droits : l'un pour la modification de fichiers et l'autre pour la lecture de journaux.

```
# profiles -p -S LDAP "Hosts Configuration"
profiles:Network Configuration> set desc="Edits root-owned network files"
...Configuration> add auth=solaris.admin.edit/etc/hosts
...Configuration> commit
...Configuration> end
...Configuration> exit

# profiles -p -S LDAP "Read Network Logs"
profiles:Read Network Logs> set desc="Reads root-owned network log files"
...Logs> add cmd=/usr/bin/more
...Logs:more>set privs={file_dac_read}:/var/user/ikeuser/*
...Logs:more>set privs={file_dac_read}:/var/log/ikev2/*
...Logs:more> set privs={file_dac_read}:/etc/inet/ike/*
...Logs:more> set privs={file_dac_read}:/etc/inet/secret/*
...Logs:more>end
...Logs> add cmd=/usr/bin/tail
...Logs:tail>set privs={file_dac_read}:/var/user/ikeuser/*
...Logs:tail>set privs={file_dac_read}:/var/log/ikev2/*
...Logs:tail>set privs={file_dac_read}:/etc/inet/ike/*
...Logs:tail> set privs={file_dac_read}:/etc/inet/secret/*
...Logs:tail>end
...Logs> add cmd=/usr/bin/page
...Logs:page>set privs={file_dac_read}:/var/user/ikeuser/*
...Logs:page>set privs={file_dac_read}:/var/log/ikev2/*
...Logs:page>set privs={file_dac_read}:/etc/inet/ike/*
...Logs:page> set privs={file_dac_read}:/etc/inet/secret/*
...Logs:page>end
...Logs> exit
```

   Le profil de droits permet à l'utilisateur d'utiliser les commandes `more`, `tail` et `page` pour afficher les journaux. Les commandes `cat` et `head` ne peuvent pas être utilisées.

2. L'administrateur crée le profil de droits qui permet à l'utilisateur d'effectuer toutes les tâches de configuration et de gestion d'IPsec et de ses services de génération de clés.

```
# profiles -p "Site Network Management"
profiles:Site Network Management> set desc="Handles all network files and logs"
...Management> add profiles="Network Management"
...Management> add profiles="Network IPsec Management"
...Management> add profiles="Hosts Configuraton"
```

```
...Management> add profiles="Read Network Logs"
...Management> commit; end; exit
```

3. L'administrateur crée un rôle pour le profil, lui attribue un mot de passe, puis attribue le rôle à un utilisateur de confiance qui connaît les réseaux et la sécurité.

```
# roleadd -S LDAP -c "Network Management Guru" \
-m -K profiles="Site Network Management" netadm
# passwd netadm
Password: xxxxxxxx
Confirm password: xxxxxxxx
# usermod -S LDAP -R +netadm jdoe
```

4. Hors bande, l'adminitrateur donne le mot de passe du rôle à `jdoe`.

▼ Vérification de la protection des paquets par IPsec

Pour vérifier que les paquets sont protégés, testez la connexion à l'aide de la commande `snoop`. Les préfixes suivants peuvent apparaître dans la sortie `snoop` :

- Le préfixe `AH:` indique que AH protège les en-têtes. Ce préfixe s'affiche si le trafic est protégé à l'aide d'`auth_alg`.
- Le préfixe `ESP:` indique le transfert de données chiffrées. Ce préfixe s'affiche si le trafic est protégé à l'aide d'`encr_auth_alg` ou `encr_alg`.

Avant de commencer

Vous devez avoir accès aux deux systèmes afin de tester la connexion.

Vous devez prendre le rôle `root` pour créer la sortie `snoop`. Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

1. **Sur un système, par exemple `partym`, prenez le rôle `root`.**

```
% su -
Password: xxxxxxxx
#
```

2. **(Facultatif) Afficher les détails concernant l'EdC.**

```
# ipseckey dump
```

SPI cette sortie indique les valeurs correspondent aux EdC qui sont utilisés, lequel les algorithmes qui étaient utilisées, les clés localisées, et ainsi de suite.

3. **Sur ce système, préparez l'analyse des paquets à l'aide de la commande snoop à partir d'un système distant.**

Dans une fenêtre de terminal sur `partym`, analysez les paquets du système `enigma`.

```
# snoop -d net0 -o /tmp/snoop_capture enigma
Using device /dev/e1000g (promiscuous mode)
```

4. **Envoyez un paquet à partir du système distant.**

   Dans une autre fenêtre de terminal, connectez-vous à distance au système `enigma`. Entrez votre mot de passe. Ensuite, prenez le rôle `root` et envoyez un paquet à partir du système `enigma` au système `partym`. Le paquet doit être capturé à l'aide de la commande `snoop -v enigma`.

```
partym% ssh enigma
Password: xxxxxxxx
enigma% su -
Password: xxxxxxxx
enigma# ping partym
```

5. **Examinez la sortie de la commande `snoop`.**

```
partym# snoop -i /tmp.snoop_capture -v
```

   Vous pouvez aussi charger la sortie de `snoop` dans l'application Wireshark. Pour plus d'informations, reportez-vous à “Préparation des systèmes IPsec et IKE pour la résolution de problèmes” à la page 206 et “IPsec et commande `snoop`” à la page 227.

   Dans le fichier, la sortie devrait contenir les informations `AH` et `ESP` après les informations d'en-tête `IP` initiales. Les informations `AH` et `ESP` semblables à l'exemple ci-dessous indiquent que les paquets sont protégés :

```
IP:   Time to live = 64 seconds/hops
IP:   Protocol = 51 (AH)
IP:   Header checksum = 4e0e
IP:   Source address = 192.168.116.16, enigma
IP:   Destination address = 192.168.13.213, partym
IP:   No options
IP:
AH:  ----- Authentication Header -----
AH:
AH:  Next header = 50 (ESP)
AH:  AH length = 4 (24 bytes)
AH:  <Reserved field = 0x0>
AH:  SPI = 0xb3a8d714
AH:  Replay = 52
AH:  ICV = c653901433ef5a7d77c76eaa
AH:
ESP:  ----- Encapsulating Security Payload -----
ESP:
ESP:  SPI = 0xd4f40a61
ESP:  Replay = 52
ESP:     ....ENCRYPTED DATA....

ETHER:  ----- Ether Header -----
...
```

♦ ♦ ♦ CHAPITRE 8

A propos d'Internet Key Exchange

Le protocole IKE (Internet Key Exchange, échange de clé Internet) automatise la gestion des clés pour IPsec. Ce chapitre contient les informations suivantes sur le protocole IKE :

Pour obtenir des instructions sur l'implémentation de la dernière version du protocole IKE, reportez-vous au Chapitre 9, Configuration d'IKEv2. Pour continuer à utiliser IKEv1, reportez-vous au Chapitre 10, Configuration d'IKEv1. Pour obtenir des informations de référence, reportez-vous au Chapitre 12, IPsec et référence de gestion des clés Pour plus d'informations sur les protocoles IPsec, reportez-vous au Chapitre 6, A propos de l'architecture de sécurité IP.

Introduction au protocole IKE

La gestion des numéros de clé des associations de sécurité (Security Associations, SA) pour IPsec est appelée *gestion des clés*. La gestion automatique des clés requiert un canal de communication sécurisé pour la création, l'authentification et l'échange des clés. Oracle Solaris utilise IKE pour automatiser la gestion des clés. Le protocole IKE simplifie l'administration et élimine les risques de sécurité liés à la distribution manuelle des clés secrètes.

IKE peut profiter de l'accélération de chiffrement matérielle et du stockage de clés disponible. Les accélérateurs de chiffrement matériels permettent de gérer les opérations de clés intensives hors du système. Le stockage des clés sur des composants matériels fournit une couche de protection supplémentaire.

Oracle Solaris prend en charge deux versions du protocole IKE.

- IKE version 2 (IKEv2), qui est basé sur Internet Key Exchange Protocol version 2 (IKEv2), RFC 5996
- IKE version 1 (IKEv1), qui est basé sur The Internet Key Exchange (IKE), RFC 2409

Concepts et terminologie du protocole IKE

Les conditions et termes suivants sont communs aux deux versions d'IKE. Ils peuvent être implémentés différemment dans les deux versions.

- **Négociation et échange de clés –** Echange de clés et authentification sécurisée de l'identité de l'homologue. Le processus utilise des algorithmes cryptographiques asymétriques. Les principales méthodes utilisées sont les protocoles RSA et Diffie-Hellman.

  IKE crée et gère les SA IPsec entre les systèmes qui exécutent un démon IKE. IKE négocie un canal sécurisé qui protège la transmission des numéros de clé. Le démon crée les clés à partir d'un générateur de nombres aléatoires à l'aide du périphérique `/dev/random`. Le démon modifie les clés à une fréquence qui peut être configurée. Les numéros de clé sont accessibles aux algorithmes spécifiés dans le fichier de configuration `ipsecinit.conf` de la stratégie IPsec.
- **Authentification Diffie-Hellman (DH) –** Algorithme d'échange de clés qui permet à deux systèmes de générer une clé secrète partagée de manière sécurisée sur un canal non sécurisé.
- **Algorithme RSA –** Un algorithme asymétrique utilisé pour authentifier l'identité de systèmes homologues, généralement en prouvant l'identification du propriétaire d'un certificat X.509. Cet algorithme porte le nom de ses trois créateurs : Rivest, Shamir et Adleman.

  Les algorithmes DSA ou ECDSA peuvent également être utilisés à cette fin.
- **Confidentialité persistante parfaite (PFS, Perfect forward secrecy) –** Avec la fonction PFS, la clé visant à protéger la transmission des données n'est pas utilisée pour dériver d'autres clés. Il en est de même pour la source de la clé. Par conséquent, PFS peut empêcher le déchiffrement du trafic précédemment enregistré.
- **Groupe Oakley –** Utilisé pour négocier la PFS. Reportez-vous à la section 6 de la RFC The Internet Key Exchange (IKE).
- **Fichier de stratégie IKE –** Le jeu de règles d'IKE qui définit les paramètres acceptables à utiliser par un démon IKE lorsqu'il tente de créer un canal d'échange sécurisé avec un système homologue. Il s'agit d'un IKE SA dans IKEv2 ou Phase 1 dans IKEv1.

  Les paramètres incluent des algorithmes, des tailles de clé, des groupes Oakley et la méthode d'authentification. Les démons IKE Oracle Solaris prennent en charge les clés prépartagées et les certificats en tant que méthodes d'authentification.

Fonctionnement d'IKE

Un système exécutant un démon IKE peut négocier les paramètres nécessaires pour créer une association de sécurité (SA) entre ce système et un autre système exécutant le démon IKE. Le protocole utilisé pour négocier cette SA et les SA IPsec suivantes est appelé IKE. Cette version d'Oracle Solaris prend en charge la version 1 (IKEv1) et la version 2 (IKEv2) du protocole IKE.

L'association de sécurité IKE (également appelée ISAKMP ou Phase 1 SA dans IKEv1) sécurise d'autres échanges de protocole entre ces deux systèmes IKE. Ces échanges négocient

les algorithmes cryptographiques, la stratégie IPsec, ainsi que d'autres paramètres nécessaires à la création des SA IPsec.

Les systèmes exécutant un démon IKE peuvent également être configurés pour négocier des SA IPsec pour le compte d'autres systèmes. Les systèmes configurés de cette manière sont appelés *passerelles de sécurité*. Si la négociation IKE s'est déroulée avec succès, les SA IPsec permettent de protéger les paquets du réseau.

Remarque - Dans Oracle Solaris 11.2, IKEv2 utilise des algorithmes cryptographiques à partir de la structure cryptographique qui sont validés pour FIPS 140-2, niveau 1, à l'inverse de IKEv1. Par défaut, FIPS 140 n'est pas activé. Pour une comparaison des fonctionnalités entre ces deux versions, reportez-vous à “Comparaison d'IKEv2 et IKEv1” à la page 138. Pour activer le mode FIPS 140-2, reportez-vous à “ Création d’un environnement d’initialisation compatible FIPS 140 ” du manuel “ Gestion du chiffrement et des certificats dans Oracle Solaris 11.2 ”.

Si l'exigence stricte est de n'utiliser qu'une cryptographie validée FIPS 140-2, vous devez exécuter la version Oracle Solaris 11.1 SRU 5.5 ou la version Oracle Solaris 11.1 SRU 3. Oracle a obtenu la validation FIPS 140-2 relative à la structure cryptographique dans ces deux versions spécifiques. Oracle Solaris 11.2 s'appuie sur cette base validée pour apporter des améliorations logicielles concernant la performance, les fonctionnalités et la fiabilité. Configurez Oracle Solaris 11.2 chaque fois que possible en mode FIPS 140-2 pour bénéficier de ces améliorations.

Les paramètres négociés pour créer le SA IKE incluent les algorithmes cryptographiques qui protègent les échanges IKE et le matériel d'authentification. Le matériel d'authentification est utilisé pour déterminer si les paquets contenant les échanges de protocole IKE sont de confiance. Autrement dit, si les paquets proviennent d'un système approuvé et non pas d'un système se faisant passer pour ce système.

Oracle Solaris prend en charge deux types de supports d'authentification pour IKE, les clés prépartagées et les certificats de clés publiques.

IKE avec l'authentification des clés prépartagées

Une clé prépartagée est une chaîne de caractères hexadécimaux ou ASCII uniquement connue des deux systèmes IKE. Les clés sont appelées *prépartagées* car les deux extrémités doivent connaître la valeur de la clé avant l'échange IKE. Cette clé doit faire partie de la configuration IKE sur les deux systèmes. La clé prépartagée est utilisée pour la génération de charges IKE, qui constituent les paquets qui implémentent le protocole IKE. Le système qui traite ces charges IKE utilise la même clé pour authentifier les charges qu'il reçoit.

La clé prépartagée IKE n'est pas échangée entre les points d'extrémité IKE en utilisant le protocole IKE. En règle générale, la clé est partagée avec le système homologue via un autre moyen, tel qu'un appel téléphonique.

La clé prépartagée des homologues utilisant cette authentification doit être identique. Les clés sont stockés dans un fichier sur chacun des systèmes.

IKE avec certificats de clés publiques

Sécurisation des certificats de clés publiques et leur numériquement chaînes fournissent un mécanisme permettant d'identifier un système n'importe quel sans avoir à manuellement de change secret informations. Par conséquent, les certificats de clés publiques sont plus sûre que des clés prépartagées.

Un certificat de clé publique de données qui est un encode un BLOB en cas de présence de valeur de clé publique, certaines informations concernant la génération du certificat, comme son nom et un hachage qui l'a signé ou de la somme de contrôle du, le certificat et une signature numérique du dièse. Ensemble, ces valeurs constituent le certificat. La signature numérique garantit que le certificat n'a pas été modifié.

Une clé publique est une valeur mathématiquement dérivée d'une autre valeur, la *clé privée*. Qui permet d'obtenir la mathématique clé publique algorithme la clé privée rend l'extraction à partir de la clé privée à partir de la clé publique irréalisables. Par conséquent, certificats de clés publiques peuvent être gratuitement partagé. Parmi les algorithmes utilisés pour dériver des clés publiques, on peut citer RSA et les courbes elliptiques.

Une signature numérique est le résultat du traitement du contenu d'un certificat par un algorithme de signature tel que RSA, DSA ou ECDSA. Ces algorithmes, utiliser une clé de signature privée qui ne fait pas partie du certificat, et de produire une signature numérique. La signature est ajoutée au certificat. Là encore, le calcul de la clé de signature à partir du le contenu d'un certificat et la signature n'est pas pratique. D'autres informations pour ce point, la signature de certificat et que, par conséquent, les peuvent être facilement vérifier le contenu d'un certificat à l'aide d'un valeur, qui a été dérivée de la clé de signature.

Un certificat peut être auto-signé , auquel cas la signature du certificat peut-être vérifié par la clé publique du certificat, ou il peut être signée à l'aide d'une autre entité. Lorsqu'un autre entité signe le certificat, la valeur de la clé publique utilisé pour vérifier le certificat est également distribué sous forme de certificat de clé publique. Ce deuxième certificat sera signé par une autorité de certification (AC) de confiance ou par un intermédiaire. L'entité de signature fait confiance, en dernière analyse, à l'intermédiaire, c'est-à-dire l'AC root ou l'ancre sécurisée.

Ces certificats à clés publiques et les composants des structures, ainsi que les composants qui les implémentent les procédures, sont souvent appelés infrastructure à clé publique (PKI). La portée d'une PKI de l'organisation peut varier. Peut résider dans une simple une PKI permettant de signer les certificats CA pour utilisation locaux quelques. Vous pouvez saisir une grande diversité de PKI un est d'utiliser une en tant que globalement faisant autorité reconnu CA ancre sécurisée.

Utilisation de certificats de clés publiques dans IKE

Cette section décrit les étapes générales à créer et à utiliser dans IKE les certificats de clés publiques. Les procédures spécifiques sont décrites aux sections “Configuration d'IKEv2 avec des clés prépartagées” à la page 144 et “Configuration d'IKEv1 avec des clés prépartagées” à la page 172.

1. Si vous souhaitez utiliser un certificat autosigné ou un certificat issu d'une autorité de certification (CA), vous devez d'abord générer une paire de clés publique / privée
   - Pour un certificat autosigné, les homologues IKE échangent ensuite ces certificats, vérifient hors bande que les certificats sont authentiques, puis importent les certificats des homologues dans le keystore local. Keystore contient alors la le certificat auto-signé d'origine plus les certificats importés.
   - Pour les certificats d'une AC, il faut réaliser encore quelques opérations. Lorsque vous générez une paire clé publique / clé privée, vous pouvez également générer une CSR (certificate signing request). Une CSR contient la clé publique et les *identificateurs*. Un identificateur typique est un nom distinctif (ND), par exemple :

     ```
     DN=”O=Example\, Inc, OU=qa, L=Silicon Valley, ST=CA, CN=enigma”
     ```

Astuce - Créez un ND ou un autre identificateur aussi spécifique que possible afin de réduire les risques que votre identificateur soit identique à celui d'un autre certificat.

---

2. Envoyez la demande de signature de certificat à votre autorité de certification (CA).

   Dans un processus type, vous devez coller la demande de signature de certificat dans un formulaire Web et soumettre ce dernier à votre autorité de certification. L'autorité de certification peut vous envoyer plusieurs certificats.
3. Obtenez les certificats signés auprès de l'autorité de certification (CA), puis importez-les dans votre keystore IKEv2 ou base de données IKEv1.

   Vous devez importer tous les certificats que la CA a envoyés. Ces certificats se composent d'une "chaîne de confiance" à partir de la CA ancre sécurisée ou pour l'utilisateur root chacun identifiés, à votre certificat signé.
4. IKE répétez la procédure sur un pair.
5. Les règles IKE utiliser les certificats dans.

   Vous indiquer le certificat par un DN, tel qu'un identificateur. Les certificats CA-signé pour, vous pouvez configurer IKE pour accepter tout certificat CA est signée par une particulière.

Gestion des certificats révoqués

Un certificat signé est de confiance et valide car l'autorité de signature assure sa validité. Si un certificat est découvert ou déterminé est incorrect, le CA en tant que le révoquer.

Les AC conservent une liste de certificats révoqués, souvent appelée liste des certificats révoqués (CRL). Vous pouvez utiliser le protocole OCSP (Online Certificate Status Protocol) pour vérifier de manière dynamique le statut d'un certificat. Certains certificats de clés publiques intègrent des URI. Ils identifient un emplacement Web où vous pouvez vérifier la liste de révocation de certificats ou l'emplacement Web d'un serveur OCSP.

Pour plus d'informations, reportez-vous à RFC 2459: Certificate and CRL Profile et RFC 2560: Online Certificate Status Protocol - OCSP.

Coordination du temps sur des systèmes qui utilisent les certificats publics

Les certificats de clés publiques contiennent la date et l'heure d'émission et leur durée de validité. Par conséquent, les horloges de systèmes générant doivent être précises et utiliser des certificats. Le NTP (Network Time Protocol) : ce logiciel peut être utilisé pour synchroniser les horloges sur les systèmes. Le logiciel NTP du domaine public de l'Université du Delaware est inclus dans la version Oracle Solaris. La documentation est disponible sur le site Web NTP Documentation. Vous pouvez également installer le package `service/network/ptp` pour configurer le service Precision Time Protocol (PTP). Reportez-vous à la norme IEEE 1588 Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems.

Comparaison d'IKEv2 et IKEv1

Le tableau ci-dessous compare l'implémentation des versions IKEv2 et IKEv1 sur un système Oracle Solaris.

TABLEAU 8-1 Implémentation d'IKEv2 et d'IKEv1 dans Oracle Solaris

Fonctionnalité	IKEv2	IKEv1
chaîne de confiance de certificats	Keystore implicite basée sur les objets dans	Paramètre `cert_trust` dans le fichier `ike/config`
De création du certificat	Commande `ikev2cert`	Commande `ikecert certlocal`
Import du certificat.	La commande `ikev2cert import` peut importer des certificats et des clés dans le keystore PKCS #11	La commande `ikecert certdb`peut importer des certificats autonomes dans le keystore
Propriétaire du certificat	`ikeuser`	`root`
Fichier de stratégie de certificat	`kmf-policy.xml`	Une stratégie dans le fichier `ike/config`
L'emplacement de stockage des certificats	Bibliothèque softtoken PKCS #11	Bases de données IKEv1 locales
Répertoire du fichier de configuration	`/etc/inet/ike/`	`/etc/inet/ike/` et `/etc/inet/secret/`
Responsable de la configuration	Compte `ikeuser`	Compte `root`

Fonctionnalité	IKEv2	IKEv1
Démon	`in.ikev2d`	`in.iked`
Algorithmes FIPS 140 pour le trafic entre les démons †	Utiliser la structure cryptographique IKE SA	Les échanges ne prennent pas tous utiliser la structure cryptographique
Algorithmes FIPS 140 pour le trafic IPsec†	Utiliser la structure cryptographique	Utiliser la structure cryptographique
Fichier de stratégie IKE	`ike/ikev2.config`	`ike/config`
Clés IKE prépartagées	`ike/ikev2.preshared`	`secret/ike.preshared`
Port NAT	Port UDP `4500`	Port UDP `4500`
Port (Port)	Port UDP `500`	Port UDP `500`
Profil de droits	Network IPsec Management	Network IPsec Management
Nom de service (FMRI)	`svc:/ipsec/ike:ikev2`	`svc:/ipsec/ike:default`

†La fonctionnalité de structure cryptographique de Oracle Solaris 11.1 SRU 5.5 et SRU 3 est validée pour FIPS 140-2, Niveau 1. Si le Mode FIPS 140 est activé et que la structure cryptographique est en cours d'utilisation, les algorithmes validés par FIPS 140 sont alors utilisés. Par défaut, le Mode FIPS 140 n'est pas activé.

Protocole IKEv2

Cette section traite de l'implémentation d'IKEv2. Pour des informations sur IKEv1, reportez-vous à “Protocole IKEv1” à la page 140. Pour une comparaison, reportez-vous à “Comparaison d'IKEv2 et IKEv1” à la page 138. Pour plus d'informations qui s'appliquent aux deux protocoles, reportez-vous à “Introduction au protocole IKE” à la page 133. Oracle Solaris prend en charge les deux versions du protocole IKE simultanément.

Le démon IKEv2, `in.ikev2d`, négocie et authentifie les numéros de clé des AS IPsec. Reportez-vous à la page de manuel `in.ikev2d`(1M).

Choix de configuration pour IKEv2

Le fichier de configuration `/etc/inet/ike/ikev2.config` contient la configuration du démon `in.ikev2d`. La configuration se compose d'un certain nombre de règles. Chaque entrée contient des paramètres tels que des algorithmes et des données d'authentification que ce système peut utiliser avec un homologue IKEv2 configuré de manière similaire.

Le démon `in.ikev2d` prend en charge les clés prépartagées (PSK, Preshared Keys) et les certificats de clés publiques pour l'identité.

La page de manuel `ikev2.config`(4) présente des exemples de règles. Chaque règle doit avoir une étiquette unique. Vous trouverez ci-après une liste des étiquettes descriptives des exemples de règles de la page du manuel :

- `IP identities and PSK auth`

- `IP address prefixes and PSK auth`
- `IPv6 address prefixes and PSK auth`
- `Certificate auth with DN identities`
- `Certificate auth with many peer ID types`
- `Certificate auth with wildcard peer IDs`
- `Override transforms`
- `Mixed auth types`
- `Wildcard with required signer`

Remarque - Une clé prépartagée peut être utilisé avec n'importe quel ID de l'un des nombreux sites homologue (peer),. il autorise notamment des noms de domaine complets IP, noms distinctifs (DN), les adresses et des adresses électroniques.

Stratégie IKEv2 pour les certificats publics

Le fichier`kmf-policy.xml` contient les stratégies de validation de certificats pour IKEv2. La commande `kmfcfg dbfile=/etc/inet/ike/kmf-policy.xml policy=default` est utilisée pour modifier la stratégie de validation du certificat. Inclut l'utilisation de standard OCSP les modifications et des listes de certificats révoqués réseau, ainsi que la durée totale des délais d'expiration lors de la vérification du certificat. En outre, la stratégie permet aux administrateurs de modifier différents aspects de la validation de certificat, telles que la date de validité mise en oeuvre et de l'utilisation des clés. Il n'est pas recommandé d'assouplir les exigences par défaut pour la validation de certificat.

Protocole IKEv1

Les sections suivantes présentent un aperçu général d'IKEv1. IKEv1 est remplacé par IKEv2, ce qui offre une gestion des clés plus rapide et sécurisée. Pour plus d'informations sur IKEv2, reportez-vous à “Protocole IKEv2” à la page 139. Pour une comparaison, reportez-vous à “Comparaison d'IKEv2 et IKEv1” à la page 138. Pour des informations communes aux deux protocoles, reportez-vous à “Introduction au protocole IKE” à la page 133. IKEv1 et IKEv2 peuvent être exécutés simultanément et négocier avec leur protocole pair sur d'autres systèmes.

Négociation de clé IKEv1

Le démon IKEv1 `in.iked` négocie les clés et authentifie IPsec de façon sécurisée. IKEv1 fournit une confidentialité persistante parfaite (PFS). En mode PFS, les clés qui protègent

la transmission des données ne sont pas utilisées pour générer des clés complémentaires. De même, les amorces utilisées pour créer des clés de transmission de données ne sont pas réutilisées. Reportez-vous à la page de manuel `in.iked`(1M).

Echange IKEv1 de phase 1

Le protocole IKEv1 comporte deux phases. Oracle Solaris prend en charge l'échange *Main Mode* de phase 1. Négocie les paramètres acceptables Main Mode (mode principal) de change pour créer une association de sécurité (SA) ISAKMP entre les deux pairs. Cette SA ISAKMP utilise un chiffrement asymétrique pour échanger son matériel de clé et authentifier ses homologues à l'aide d'une clé prépartagée ou d'un certificat de clé publique. Contrairement aux SA IPsec, les SA ISAKMP sont bidirectionnelles. Il n'est donc pas nécessaire de disposer de plus d'une association de sécurité.

La manière dont IKEv1 négocie les AS ISAKAMP dans l'échange de phase 1 est configurable. IKEv1 lit les informations de configuration dans le fichier `/etc/inet/ike/config`. Ces informations incluent :

- Des paramètres généraux tels que le nom des certificats de clés publiques
- Nécessité ou non de la confidentialité persistante parfaite (PFS)
- IKE homologues cette du système
- La phase 1 les algorithmes qui protègent les échanges
- la méthode d'authentification.

  Les deux méthodes d'authentification utilisent respectivement les clés prépartagées et les certificats de clés publiques. Les certificats de clés publiques peuvent être autosignés ou émis par une autorité de certification (AC).

Pour plus d'informations, reportez-vous à la page de manuel `ike.config`(4).

Echange IKEv1 de phase 2

La phase 2 est connue sous le nom de *Quick Mode* (mode rapide). L'échange Quick Mode (mode rapide) négocie les algorithmes IPsec et les numéros de clés nécessaires pour créer des SA IPsec. Cet échange est protégé (chiffré) par le SA ISAKMP négocié au cours de la phase 1.

Les algorithmes et les protocoles de sécurité pour l'échange Quick Mode proviennent du fichier de stratégie IPsec, `/etc/inet/ipsecinit.conf`.

Les SA IPsec sont à nouveau générés lorsqu'ils expirent. La durée de vie de l'AS est définie par le démon `in.iked` lors de la création de l'AS IPsec. Cette valeur est configurable.

Pour plus d'informations, reportez-vous aux pages de manuel `ipsecconf`(1M) et `in.iked`(1M).

Choix de configuration pour IKEv1

Le fichier de configuration `/etc/inet/ike/config` contient la configuration du démon `in.iked`. La configuration se compose d'un certain nombre de règles. Chaque entrée contient des paramètres tels que des algorithmes et des données d'authentification que ce système peut utiliser avec un homologue IKEv1 configuré de manière similaire. Le démon `in.iked` prend en charge les clés prépartagées et les certificats de clés publiques pour l'identité.

Si l'entrée est `auth_method preshared`, ce sont les clés prépartagées qui sont utilisées pour authentifier l'échange. Si `auth_method` possède une valeur autre que `preshared`, l'authentification s'effectue à l'aide de certificats de clés publiques.

Dans IKEv1, les clés prépartagées sont associées à une adresse ou à une plage d'adresses IP. Les clés sont placées dans le fichier `/etc/inet/secret/ike.preshared` de chaque système.

Pour plus d'informations, reportez-vous à “Fonctionnement d'IKE” à la page 134 et aux pages de manuel `ike.config`(4) et `ike.preshared`(4).

♦ ♦ ♦ CHAPITRE 9

Configuration d'IKEv2

Ce chapitre décrit la procédure de configuration du protocole Internet Key Exchange version 2 (IKEv2) sur vos systèmes. Une fois IKEv2 configuré et activé, il génère automatiquement des numéros de clés pour les extrémités IPsec qu'il spécifie. Le présent chapitre contient les informations suivantes :

Pour une présentation du protocole IKE, reportez-vous au Chapitre 8, A propos d'Internet Key Exchange. Pour obtenir des informations de référence sur IKE, reportez-vous au Chapitre 12, IPsec et référence de gestion des clés. Pour d'autres procédures, reportez-vous aux exemples figurant dans les pages de manuel `ikeadm`(1M), `pktool`(1), `ikev2cert`(1M), `ikev2.config`(4), `in.ikev2d`(1M), and `kmfcfg`(1).

Configuration d'IKEv2

L'authentification du protocole IKE peut s'effectuer à l'aide de clés prépartagées ou de certificats autosignés ou émanant d'autorités de certifications (AC). Les méthodes d'authentification IKE sont liées par des règles aux points d'extrémité protégés. Vous pouvez donc utiliser toutes les méthodes d'authentification ou une seule d'entre elles sur un système. Vous pouvez également exécuter IKEv1 sur un système IKEv2. En règle générale, vous exécutez IKEv1 pour protéger les communications avec des systèmes qui ne prennent pas en charge IKEv2. IKEv2 peut également utiliser un jeton matériel PKCS #11 pour le stockage de la clé et du certificat.

Remarque - Les tâches suivantes partent du principe que les systèmes reçoivent des adresses IP statiques et que vous exécutez le profil de configuration réseau `DefaultFixed`. Si la commande `netadm list` renvoie `Automatic`, reportez-vous à la page de manuel `netcfg`(1M) pour plus d'informations.

Après avoir configuré IKEv2, réalisez les procédures IPsec du Chapitre 7, Configuration d'IPsec qui utilisent ces règles IKEv2 pour gérer leurs clés. Les sections suivantes examinent des configurations IKEv2 spécifiques.

Configuration d'IKEv2 avec des clés prépartagées

Elles s'utilisent notamment lors de la configuration du protocole IKEv2 sur deux systèmes homologues ou sous-réseaux gérés par le même administrateur. Les clés prépartagées peuvent également être utilisé lors du test. Pour plus d'informations, reportez-vous à “IKE avec l'authentification des clés prépartagées” à la page 135

▼ Configuration d'IKEv2 avec des clés prépartagées

Dans cette procédure, remplacez les noms `enigma` et `partym` par ceux de vos systèmes. Points d'extrémité de la configuration IKE.

Avant de commencer

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Vous devez saisir dans un shell de profil. Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, “Configuration de la stratégie IPsec à distance via une connexion `ssh`” et à “ Administration à distance de ZFS avec le shell sécurisé ” du manuel “ Gestion de l’accès au shell sécurisé dans Oracle Solaris 11.2 ” pour des instructions sur la connexion à distance sécurisée.

1. **Sur chaque système, modifiez le fichier `/etc/inet/ike/ikev2.config`.**

```
# pfedit /etc/inet/ike/ikev2.config
```

2. **Dans le fichier, créez une règle utilisant des clés prépartagées.**

Remarque - Vous créez ainsi les clés de l'Étape 4

Les règles et paramètres globaux de ce fichier doivent gérer les clés dans la stratégie IPsec dans le fichier `ipsecinit.conf` du système. Les exemples suivants de configuration d'IKEv2 gèrent les clés des exemples d'`ipsecinit.conf` dans “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

a. **Par exemple, modifiez le fichier `ikev2.config` sur le système `enigma` :**

Remarque - Dans cet exemple, deux section se transforme par les paramètres globaux. Un pair peut être configuré avec l'une de ces transformations. Transformation de façon à exiger un, incluez permettant de transformer particulier dans la règle.

```
### ikev2.config file on enigma, 192.168.116.16

## Global parameters
# This default value will apply to all transforms that follow
#
ikesa_lifetime_secs 3600
#
# Global transform definitions.  The algorithm choices are
# based on RFC 4921.
#
## Two transforms are acceptable to this system, Group 20 and Group 19.
## A peer can be configured with 19 or 20.
## To ensure that a particular peer uses a specific transform,
## include the transform in the rule.
##
# Group 20 is 384-bit ECP - Elliptic Curve over Prime
ikesa_xform { encr_alg aes(256..256) auth_alg sha384 dh_group 20 }
# Group 19 is 256-bit ECP
ikesa_xform { encr_alg aes(128..128) auth_alg sha256 dh_group 19 }
#
## The rule to communicate with partym
##  Label must be unique
{ label "enigma-partym"
  auth_method preshared
  local_addr  192.168.116.16
  remote_addr 192.168.13.213
}
```

b. Modifiez le `ikev2.config`fichier sur le système `partym` :

```
## ikev2.config file on partym, 192.168.13.213
## Global Parameters
#
...
ikesa_xform { encr_alg aes(256..256) auth_alg sha384 dh_group 20 }
ikesa_xform { encr_alg aes(128..128) auth_alg sha256 dh_group 19 }
...
## The rule to communicate with enigma
##  Label must be unique
{ label "partym-enigma"
  auth_method preshared
  local_addr  192.168.13.213
  remote_addr 192.168.116.16
}
```

3. Sur chaque système, vérifiez la syntaxe du fichier.

```
# /usr/lib/inet/in.ikev2d -c
```

4. **Mettez la clé prépartagée dans le fichier `/etc/inet/ike/ikev2.preshared` sur chaque système.**

Attention - Ce fichier a des autorisations spéciales et son propriétaire est `ikeuser`. Ne jamais supprimer ou de remplacer ce fichier. Au lieu de cela, utilisez la commande `pfedit` pour modifier son contenu de façon à ce que le fichier conserve ses propriétés d'origine.

   a. **Sur le système `enigma` par exemple, le fichier `ikev2.preshared` se présente comme suit :**

```
# pfedit -s /etc/inet/ike/ikev2.preshared
## ikev2.preshared on enigma, 192.168.116.16
#…
## label must match the rule that uses this key
{ label "enigma-partym"
## The preshared key can also be represented in hex
## as in 0xf47cb0f432e14480951095f82b
   key "This is an ASCII Cqret phrAz, use str0ng p@ssword tekniques"
}
```

   Pour plus d'informations sur les options de la commande `pfedit`, reportez-vous à la page de manuel `pfedit`(1M).

   b. **Sur le système `partym`, le fichier `ikev2.preshared` est similaire, mis à part son étiquette unique :**

```
## ikev2.preshared on partym, 192.168.13.213
#…
## label must match the label of the rule that uses this key
{ label "partym-enigma"
## The preshared key can also be represented in hex
## as in 0xf47cb0f432e14480951095f82b
 key "This is an ASCII Cqret phrAz, use str0ng p@ssword tekniques"
 }
```

5. **Activez l'instance de service IKEv2.**

```
# svcadm enable ipsec/ike:ikev2
```

Lorsque vous remplacez la clé prépartagée, modifiez les fichiers de clés prépartagées sur ses systèmes homologues et redémarrez le service `ikev2`.

```
# svcadm restart ikev2
```

Exemple 9-1 Utilisation de différentes clés prépartagées IKEv2

Dans cet exemple, les administrateurs IKEv2 créent une clé prépartagée par système, les échangent et ajoutent chaque clé au fichier de clés prépartagées. L'étiquette de l'entrée de clé prépartagée correspond à l'étiquette dans une règle du fichier `ikev2.config`. Ensuite, ils redémarrent le démon `in.ikev2d`.

Après avoir reçu la clé prépartagée de l'autre système, l'administrateur modifie le fichier `ikev2.preshared`. Le fichier sur `partym` est le suivant :

```
# pfedit -s /etc/inet/ike/ikev2.preshared
#…
{ label "partym-enigma"
## local and remote preshared keys
local_key  "P-LongISH key Th@t m^st Be Ch*angEd \'reguLarLy)"
remote_key "E-CHaNge lEyeGhtB+lBs et KeeS b4 2LoOoOoOoOng"
}
```

Par conséquent, le fichier `ikev2.preshared` sur `enigma` doit être le suivant :

```
#…
{ label "enigma-partym"
## local and remote preshared keys
local_key  "E-CHaNge lEyeGhtB+lBs et KeeS b4 2LoOoOoOoOng"
remote_key "P-LongISH key Th@t m^st Be Ch*angEd \'reguLarLy)"
}
```

Les administrateurs redémarrent l'instance de service IKEv2 sur chacun des systèmes.

```
# svcadm restart ikev2
```

Étapes suivantes Si vous n'avez pas terminé d'établir la stratégie IPsec, effectuez de nouveau la procédure IPsec pour activer ou actualiser la stratégie IPsec. Pour obtenir des exemples de la stratégie IPsec de protection de VPN, reportez-vous à “Protection d'un VPN à l'aide d'IPsec” à la page 117. Pour plus d'exemples de la stratégie IPsec, reportez-vous à la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

Pour plus d'exemples, reportez-vous aux pages de manuel `ikev2.config`(4) et `ikev2.preshared`(4).

▼ Ajout d'un nouvel homologue lors de l'utilisation de clés prépartagées dans IKEv2

Si vous ajoutez des entrées de stratégie IPsec à une configuration de travail entre des systèmes homologues, vous devez actualiser le service de stratégie IPsec. Il n'est pas nécessaire de reconfigurer ou de redémarrer IKE.

Si vous ajoutez un nouveau système homologue à la stratégie IPsec, vous devez modifier non seulement IPsec, mais aussi la configuration IKEv2.

Avant de commencer

Vous avez mis à jour le fichier `ipsecinit.conf` et actualisé la stratégie IPsec pour les systèmes homologues.

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Vous devez saisir dans un shell de profil. Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, "Configuration de la stratégie IPsec à distance via une connexion `ssh`" et à " Administration à distance de ZFS avec le shell sécurisé " du manuel " Gestion de l'accès au shell sécurisé dans Oracle Solaris 11.2 " pour des instructions sur la connexion à distance sécurisée.

1. **Créez une règle pour la gestion des clés par IKEv2, pour le nouveau système qui utilise IPsec.**

   a. **Par exemple, sur le système `enigma`, ajoutez la règle suivante au fichier `/etc/inet/ike/ikev2.config` :**

   ```
   # pfedit ikev2.config
   ## ikev2.config file on enigma, 192.168.116.16
   ...
   ## The rule to communicate with ada
   ##  Label must be unique
   {label "enigma-ada"
    auth_method preshared
    local_addr  192.168.116.16
    remote_addr 192.168.15.7
   }
   ```

   Pour plus d'informations sur les options de la commande `pfedit`, reportez-vous à la page de manuel `pfedit`(1M).

   b. **Sur le système `ada`, ajoutez la règle suivante :**

   ```
   ## ikev2.config file on ada, 192.168.15.7
   ...
   ## The rule to communicate with enigma
   {label "ada-enigma"
    auth_method preshared
    local_addr  192.168.15.7
    remote_addr 192.168.116.16
   }
   ```

2. **(Facultatif) Sur chaque système, vérifiez la syntaxe du fichier.**

```
# /usr/lib/inet/in.ikev2d -c -f /etc/inet/ike/ikev2.config
```

3. **Créez une clé prépartagée IKEv2 pour les systèmes homologues.**

   a. **Sur le système `enigma`, ajoutez les informations suivantes au fichier `/etc/inet/ike/ikev2.preshared` :**

   ```
   # pfedit -s /etc/inet/ike/ikev2.preshared
   ## ikev2.preshared on enigma for the ada interface
   ...
   ## The rule to communicate with ada
   ##  Label must match the label of the rule
   { label "enigma-ada"
     # enigma and ada's shared key
     key "Twas brillig and the slivey toves did *s0mEtHiNg* be CareFULL hEEEr"
   }
   ```

   Pour plus d'informations sur les options de la commande `pfedit`, reportez-vous à la page de manuel `pfedit`(1M).

   b. **Sur le système `ada`, ajoutez les informations suivantes au fichier `ikev2.preshared` :**

   ```
   # ikev2.preshared on ada for the enigma interface
   #
   { label "ada-enigma"
     # ada and enigma's shared key
     key "Twas brillig and the slivey toves did *s0mEtHiNg* be CareFULL hEEEr"
   }
   ```

4. **Sur chaque système, lisez les modifications dans le noyau.**

   - **Si le service est activé, actualisez-le.**

     ```
     # svcadm refresh ikev2
     ```

   - **Si le service n'est pas activé, activez-le .**

     ```
     # svcadm enable ikev2
     ```

Étapes suivantes Si vous n'avez pas terminé d'établir la stratégie IPsec, effectuez de nouveau la procédure IPsec pour activer ou actualiser la stratégie IPsec. Pour obtenir des exemples de la stratégie IPsec de protection de VPN, reportez-vous à “Protection d'un VPN à l'aide d'IPsec” à la page 117. Pour plus d'exemples de la stratégie IPsec, reportez-vous à la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

Initialisation du keystore en vue du stockage de certificats de clés publiques pour IKEv2

Pour utiliser des certificats publics avec IKEv2, vous devez créer un keystore PKCS #11. Le keystore le plus fréquemment utilisé emploie `pkcs11_softtoken`, fourni par la Structure cryptographique d'Oracle Solaris.

Le keystore `pkcs11_softtoken` pour IKEv2 se trouve dans un répertoire dont le propriétaire est un utilisateur spécial, `ikeuser`. Le répertoire par défaut est `/var/user/ikeuser`. L'ID utilisateur `ikeuser` est fourni avec le système, mais vous devez créer le keystore. Lorsque vous créez le PIN, vous créez un keystore pour le keystore. Le service IKEv2 a besoin de ce PIN pour se connecter au keystore.

Le keystore `pkcs11_softtoken` contient les clés privées et publiques et les certificats publics utilisés par IKEv2. Ces clés et les certificats sont gérés à l'aide de la commande `ikev2cert`, qui est un wrapper pour la commande `pktool`. Le wrapper assure que toutes les clés et opérations de certificats sont appliquées au keystore `pkcs11_softtoken` appartenant à `ikeuser`.

Si vous n'avez pas ajouté le code PIN en tant que valeur de propriété du service `ikev2`, le message suivant s'affiche dans le fichier `/var/log/ikev2/in.ikev2d.log` :

```
date: (n)  No PKCS#11 token "pin" property defined
for the smf(5) service: ike:ikev2
```

Si vous n'utilisez pas les certificats de clés publiques, vous pouvez ignorer ce message.

▼ Création et utilisation d'un keystore pour les certificats de clés publiques IKEv2

Vous devez créer un keystore si vous prévoyez d'utiliser des certificats publics avec IKEv2. Pour utiliser le keystore, vous devez vous connecter à. Lorsque le démon `in.ikev2d` démarre, le code PIN lui est fourni par vous ou par un processus automatique. Si la sécurité du site permet à la connexion automatique, vous devez le configurer. La valeur par défaut est un keystore se connecter et utiliser l'interactif.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Vous devez saisir dans un shell de profil. Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

1. **Déterminez le code PIN pour le keystore IKEv2.**

Utilisez la commande `ikev2cert setpin` pour créer le keystore IKEv2 . Cette commande définit `ikeuser` comme propriétaire du keystore PKCS #11.

Ne laissez pas d'espaces dans le code PIN. Par exemple, la valeur `WhatShouldIWrite` est valide, mais la valeur `"What Should"` ne l'est pas.

```
% pfbash
# /usr/sbin/ikev2cert setpin
Enter token passphrase: changeme
Create new passphrase:      Type strong passphrase
Re-enter new passphrase: xxxxxxxx
Passphrase changed.
```

Attention - Stockez cette phrase secrète dans un emplacement sûr. Elle est nécessaire pour utiliser le keystore.

2. **Connectez-vous au keystore automatiquement ou de façon interactive.**

   La connexion automatique est recommandée. Si la stratégie de sécurité du site n'autorise pas la connexion automatique, vous devez vous connecter au keystore de manière interactive au redémarrage du démon `in.ikev2d`.

   - **Configurez le keystore pour permettre la connexion automatique.**

     a. **Ajoutez le code PIN comme valeur de la propriété de service `pkcs11_softtoken/pin`.**

        ```
        # svccfg -s ike:ikev2 editprop
        ```

        Fenêtre d'édition de temporaire s'ouvre.

     b. **Supprimez les marques de commentaire de la ligne `setprop pkcs11_token/pin =`.**

        ```
        # setprop pkcs11_token/pin = astring: ()      Original entry
        setprop pkcs11_token/pin = astring: ()  Uncommented entry
        ```

     c. **Remplacez les parenthèses par le code PIN de l'Étape 1.**

        ```
        setprop pkcs11_token/pin = astring: PIN-from-Step-1
        ```

        Laissez un espace entre le caractère deux-points et le PIN.

     d. **Supprimez les marques de commentaire de la ligne `refresh` à la fin du fichier, puis enregistrez les modifications.**

        ```
        # refresh
        refresh
        ```

e. **(Facultatif) Vérifiez la valeur de la propriété `pkcs11_token/pin`.**

La propriété `pkcs11_token/pin` contient la valeur qui sera vérifiée lors de l'accès au keystore appartenant à `ikeuser`.

```
# svccfg -s ike:ikev2 listprop pkcs11_token/pin
pkcs11_token/pin     astring      PIN
```

- **Keystore de connexion dans ce n'est pas configuré, connectez-vous au keystore manuellement.**

Exécutez cette commande chaque fois que le démon `in.ikev2d` démarre.

```
# pfbash
# ikeadm -v2 token login "Sun Metaslot"
Enter PIN for PKCS#11 token 'Sun Metaslot':      Type the PIN from Step 1
ikeadm: PKCS#11 operation successful
```

3. **(Facultatif) Vérifiez si un code PIN a bien été défini dans le keystore.**

```
# ikev2cert tokens
Flags: L=Login required  I=Initialized  X=User PIN expired  S=SO PIN expired
Slot ID     Slot Name                   Token Name                      Flags
-------     ---------                   ----------                      -----
1           Sun Crypto Softtoken        Sun Software PKCS#11 softtoken    LI
```

La valeur `LI` dans la colonne `Flags` indique que le code PIN a été défini.

4. **Pour vous déconnecter manuellement de `pkcs11_softtoken`, utilisez la commande `ikeadm`.**

```
# ikeadm -v2 token logout "Sun Metaslot"
ikeadm: PKCS#11 operation successful
```

Vous pouvez vous déconnecter pour limiter la communication entre deux sites pour une période précise donnée. En vous déconnectant, la clé privée n'est plus disponible. Aucune nouvelle session IKEv2 ne peut être démarrée. La session IKEv2 en cours se poursuit, à moins que vous supprimiez les clés de session à l'aide de la commande `ikeadm delete ikesa`. Les règles de clé prépartagée continuent de fonctionner. Reportez-vous à la page de manuel `ikeadm`(1M).

Configuration d'IKEv2 avec des certificats à clé publique

Les certificats publics émanant est particulièrement utile pour les déploiements de grande envergure. Pour plus d'informations, reportez-vous à la section “IKE avec certificats de clés publiques” à la page 136.

Les certificats de clés publiques sont stockées dans un keystore softtoken fourni par la structure cryptographique. Sur les systèmes dotés de matériel connecté, les certificats peuvent également être stockés sur le matériel. Pour plus d'informations sur cette procédure, reportez-vous à “Génération et stockage de certificats de clés publiques pour IKEv2 dans le matériel” à la page 166.

Pour plus d'informations d'ordre général, reportez-vous à la section “Fonctionnement d'IKE” à la page 134.

la liste de tâches suivantes décrit les procédures de création de certificats de clé publique pour IKEv2. Ces procédures incluent la procédure de stockage des certificats dans un keystore matériel si votre système dispose d'une carte Sun Crypto Accelerator 6000 attachée.

TABLEAU 9-1 Liste des tâches pour la configuration d'IKEv2 avec des certificats de clés publiques

Tâche	Description	Pour obtenir des instructions
Créer un keystore de certificats.	Initialise le keystore PKCS #11 où sont stockés les certificats pour IKEv2.	“Initialisation du keystore en vue du stockage de certificats de clés publiques pour IKEv2” à la page 150
Configurez le protocole IKEv2 avec des certificats de clés publiques autosignés.	Signé crée un certificat de clé publique par vous-même . Les exports et importe le certificat à celles de ses pairs les certificats du pair.	“Configuration d'IKEv2 avec des certificats de clés publiques autosignés” à la page 153
Configurer IKEv2 avec un certificat issu d'une CA.	Nécessite que vous créiez une renvoyé CSR des certificats et les importer dans le keystore toutes les. Vérifiez ensuite pair et importez les certificats IKE.	“Configuration du protocole IKEv2 avec des certificats signés par une AC” à la page 160
Configurer la manière dont les certificats révoqués sont gérées.	Détermine si les serveurs sont utilisés et le protocole OCSP des listes de certificats révoqués sont interrogées, y compris la marche à suivre pour traiter retard réseau.	“Définition d'une stratégie de validation de certificats dans IKEv2” à la page 162
Configurez le stockage de certificats dans le keystore de la connexion des composants matériels.	Localise la carte Sun Crypto Accelerator 6000 et configure IKEv2 afin de l'utiliser.	“Génération et stockage de certificats de clés publiques pour IKEv2 dans le matériel” à la page 166

▼ Configuration d'IKEv2 avec des certificats de clés publiques autosignés

Dans cette procédure, vous créez et signez un certificat de clé publique. La clé privée et le certificat sont stockés dans le keystore de jeton logiciel PKCS #11 pour IKEv2. Vous envoyez le certificat de clé publique aux homologues IKE, qui, à leur tour, vous envoient leur certificat public.

Vous effectuez cette procédure sur tous les systèmes utilisant IKE les certificats auto-signés .

Avant de commencer

Pour utiliser les certificats, vous devez avoir effectué la “Création et utilisation d'un keystore pour les certificats de clés publiques IKEv2” à la page 150.

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Vous devez utiliser un shell de profil. Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, "Configuration de la stratégie IPsec à distance via une connexion `ssh`" et à " Administration à distance de ZFS avec le shell sécurisé " du manuel " Gestion de l'accès au shell sécurisé dans Oracle Solaris 11.2 " pour des instructions sur la connexion à distance sécurisée.

1. **Créez un certificat autosigné dans le keystore.**

   Pour obtenir une description des arguments de la commande `ikev2cert gencert`, vérifiez la sous-commande `pktool gencert keystore=pkcs11` de la page de manuel `pktool`(1).

   Pour le format de l'argument `subject`, reportez-vous à "Utilisation de certificats de clés publiques dans IKE" à la page 137.

   Remarque - Donnez une étiquette au certificat. L'étiquette identifie le certificat, ainsi que ses keystore des clés correspondantes dans les fichiers locaux.

   a. **Par exemple, la commande du système `partym` apparaîtrait comme suit :**

```
# pfbash
 # ikev2cert gencert \
 label="ITpartym" \
 subject="O=exampleco, OU=IT, C=US, CN=partym" \
 serial=0x87654321
    keytype=rsa
    keylen=2048
 Enter PIN for Sun Software PKCS#11 softtoken: xxxxxxxx
```

   Les messages d'erreur suivants indiquent un code PIN contenant une faute de frappe ou un keystore non initialisé :

```
Error creating certificate and keypair:
 keystore error: CKR_PIN_INCORRECT
 libkmf error: KMF_ERR_AUTH_FAILED

 Error creating certificate and keypair:
 keystore error: CKR_PIN_EXPIRED: PIN expired and must be changed
 libkmf error: KMF_ERR_BAD_PARAMETER: invalid parameter
```

Astuce - Un affichage de la syntaxe de la commande `pktool` indique qu'une partie de votre certificat est mal saisie. , reportez-vous à l'utilisation d'une commande non-schéma ne sont pas autorisés pour le, les guillemets et algorithme de signes égal manquant, ainsi que d'autres des erreurs typographiques. Une stratégie est pour localiser l'argument non valide, puis pour extraire les enlever un argument de commande, l'un après l'autre.

b. La commande sur le système `enigma` se présenterait de la manière suivante :

```
# ikev2cert gencert \
 label=ITenigma \
 subject="O=exampleco, OU=IT, C=US, CN=enigma" \
 serial=0x86428642
    keytype=rsa
    keylen=2048
 Enter PIN for Sun Software PKCS#11 softtoken: xxxxxxxx
```

2. (Facultatif) Listez les clés et le certificat.

```
enigma # /usr/sbin/ikev2cert list objtype=both
 Enter PIN for Sun Software PKCS#11 softtoken: xxxxxxxx
 No.     Key Type        Key Len.        Key Label
 -----------------------------------------------------
 Asymmetric private keys:
 1)      RSA                             ITenigma
 Asymmetric public keys:
 1)      RSA                             ITenigma
 Certificates:
 1) X.509 certificate
  Label: ITenigma
  Subject: C=US, O=exampleco, OU=IT, CN=enigma
  Issuer: C=US, O=exampleco, OU=IT, CN=enigma
  Not Before: April 10 21:49:00 2014 GMT
  Not After: April 10 21:49:00 2015 GMT
  Serial: 0x86426420
  Signature Algorithm: sha1WithRSAEncryption
  X509v3 Subject Key Identifier:
   34:7a:3b:36:c7:7d:4f:60:ed:ec:4a:96:33:67:f2:ac:87:ce:35:cc
  SHA1 Certificate Fingerprint:
   68:07:48:65:a2:4a:bf:18:f5:5b:95:a5:01:42:c0:26:e3:3b:a5:30
```

Astuce - L'algorithme de hachage par défaut est `SHA1`. Pour créer certificat avec un algorithme de signature plus fort, utilisez l'option `keytype` et un autre algorithme de hachage, par exemple `keytype=rsa hash=sha384`. Pour connaître les options, reportez-vous à la page de manuel `pktool`(1)

3. Livrer le certificat à l'autre système.

a. Sur chaque système, exportez uniquement le certificat dans un fichier.

L'option `outformat=pem` permet de vous assurer que le certificat public est placé dans le fichier dans un format convenant à l'importation directe. L'étiquette keystore identifie le certificat dans le.

```
# cd /tmp
 # ikev2cert export objtype=cert outformat=pem outfile=filename label=label
 Enter PIN for Sun Software PKCS#11 softtoken:xxxxxxxx
```

b. Envoyez ce certificat à l'autre système par e-mail, `sftp` ou `ssh`.

Par exemple administrez les deux systèmes, utilisez la commande `sftp` pour ramener le certificat de l'autre système.

```
enigma # sftp jdoe@partym:/tmp/ITpartym.pem /tmp/ITpartym.pem.cert
 partym # sftp jdoe@enigma:/tmp/ITenigma.pem /tmp/ITenigma.pem.cert
```

Vous êtes invité à saisir votre mot de passe. Dans cet exemple, `jdoe` doit fournir un mot de passe.

4. Vérifiez que les certificats sont identiques.

Avant de charger un certificat dans le keystore, assurez-vous que c'est le bon.

a. Créer une synthèse du fichier exporté sur chaque système.

Par exemple, l'administrateur de `partym` envoie par e-mail à l'autre administrateur l'empreinte du fichier contenant le certificat de `partym`. L'administrateur de `enigma` envoie pour sa part l'empreinte du fichier de certificat de `enigma`.

```
partym # digest -a sha1 /tmp/ITpartym.pem
 c6dbef4136c0141ae62110246f288e5546a59d86

 enigma # digest -a sha1 ITenigma.pem
 6b288a6a6129d53a45057065bd02b35d7d299b3a
```

b. Sur l'autre système, exécutez la commande `digest` sur le fichier contenant le certificat du premier système.

```
enigma # digest -a sha1 /tmp/ITpartym.pem.cert
 c6dbef4136c0141ae62110246f288e5546a59d86

 partym # digest -a sha1 /tmp/ITenigma.pem.cert
 6b288a6a6129d53a45057065bd02b35d7d299b3a
```

Les condensés doivent correspondre. Si elles ne correspondent pas, ne pas importer le contenu du fichier dans le keystore. Pour une autre manière de vérifier la validité d'un

certificat, voir l'Exemple 9-3, “Vérification d'un certificat de clé publique à l'aide de son empreinte”.

5. **Après vérification, importez le certificat de l'autre système dans le keystore.**

   Lorsque vous importez un certificat dans votre keystore, vous devez lui assigner une étiquette unique pour l'identifier sur votre système. La clé publique l'étiquette avec ses liens certificat de clé publique.

```
enigma# ikev2cert import label=ITpartym1 infile=/tmp/ITpartym.pem.cert

partym# ikev2cert import label=ITenigma1 infile=/tmp/ITenigma.pem.cert
```

6. **(Facultatif) Listez les objets dans le keystore.**

   Comparez la liste à celle de l'Étape 2. Par exemple, le certificat de `partym` est ajouté dans le keystore de `enigma`.

```
enigma # /usr/sbin/ikev2cert list objtype=both
 Enter PIN for Sun Software PKCS#11 softtoken: xxxxxxxx
 No.     Key Type        Key Len.        Key Label
 -----------------------------------------------------
 Asymmetric private keys:
 1)      RSA                             ITenigma
 Asymmetric public keys:
 1)      RSA                             ITenigma
 Certificates:
 1) X.509 certificate
  Label: ITenigma
  Subject: C=US, O=exampleco, OU=IT, CN=enigma
  Issuer: C=US, O=exampleco, OU=IT, CN=enigma
  Not Before: April 10 21:49:00 2014 GMT
  Not After: April 10 21:49:00 2015 GMT
  Serial: 0x86426420
  Signature Algorithm: sha1WithRSAEncryption
  X509v3 Subject Key Identifier:
   34:7a:3b:36:c7:7d:4f:60:ed:ec:4a:96:33:67:f2:ac:87:ce:35:cc
  SHA1 Certificate Fingerprint:
   68:07:48:65:a2:4a:bf:18:f5:5b:95:a5:01:42:c0:26:e3:3b:a5:30

 2) X.509 certificate
  Label: ITpartym1
  Subject: C=US, O=exampleco, OU=IT, CN=partym
  Issuer: C=US, O=exampleco, OU=IT, CN=partym
  Not Before: April 10 21:40:00 2014 GMT
  Not After: April 10 21:40:00 2015 GMT
  Serial: 0x87654321
  Signature Algorithm: sha1WithRSAEncryption
  X509v3 Subject Key Identifier:
   ae:d9:c8:a4:19:68:fe:2d:6c:c2:9a:b6:06:55:b5:b5:d9:d9:45:c6
  SHA1 Certificate Fingerprint:
   83:26:44:29:b4:1f:af:4a:69:0d:87:c2:dc:f4:a5:1b:4f:0d:36:3b
```

7. **Sur chaque système, utilisez les certificats dans une règle IKEv2.**

Utilisez la commande `pfedit` pour modifier le fichier `/etc/inet/ike/ikev2.config`.

a. Par exemple, sur le système `partym`, la règle du fichier `ikev2.config` ressemble à ceci :

```
##  ... Global transform that applies to any rule without a declared transform
 ikesa_xform { dh_group 21 auth_alg sha512 encr_alg aes }
 ##  ... Any self-signed
 ## end-entity certificates must be present in the keystore or
 ## they will not be trusted.
 {
  label "partym-enigma"
  auth_method cert
  local_id DN = "O=exampleco, OU=IT, C=US, CN=partym"
  remote_id DN = "O=exampleco, OU=IT, C=US, CN=enigma"
 }
 ...
```

b. Sur le système `enigma`, utilisez le ND du certificat `enigma` comme valeur de `local_id` dans le fichier `ikev2.config`.

Pour la valeur du paramètre distant, utilisez le ND du certificat de `partym`. Assurez-vous que la valeur du mot-clé `label` est unique sur le système local.

```
…
 ikesa_xform { dh_group 21 auth_alg sha512 encr_alg aes }
 …
 {
  label "enigma-partym"
  auth_method cert
  local_id DN = "O=exampleco, OU=IT, C=US, CN=enigma"
  remote_id DN = "O=exampleco, OU=IT, C=US, CN=partym"
 }
 ...
```

8. (Facultatif) Sur chaque système, vérifiez la validité des fichiers `ikev2.config`.

```
# /usr/lib/inet/inikev2.d -c
```

Des erreurs typographiques ou des inexactitudes corrigez les éventuelles avant de continuer.

9. Sur chaque système, vérifiez l'état de l'instance de service IKEv2.

```
# svcs ikev2
 STATE          STIME    FMRI
 disabled       Sep_07   svc:/network/ipsec/ike:ikev2
```

10. Sur chaque système, activez l'instance de service IKEv2.

```
partym # svcadm enable ipsec/ike:ikev2
```

```
enigma # svcadm enable ipsec/ike:ikev2
```

Exemple 9-2 Création d'un certificat autosigné à l'aide d'une durée de vie limitée

Dans cet exemple, l'administrateur indique que le certificat est valide pendant deux ans.

```
# ikev2cert gencert \
  > label=DBAuditV \
  > serial=0x12893467235412 \
  > subject="O=exampleco, OU=DB, C=US, CN=AuditVault" \
  > altname=EMAIL=auditV@example.com \
  > keytype=ec curve=secp521r1 hash=sha512 \
    > lifetime=2-year
```

Exemple 9-3 Vérification d'un certificat de clé publique à l'aide de son empreinte

Dans cet exemple, l'administrateur utilise l'empreinte du certificat pour vérifier le certificat. L'inconvénient de cette méthode est qu'il appartient à l'administrateur d'importer le certificat dans le keystore du pair avant de visualiser l'empreinte.

L'administrateur importe le certificat, le liste à l'aide de la commande `ikev2cert list objtype=cert`, puis copie son empreinte numérique depuis la sortie et l'envoie à l'administrateur de l'autre système.

```
SHA1 Certificate Fingerprint:
            83:26:44:29:b4:1f:af:4a:69:0d:87:c2:dc:f4:a5:1b:4f:0d:36:3b
```

Si la vérification échoue, l'administrateur qui a importé le certificat doit supprimer ce certificat et sa clé publique du keystore.

```
# ikev2cert delete label=label-name
      Enter PIN for Sun Software PKCS#11 softtoken: xxxxxxxx
      1 public key(s) found, do you want to delete them (y/N) ? y
      1 certificate(s) found, do you want to delete them (y/N) ? y
```

Étapes suivantes Si vous n'avez pas terminé d'établir la stratégie IPsec, effectuez de nouveau la procédure IPsec pour activer ou actualiser la stratégie IPsec. Pour obtenir des exemples de la stratégie IPsec de protection de VPN, reportez-vous à “Protection d'un VPN à l'aide d'IPsec” à la page 117. Pour plus d'exemples de la stratégie IPsec, reportez-vous à la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

Configuration du protocole IKEv2 avec des certificats signés par une AC

Les organisations qui protéger un grand nombre de systèmes communicants utilisent généralement les certificats publics émanant d'une autorité de certification (CA). Pour plus d'informations, reportez-vous à la section “IKE avec certificats de clés publiques” à la page 136.

Vous effectuez cette procédure sur tous les certificats IKE à partir d'un systèmes CA qui utilisent.

Avant de commencer

Pour utiliser les certificats, vous devez avoir effectué la “Création et utilisation d'un keystore pour les certificats de clés publiques IKEv2” à la page 150.

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Vous devez saisir dans un shell de profil. Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, “Configuration de la stratégie IPsec à distance via une connexion `ssh`” et à “ Administration à distance de ZFS avec le shell sécurisé ” du manuel “ Gestion de l’accès au shell sécurisé dans Oracle Solaris 11.2 ” pour des instructions sur la connexion à distance sécurisée.

1. **Passez à un répertoire accessible en écriture.**

   Le message d'erreur suivant peut indiquer que le fichier CSR ne peut pas être écrit sur le disque :

   ```
   Warning: error accessing "CSR-file"
   ```

   Par exemple, utilisez le répertoire `/tmp`.

   ```
   # cd /tmp
   ```

2. **Créez une demande de signature de certificat.**

   Vous utilisez la commande `ikev2cert gencsr` pour créer une demande de signature de certificat (CSR, Certificate Signing Request). Pour obtenir une description des arguments de la commande, vérifiez la sous-commande `pktool gencsr keystore=pkcs11` à la page de manuel `pktool(1)` man page.

   Par exemple, la commande suivante crée un fichier qui contient les CSR du système `partym` :

   ```
   # pfbash
   # /usr/sbin/ikev2cert gencsr \
   keytype=rsa
   keylen=2048
   label=Partym1 \
   outcsr=/tmp/Partymcsr1 \
   ```

```
subject="C=US, O=PartyCompany\, Inc., OU=US-Partym, CN=Partym"
Enter PIN for Sun Software PKCS#11 softtoken: xxxxxxxx
```

3. **(Facultatif) Tout d'abord copier le contenu du CA CSR de et le coller dans le formulaire Web.**

```
# cat /tmp/Partymcsr1
-----BEGIN CERTIFICATE REQUEST-----
MIICkDCCAXoCAQAwTzELMAkGA1UEBhMCVVMxGzAZBgNVBAoTElBhcnR5Q29tcGFu
eSwgSW5jLjESMBAGA1UECxMJVVMtUGFydHltMQ8wDQYDVQQDEwZQYXJ0eW0wggEi
MA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCMbINmgZ4XWUv2q1fshZUN/SLb
WNLXZxdKwt5e71o0owjyby69eL7HE0QBUij73nTkXE3n4gxojBZE+hvJ6GOCbREA
jgSquP2US7Bn9XEcXRrsOc7MCFPrsA+hVIcNHpKNseUOU/rg+wzoo5hA1ixtWuXH
bYDeEWQi5tlZgDZoCWGrdHEjwVyHfvz+a0WBjyZBYOueBhXaa68QqSOSnRVDX56Q
3p4H/AR4h0dcSja72XmMKPU5p3RVb8n/hrfKjiDjiGYXD4D+WZxQ65xxCcnALvVH
nZHUlAtP7QHX4RXlQVNNwEsY6C95RX9297rNWLsYvp/86xWrQkTlNqVAeUKhAgMB
AAEwCwYJKoZIhvcNAQEFA4IBAQB3R6rmZdqcgN8Tomyjp2CFTdyAWixkIATXpLM1
GL5ghrnDvadD61M+vS1yhFlIcSNM8fLRrCHIKtAmB8ITnggJ//rzbHq3jdla/iQt
kgGoTXfz8j6B57Ud6l+MBLiBSBy0QK4GIg8Ojlb9Kk5HsZ48mIoI/Qb7FFW4p9dB
JEUn0eYhkaGtwJ21YNNvKgOeOcnSZy+xP9Wa9WpfdsBO4TicLDw0Yq7koNnfL0IB
Fj2bt/wI7iZ1DcpwphsiwnW9K9YynAJZzHd1ULVpn5Kd7vSRz9youLLzSb+9ilgO
E43DW0hRk6P/Uq0N4e1Zca4otezNxyEqlPZI7pJ5uOo0sbiw
-----END CERTIFICATE REQUEST-----
```

4. **Soumettez le CSR à une autorité de certification (AC).**

L'AC peut vous indiquer comment soumettre la CSR. Dans la plupart des cas, celle-ci s'effectue en remplissant un formulaire directement sur le site Web du fournisseur. Dans ce formulaire, vous devrez notamment indiquer la preuve de la légitimité de votre demande. Il suffit généralement de coller votre CSR dans le formulaire.

Astuce - L'accès à certains formulaires comptent par défaut un bouton Avancé dans laquelle vous pouvez coller votre certificat. La CSR PKCS # 10 est générée dans format. Par conséquent, trouver la portion du Web PKCS # 10 format pris en charge par les mentions.

---

5. **Importez tous les certificats reçus depuis l'AC dans votre keystore.**

`ikev2cert import` importe le certificat dans le keystore.

a. **Importez la clé publique et le certificat que vous avez reçus de la part de l'autorité de certification.**

```
# ikev2cert import objtype=cert label=Partym1 infile=/tmp/Partym1Cert
```

Astuce - Pour faciliter l'administration, assignez au certificat importé la même étiquette que celle de la CSR initiale.

---

b. **A partir de l'importer le certificat CA root.**

```
# ikev2cert import objtype=cert infile=/tmp/Partym1CAcert
```

c. **Les certificats intermédiaires d'importation dans le keystore CA.**

Astuce - Pour les tâches d'administration plus de commodité, affecter la même étiquette à des certificats intermédiaires importés en tant que libellé de la feuille de calcul d'origine CSR.

A envoyé si le CA des fichiers distincts pour chaque intermédiaire, puis les importer en tant qu'certificat vous avez importé le ci-dessus des certificats. Toutefois, si l'autorité de certification fournit sa chaîne de certificat en tant que fichier PKCS#7, vous devez extraire les certificats individuels du fichier, puis importer chaque certificat de la même manière que vous aviez importé les certificats précédents :

Remarque - Vous devez prendre le rôle `root` pour exécuter la commande `openssl`. Reportez-vous à la page de manuel `openssl`(5).

```
# openssl pkcs7 -in pkcs7-file -print_certs
# ikev2cert import objtype=cert label=Partym1 infile=individual-cert
```

6. **Définissez la stratégie de validation du certificat.**

   Si le certificat contient des sections pour les CRL ou OCSP, vous devez configurer la stratégie de validation du certificat en fonction des exigences de votre site. Pour obtenir des instructions, reportez-vous à la section “Définition d'une stratégie de validation de certificats dans IKEv2” à la page 162

7. **Une fois que vous avez terminé la procédure sur tous les systèmes utilisant votre certificat IKE, activez le service `ikev2` sur tous les systèmes.**

   Les systèmes homologues ont besoin du certificat ancre sécurisée et d'un fichier `ikev2.config` configuré.

Étapes suivantes Si vous n'avez pas terminé d'établir la stratégie IPsec, effectuez de nouveau la procédure IPsec pour activer ou actualiser la stratégie IPsec. Pour obtenir des exemples de la stratégie IPsec de protection de VPN, reportez-vous à “Protection d'un VPN à l'aide d'IPsec” à la page 117. Pour plus d'exemples de la stratégie IPsec, reportez-vous à la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

▼ Définition d'une stratégie de validation de certificats dans IKEv2

Vous pouvez configurer plusieurs aspects de la manière dont les certificats sont gérés pour votre système IKEv2.

Avant de commencer

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Vous devez saisir dans un shell de profil. Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, "Configuration de la stratégie IPsec à distance via une connexion `ssh`" et à " Administration à distance de ZFS avec le shell sécurisé " du manuel " Gestion de l'accès au shell sécurisé dans Oracle Solaris 11.2 " pour des instructions sur la connexion à distance sécurisée.

1. Vérifiez la stratégie par défaut de validation des certificats.

La stratégie pour les certificats est réglée à l'installation dans le fichier `/etc/inet/ike/kmf-policy.xml`. Le fichier appartient à `ikeuser` et se modifie à l'aide de la commande `kmfcfg`. La stratégie par défaut de validation de certificats est de télécharger les LCR vers le répertoire `/var/user/ikeuser/crls`. L'utilisation d'OCSP est également activée par défaut. Si votre site requiert un proxy pour accéder à Internet, vous devez configurer le proxy. Reportez-vous à la section "Gestion des certificats révoqués dans IKEv2" à la page 164.

```
# pfbash
# kmfcfg list dbfile=/etc/inet/ike/kmf-policy.xml policy=default
Policy Name: default
Ignore Certificate Validity Dates: false      Unknown purposes or applications for the certificate
Ignore Unknown EKUs: false
Ignore Trust Anchor in Certificate Validation: false
Trust Intermediate CAs as trust anchors: false
Maximum Certificate Path Length: 32
Certificate Validity Period Adjusted Time leeway: [not set]
Trust Anchor Certificate: Search by Issuer
Key Usage Bits: 0     Identifies critical parts of certificate
Extended Key Usage Values: [not set]      Purposes or applications for the certificate
HTTP Proxy (Global Scope): [not set]
Validation Policy Information:
    Maximum Certificate Revocation Responder Timeout: 10
    Ignore Certificate Revocation Responder Timeout: true
    OCSP:
        Responder URI: [not set]
        OCSP specific proxy override: [not set]
        Use ResponderURI from Certificate: true
        Response lifetime: [not set]
        Ignore Response signature: false
        Responder Certificate: [not set]
    CRL:
        Base filename: [not set]
        Directory: /var/user/ikeuser/crls
        Download and cache CRL: true
        CRL specific proxy override: [not set]
        Ignore CRL signature: false
        Ignore CRL validity date: false
IPsec policy bypass on outgoing connections: true
Certificate to name mapper name: [not set]
Certificate to name mapper pathname: [not set]
```

```
Certificate to name mapper directory: [not set]
Certificate to name mapper options: [not set]
```

2. **Vérifiez le certificat pour les fonctionnalités qui indiquent les options de validation permettant d'apporter des modifications.**

   Par exemple, un certificat OCSP qui fait l'objet d'un URI CRL URI ou stratégie peut utiliser URI qui spécifie la validation à utiliser pour contrôler le statut de révocation de certificat. Vous pouvez également configurer des délais d'expiration.

3. **Passez en revue la page de manuel `kmfcfg(1)` pour des options configurables.**

4. **Configurer la stratégie de validation du certificat.**

   Pour consulter un exemple de stratégie, reportez-vous à “Gestion des certificats révoqués dans IKEv2” à la page 164.

▼ Gestion des certificats révoqués dans IKEv2

Les certificats révoqués est compromis s'effectue à l'aide de certificats pour quelque raison que ce soit. Un certificat révoqué en cours d'utilisation peut poser des problèmes de sécurité. Vous avez la possibilité lors de la vérification de la révocation de certificat. Vous pouvez utiliser une liste statique ou vous pouvez vérifier révocations dynamiquement sur le protocole HTTP.

Avant de commencer

Vous avez réceptionné et à partir d'un CA certificats installés.

Vous êtes familiarisé avec les méthodes OSCP CRL de recherche des et les certificats révoqués. Pour obtenir des informations et des conseils, reportez-vous à “IKE avec certificats de clés publiques” à la page 136.

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau) et utiliser un shell de profil. Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

1. **Localisez les sections LCR et OCSP dans le certificat reçu depuis l'AC.**

   Vous pouvez identifier le certificat à partir de l'étiquette de votre CSR.

```
# pfbash
# ikev2cert list objtype=cert | grep Label:
Enter PIN for Sun Software PKCS#11 softtoken:
        Label: Partym1
```

   Par exemple, la sortie tronquée suivante met en évidence les URI de la LCR et de l'OCSP dans un certificat.

```
# ikev2cert list objtype=cert label=Partym1
```

```
X509v3 extensions:
    ...
    X509v3 CRL Distribution Points:
         Full Name:
       URI:http://onsitecrl.PKI.example.com/OCCIPsec/LatestCRL.crl
    X509v3 Authority Key Identifier:
         ...
    Authority Information Access:
        OCSP - URI:http://ocsp.PKI.example.com/revokes/
    X509v3 Certificate Policies:
         Policy: 2.16.840.1.113733.1.7.23.2
```

Sous l'entrée `CRL Distribution Points`, l'URI indique que la LCR de cette organisation est disponible dans un fichier sur le Web. La OCSP saisie indique que le statut des différents certificats peuvent être déterminée de manière dynamique à partir d'un serveur.

2. **Activez l'utilisation de LCR ou d'un serveur OCSP en spécifiant un proxy.**

```
# kmfcfg modify \
dbfile=/etc/inet/ike/kmf-policy.xml \
policy=default \
http-proxy=www-proxy.ja.example.com:80
```

Sur les sites où un proxy est facultatif, vous n'avez pas besoin d'indiquer un vous-même .

3. **Stratégie assurez-vous que le certificat de validation est mis à jour.**

Par exemple, vérifiez que le OCSP a été mise à jour.

```
# kmfcfg list \
dbfile=/etc/inet/ike/kmf-policy.xml \
policy=default
...
    OCSP:
        Responder URI: [not set]
        Proxy: www-proxy.ja.example.com:80
        Use ResponderURI from Certificate: true
        Response lifetime: [not set]
        Ignore Response signature: false
        Responder Certificate: [not set]
```

4. **Redémarrez le service IKEv2.**

```
# svcadm restart ikev2
```

5. **(Facultatif) Ou OCSP arrêter d'utiliser des listes de certificats révoqués.**

   - **Pour arrêter d'utiliser les listes de révocation de certificats, saisissez :**

     ```
     # pfexec kmfcfg modify \
     dbfile=/etc/inet/ike/kmf-policy.xml policy=default \
     crl-none=true
     ```

L'argument `crl-none=true` force le système à utiliser les LCR téléchargés dans le cache local.

- **Pour arrêter d'utiliser OCSP, tapez :**

```
# pfexec kmfcfg modify \
dbfile=/etc/inet/ike/kmf-policy.xml policy=default \
ocsp-none=true
```

Exemple 9-4 Modification du temps d'attente d'un système pour la vérification de certificat IKEv2

Dans cet exemple, l'administrateur limite l'attente à vingt secondes pour la vérification de certificat.

```
# kmfcfg modify dbfile=/etc/inet/ike/kmf-policy.xml policy=default \
    cert-revoke-responder-timeout=20
```

Par défaut, lorsqu'une réponse arrive à expiration, l'authentification de l'homologue réussit. Ici, l'administrateur configure une stratégie selon laquelle la connexion est refusée quand l'authentification échoue. Dans cette configuration, la validation de certificat échoue quand un serveur OCSP ou CRL ne répond plus.

```
# kmfcfg modify dbfile=/etc/inet/ike/kmf-policy.xml policy=default \
    ignore-cert-revoke-responder-timeout=false
```

Pour activer la stratégie, l'administrateur redémarre le service IKEv2.

```
# svcadm restart ikev2
```

Génération et stockage de certificats de clés publiques pour IKEv2 dans le matériel

Les certificats de clés publiques peuvent également être stockés sur le matériel connecté. La carte Sun Crypto Accelerator 6000 permet de stocker et de décharger les opérations de clés publiques du système.

Le processus de génération et de stockage de certificats de clés publiques sur du matériel est similaire au processus de génération et de stockage de certificats de clés publiques sur un système. Sur le matériel, la commande `ikev2cert gencert token=`*hw-keystore* est utilisée pour identifier le keystore matériel.

Avant de commencer

Cette procédure suppose que la carte Sun Crypto Accelerator 6000 est connectée au système. Elle suppose aussi que le ou les logiciels correspondants ont été installés et que le keystore matériel est configuré. Pour obtenir des instructions, reportez-vous à la documentation de la carte Sun Crypto Accelerator 6000 dans la bibliothèque de produits (http://docs.oracle.com/cd/E19321-01/index.html). Ces instructions comprennent la définition du keystore.

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, "Configuration de la stratégie IPsec à distance via une connexion `ssh`" et à " Administration à distance de ZFS avec le shell sécurisé " du manuel " Gestion de l'accès au shell sécurisé dans Oracle Solaris 11.2 " pour des instructions sur la connexion à distance sécurisée.

1. **Confirmez que vous avez un ID de jeton pour la carte Sun Crypto Accelerator 6000 connectée.**

```
# pfbash
# ikev2cert tokens

Flags: L=Login required I=Initialized X=User PIN expired S=SO PIN expired
Slot ID  Slot Name                          Token Name                        Flags
-------  ---------                          ----------                        -----
1        sca6000                            sca6000                           LI
2        n2cp/0 Crypto Accel Bulk 1.0       n2cp/0 Crypto Accel Bulk 1.0
3        ncp/0 Crypto Accel Asym 1.0        ncp/0 Crypto Accel Asym 1.0
4        n2rng/0 SUNW_N2_Random_Number_Ge   n2rng/0 SUNW_N2_RNG
5        Sun Crypto Softtoken               Sun Software PKCS#11 softtoken    LI
```

2. **Générez un certificat autosigné ou une CSR et spécifiez l'ID de jeton.**

Remarque - Pour RSA, la carte Sun Crypto Accelerator 6000 prend en charge des clés d'une longueur maximum de 2 048 bits. Pour DSA, la longueur maximum des clés est de 1 024 bits.

Procédez de l'une des manières suivantes :

- **Pour un certificat autosigné, utilisez la syntaxe suivante :**

```
# ikev2cert gencert token=sca6000 keytype=rsa \
hash=sha256 keylen=2048 \
subject="CN=FortKnox, C=US" serial=0x6278281232 label=goldrepo
Enter PIN for sca6000:      See Step 3
```

- **Pour une demande de signature de certificat, utilisez la syntaxe suivante :**

```
# ikev2cert gencsr token=sca6000 -i
> keytype=
> hash=
> keylen=
> subject=
> serial=
> label=
> outcsr=
```

```
Enter PIN for sca6000 token:      See Step 3
```

Pour obtenir une description des arguments de la commande `ikev2cert`, reportez-vous à la page de manuel `pktool`(1).

3. **Lorsque vous êtes invité à saisir le PIN, entrez le nom de l'utilisateur de la carte Sun Crypto Accelerator 6000 suivi de deux-points et du mot de passe de l'utilisateur.**

Remarque - Vous devez connaître le nom d'utilisateur et le mot de passe du keystore.

Si la carte Sun Crypto Accelerator 6000 est configurée avec un utilisateur `admin` dont le mot de passe est `inThe%4ov`, tapez ce qui suit :

```
Enter PIN for sca6000 token: admin:inThe%4ov
-----BEGIN X509 CERTIFICATE-----
MIIBuDCCASECAQAwSTELMAkGA1UEBhMCVVMxFTATBgNVBAoTDFBhcnR5Q29tcGFu
…
oKUDBbZ9O/pLWYGr
-----END X509 CERTIFICATE-----
```

4. **Envoyez votre certificat aux personnes concernées.**

   Procédez de l'une des manières suivantes :

   - **Envoyez le certificat autosigné au système distant.**

     Vous pouvez le coller dans un e-mail.

   - **Envoyez la demande de signature de certificat à la CA.**

     Faire, suivez les instructions du CSR pour soumettre la CA. Pour plus d'informations, reportez-vous à “Utilisation de certificats de clés publiques dans IKE” à la page 137.

5. **Importez les certificats dans le keystore matériel.**

   Importez les certificats reçus depuis l'AC et fournissez l'utilisateur et le code PIN de l'Étape 3.

```
# ikev2cert import token=sca6000 infile=/tmp/DCA.ACCEL.CERT1
Enter PIN for sca6000 token:      Type user:password
# ikev2cert import token=sca6000 infile=/tmp/DCA.ACCEL.CA.CERT
Enter PIN for sca6000 token:      Type user:password
```

6. **Activer le keystore à être utilisé de manière automatique matériel ou de façon interactive.**

   La connexion automatique est recommandée. Si la stratégie de sécurité du site n'autorise pas la connexion automatique, vous devez vous connecter au keystore de manière interactive au redémarrage du démon `in.ikev2d`.

   - **Keystore configurer la connexion automatique à l'.**

a. **Ajoutez le PIN comme valeur de la propriété de service `pkcs11_token/uri`.**

   Pour obtenir une description de cette propriété, reportez-vous à "Service d'IKEv2" à la page 230.

   ```
   # svccfg -s ike:ikev2 editprop
   ```

   Fenêtre d'édition de temporaire s'ouvre.

b. **Décommentez la ligne `setprop pkcs11_token/uri =` et remplacez les parenthèses par le nom du jeton au format suivant :**

   ```
   # setprop pkcs11_token/uri = ()       Original entry
   setprop pkcs11_token/uri = pkcs11:token=sca6000
   ```

c. **Décommentez la ligne `setprop pkcs11_token/uri =` et remplacez les parenthèses par le *username:PIN* de l'Étape 3.**

   ```
   # setprop pkcs11_token/uri = ()       Original entry
   setprop pkcs11_token/uri = admin:PIN-from-Step-3
   ```

d. **Décommentez la ligne `refresh` à la fin du fichier, puis enregistrez les modifications.**

   ```
   # refresh
   refresh
   ```

e. **(Facultatif) Vérifiez la valeur des propriétés de `pkcs11_token`.**

   ```
   # svccfg -s ikev2 listprop pkcs11_token
   pkcs11_token/pin     astring      username:PIN
   pkcs11_token/uri     astring  pkcs11:token=sca6000
   ```

- **Si la connexion automatique n'est pas configuré, connectez-vous à plus d'informations sur la configuration matérielle keystore manuellement.**

  Exécutez cette commande chaque fois que le démon `in.ikev2d` démarre.

  ```
  # pfexec ikeadm -v2 token login sca6000
  Enter PIN for sca6000 token: admin:PIN-from-Step-3
  ikeadm: sca6000 operation successful
  ```

Étapes suivantes Si vous n'avez pas terminé d'établir la stratégie IPsec, effectuez de nouveau la procédure IPsec pour activer ou actualiser la stratégie IPsec. Pour obtenir des exemples de la stratégie IPsec de protection de VPN, reportez-vous à "Protection d'un VPN à l'aide d'IPsec" à la page 117. Pour plus d'exemples de la stratégie IPsec, reportez-vous à la section "Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec" à la page 111.

♦ ♦ ♦ CHAPITRE 10

Configuration d'IKEv1

Ce chapitre décrit la procédure de configuration du protocole Internet Key Exchange version 1 (IKEv1) sur vos systèmes. Une fois configuré, ce protocole génère automatiquement les numéros de clé IPsec sur votre réseau. Le présent chapitre contient les informations suivantes :

Remarque - Si vous prévoyez d'implémenter IKEv2 uniquement, passez au Chapitre 9, Configuration d'IKEv2.

Pour une présentation du protocole IKE, reportez-vous au Chapitre 8, A propos d'Internet Key Exchange. Pour obtenir des informations de référence sur IKE, reportez-vous au Chapitre 12, IPsec et référence de gestion des clés. Pour consulter d'autres procédures, reportez-vous aux sections Exemples des pages de manuel `ikeadm`(1M), `ikecert`(1M) et `ike.config`(4).

Remarque - #Les tâches suivantes partent du principe que les systèmes reçoivent des adresses IP statiques et que vous exécutez le profil de configuration réseau `DefaultFixed`. Si la commande `netadm list` renvoie `Automatic`, reportez-vous à la page de manuel `netcfg`(1M) pour plus d'informations.

Configuration d'IKEv1

L'authentification du protocole IKE peut s'effectuer à l'aide de clés prépartagées ou de certificats autosignés ou émanant d'autorités de certifications (AC). Une règle particulière dans le fichier `ike/config` lie la méthode d'authentification IKEv1 particulière au pair IKEv1. Vous pouvez donc utiliser toutes les méthodes d'authentification IKE ou une seule d'entre elles sur un système. Un pointeur menant à une bibliothèque PKCS #11 permet à IKEv1 d'utiliser un accélérateur matériel connecté.

Après avoir configuré IKEv1, réalisez la tâche IPsec dans le Chapitre 7, Configuration d'IPsec qui utilise la configuration IKEv1.

Configuration d'IKEv1 avec des clés prépartagées

Elles s'utilisent notamment lors de la configuration du protocole IKEv1 sur deux systèmes homologues ou sous-réseaux gérés par le même administrateur. Les clés prépartagées peuvent également être utilisé lors du test. Pour plus d'informations, reportez-vous à “IKE avec l'authentification des clés prépartagées” à la page 135

▼ Configuration d'IKEv1 avec des clés prépartagées

La longueur des clés des algorithmes d'implémentation du protocole IKE est variable. Elle dépend du niveau de sécurité dont vous souhaitez doter le site. En règle générale, la longueur des clés est proportionnelle au niveau de sécurité.

Dans cette procédure, vous générez des clés au format ASCII.

Les noms de systèmes choisis pour illustrer cette procédure sont : `enigma` et `partym`. Remplacez `enigma` et `partym` par les noms de vos systèmes.

Remarque - Pour utiliser IPsec avec des étiquettes sur un système Trusted Extensions, reportez-vous aux étapes supplémentaires indiquées à la section “ Application des protections IPsec dans un réseau TrustedÂ Extensions multiniveau ” du manuel “ Configuration et administration de Trusted Extensions ”.

Avant de commencer

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, “Configuration de la stratégie IPsec à distance via une connexion `ssh`” et à “ Administration à distance de ZFS avec le shell sécurisé ” du manuel “ Gestion de l’accès au shell sécurisé dans Oracle Solaris 11.2 ” pour des instructions sur la connexion à distance sécurisée.

1. **Sur chaque système, créez un fichier `/etc/inet/ike/config`.**

   Vous pouvez utiliser le fichier `/etc/inet/ike/config.sample` comme modèle.

2. **Entrez les règles et paramètres globaux dans le fichier `ike/config` sur chaque système.**

Les règles et paramètres globaux de ce fichier doivent garantir le fonctionnement de la stratégie IPsec du fichier `ipsecinit.conf`. Les exemples suivants de configuration d'IKEv1 fonctionnent avec les exemples d'`ipsecinit.conf` dans “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

a. **Modifiez par exemple le fichier `/etc/inet/ike/config` sur le système `enigma` :**

```
### ike/config file on enigma, 192.168.116.16

## Global parameters
#
## Defaults that individual rules can override.
p1_xform
  { auth_method preshared oakley_group 5 auth_alg sha encr_alg 3des }
p2_pfs 2
#
## The rule to communicate with partym
#  Label must be unique
{ label "enigma-partym"
  local_addr 192.168.116.16
  remote_addr 192.168.13.213
  p1_xform
   { auth_method preshared oakley_group 5 auth_alg sha256 encr_alg aes }
  p2_pfs 5
}
```

b. **Modifiez le fichier `/etc/inet/ike/config` sur le système `partym` :**

```
### ike/config file on partym, 192.168.13.213
## Global Parameters
#
p1_xform
  { auth_method preshared oakley_group 5 auth_alg sha encr_alg 3des }
p2_pfs 2

## The rule to communicate with enigma
#  Label must be unique
{ label "partym-enigma"
  local_addr 192.168.13.213
  remote_addr 192.168.116.16
p1_xform
 { auth_method preshared oakley_group 5 auth_alg sha256 encr_alg aes }
p2_pfs 5
}
```

3. **Sur chaque système, vérifiez la syntaxe du fichier.**

```
# /usr/lib/inet/in.iked -c -f /etc/inet/ike/config
```

4. **Mettez la clé prépartagée dans le fichier `/etc/inet/secret/ike.preshared` sur chaque système.**

a. Sur le système `enigma` par exemple, le fichier `ike.preshared` se présente comme suit :

```
## ike.preshared on enigma, 192.168.116.16
#…
{ localidtype IP
 localid 192.168.116.16
 remoteidtype IP
 remoteid 192.168.13.213
 # The preshared key can also be represented in hex
# as in 0xf47cb0f432e14480951095f82b
# key "This is an ASCII Cqret phrAz, use str0ng p@ssword tekniques"
}
```

b. Sur le système `partym`, le fichier `ike.preshared` se présente comme suit :

```
## ike.preshared on partym, 192.168.13.213
#…
{ localidtype IP
 localid 192.168.13.213
 remoteidtype IP
 remoteid 192.168.116.16
 # The preshared key can also be represented in hex
# as in 0xf47cb0f432e14480951095f82b
 key "This is an ASCII Cqret phrAz, use str0ng p@ssword tekniques"
 }
```

5. Activez le service IKEv1.

```
# svcadm enable ipsec/ike:default
```

Exemple 10-1 Actualisation d'une clé prépartagée IKEv1

Lorsqu'un administrateur IKEv1 souhaite actualiser la clé prépartagée, il modifie les fichiers sur ses systèmes homologues et redémarre le démon `in.iked`.

En première position, à tous les systèmes du les deux sous-réseaux qui utilise la clé prépartagée, l'administrateur modifie la clé prépartagée entrée.

```
# pfedit -s /etc/inet/secret/ike.preshared
…
{ localidtype IP
 localid 192.168.116.0/24
 remoteidtype IP
 remoteid 192.168.13.0/24
 # The two subnet's shared passphrase for keying material
key "LOooong key Th@t m^st Be Ch*angEd \'reguLarLy)"
 }
```

Ensuite, l'administrateur redémarre le service IKEv1 sur chaque système.

Pour plus d'informations sur les options de la commande `pfedit`, reportez-vous à la page de manuel `pfedit`(1M).

```
# svcadm enable ipsec/ike:default
```

Étapes suivantes Si vous n'avez pas terminé d'établir la stratégie IPsec, effectuez de nouveau la procédure IPsec pour activer ou actualiser la stratégie IPsec. Pour obtenir des exemples de la stratégie IPsec de protection de VPN, reportez-vous à “Protection d'un VPN à l'aide d'IPsec” à la page 117. Pour plus d'exemples de la stratégie IPsec, reportez-vous à la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

▼ Mise à jour d'IKEv1 pour un nouveau système homologue

Si vous ajoutez des entrées de stratégie IPsec à une configuration de travail entre des systèmes homologues, vous devez actualiser le service de stratégie IPsec. Il n'est pas nécessaire de reconfigurer ou de redémarrer IKEv1.

Si vous ajoutez un nouveau système homologue à la stratégie IPsec, vous devez modifier non seulement IPsec, mais aussi la configuration IKEv1.

Avant de commencer Vous avez mis à jour le fichier `ipsecinit.conf` et actualisé la stratégie IPsec pour les systèmes homologues.

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, “Configuration de la stratégie IPsec à distance via une connexion `ssh`” et à “ Administration à distance de ZFS avec le shell sécurisé ” du manuel “ Gestion de l’accès au shell sécurisé dans Oracle Solaris 11.2 ” pour des instructions sur la connexion à distance sécurisée.

1. **Créez une règle pour la gestion des clés par IKEv1, pour le nouveau système qui utilise IPsec.**

   a. **Par exemple, sur le système `enigma`, ajoutez la règle suivante au fichier `/etc/inet/ike/config` :**

   ```
   ### ike/config file on enigma, 192.168.116.16

   ## The rule to communicate with ada

   {label "enigma-to-ada"
   ```

```
 local_addr 192.168.116.16
 remote_addr 192.168.15.7
 p1_xform
 {auth_method preshared oakley_group 5 auth_alg sha256 encr_alg aes}
 p2_pfs 5
 }
```

b. Sur le système `ada`, ajoutez la règle suivante :

```
### ike/config file on ada, 192.168.15.7

## The rule to communicate with enigma

{label "ada-to-enigma"
 local_addr 192.168.15.7
 remote_addr 192.168.116.16
 p1_xform
 {auth_method preshared oakley_group 5 auth_alg sha256 encr_alg aes}
 p2_pfs 5
}
```

2. Créez une clé prépartagée IKEv1 pour les systèmes homologues.

a. Sur le système `enigma`, ajoutez les informations suivantes au fichier `/etc/inet/secret/ike.preshared` :

```
## ike.preshared on enigma for the ada interface
##
{ localidtype IP
  localid 192.168.116.16
  remoteidtype IP
  remoteid 192.168.15.7
  # enigma and ada's shared key
  key "Twas brillig and the slivey toves did *s0mEtHiNg* be CareFULL hEEEr"
}
```

b. Sur le système `ada`, ajoutez les informations suivantes au fichier `ike.preshared` :

```
## ike.preshared on ada for the enigma interface
##
{ localidtype IP
  localid 192.168.15.7
  remoteidtype IP
  remoteid 192.168.116.16
  # ada and enigma's shared key
  key "Twas brillig and the slivey toves did *s0mEtHiNg* be CareFULL hEEEr"
}
```

3. Sur chaque système, actualisez le service `ike`.

```
# svcadm refresh ike:default
```

Étapes suivantes Si vous n'avez pas terminé d'établir la stratégie IPsec, effectuez de nouveau la procédure IPsec pour activer ou actualiser la stratégie IPsec. Pour obtenir des exemples de la stratégie IPsec de protection de VPN, reportez-vous à “Protection d'un VPN à l'aide d'IPsec” à la page 117. Pour plus d'exemples de la stratégie IPsec, reportez-vous à la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

Configuration d'IKEv1 avec des certificats à clé publique

Grâce aux certificats de clés publiques, les systèmes communicants n'ont plus besoin de partager de numéros de clé secrets hors bande. Les certificats publics émanant d'autorités de certification (AC) requièrent une négociation avec une organisation externe. Ces certificats s'intègrent très facilement dans les environnements à grande échelle afin protéger un grand nombre de systèmes communicants.

Les certificats de clés publiques peuvent également être stockés dans le matériel connecté. Pour la procédure, reportez-vous à la section “Configuration du protocole IKEv1 en vue de l'utilisation du matériel connecté” à la page 202.

Tous les certificats possèdent un nom unique sous la forme d'un nom distinctif (ND) X.509. En outre, un certificat a peut-être au moins un domaine d'autres noms, adresse e-mail , par exemple, un, une adresse IP nom DNS, et ainsi de suite. Vous pouvez identifier le certificat dans la configuration IKEv1 par son ND complet ou par l'un des autres noms de l'objet. Le format de ces autres noms est *tag=value*, où le format de la valeur correspond à son type de balise. Par exemple, le format de la balise `email` est *name@domain.suffix*.

La liste de tâches suivantes décrit les procédures de création de certificats de clé publique pour IKEv1. Ces procédures incluent l'accélération et le stockage de certificats sur le matériel connecté.

TABLEAU 10-1 Liste de tâches pour la configuration d'IKEv1 avec des certificats de clé publique

Tâche	Description	Pour obtenir des instructions
Configuration du protocole IKEv1 avec des certificats de clés publiques autosignés.	Crée et place deux certificats sur chaque système : ■ Un certificat auto-signé et ses clés ■ Certificat de clé publique du système homologue	“Configuration d'IKEv1 avec des certificats de clés publiques autosignés” à la page 178
Configuration d'IKEv1 avec une autorité de certification.	Crée une demande de signature de certificat, puis place les certificats à partir de la CA sur chacun des systèmes. Reportez-vous à “Utilisation de certificats de clés publiques dans IKE” à la page 137.	“Configuration du protocole IKEv1 avec des certificats signés par une AC” à la page 183
Configuration de certificats de clés publiques sur le matériel local	Procédez de l'une des manières suivantes : ■ Générez un certificat autosigné sur le matériel local et ajoutez la clé publique d'un système distant sur le matériel.	“Génération et stockage de certificats de clés publiques pour IKEv1 dans le matériel” à la page 189

Tâche	Description	Pour obtenir des instructions
	■ Génération d'une demande de signature de certificat dans le matériel local et ajout de certificats de clés publiques de l'AC dans le matériel.	
Mise à jour de la liste des certificats révoqués (LCR) d'une AC.	Accédez à la CRL depuis un point de distribution central.	"Gestion des certificats révoqués dans IKEv1" à la page 192

Remarque - Pour étiqueter les paquets et les négociations IKE sur un système Trusted Extensions, effectuez les procédures décrites à la section " Configuration d'IPsec avec étiquettes " du manuel " Configuration et administration de Trusted Extensions ".

Les certificats de clés publiques sont gérés dans la zone globale sur les systèmes Trusted Extensions. Trusted Extensions ne change pas la manière dont les certificats sont gérés et stockés.

▼ Configuration d'IKEv1 avec des certificats de clés publiques autosignés

Dans cette procédure, vous créez une clé publique ou privée et un certificat, c'est ce l'on appelle une paire de certificats. La clé privée est stockée sur le disque, dans la base de données de certificats locale, et peut être référencée par la commande `ikecert certlocal`. La clé publique et le certificat est stocké dans la base de données des certificats publics. Elle peut être référencée par la commande `ikecert certdb`. Vous échangez le certificat public avec un système homologue. Les deux les certificats permettent d'authentifier les transmissions IKEv1.

Les certificats autosignés nécessitent un temps système inférieur à celui des certificats publics émanant d'une CA, mais s'intègrent plus difficilement dans un environnement à grande échelle. La différence des certificats émis par une CA, les certificats autosignés doivent être vérifiés aux administrateurs qui échangés par les deux les certificats.

Avant de commencer

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Pour plus d'informations, reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, "Configuration de la stratégie IPsec à distance via une connexion `ssh`" et à " Administration à distance de ZFS avec le shell sécurisé " du manuel " Gestion de l'accès au shell sécurisé dans Oracle Solaris 11.2 " pour des instructions sur la connexion à distance sécurisée.

1. **Sur chaque système IKEv1, créez un certificat autosigné dans la base de données `ike.privatekeys`.**

Pour les arguments de la commande `ikecert certlocal`, reportez-vous à la page de manuel `ikecert`(1M).

a. Par exemple, la commande du système `partym` apparaîtrait comme suit :

```
# ikecert certlocal -ks -m 2048 -t rsa-sha512 \
-D "O=exampleco, OU=IT, C=US, CN=partym" \
-A IP=192.168.13.213
Creating private key.
Certificate added to database.
-----BEGIN X509 CERTIFICATE-----
MIIC1TCCAb2gAwIBAgIEfdZgKjANBgkqhkiG9w0BAQUFADAaMRgwFgYDVQQDEw9T
a...+
zBGi4QkNdI3f
-----END X509 CERTIFICATE-----
```

avec

`-ks`	Crée un certificat autosigné.
`-m` *keysize*	Indique la taille de la clé.
`-t` *keytype*	Spécifie le type d'algorithme à utiliser.
`-D` *dname*	Spécifie le nom distinctif (ND) X.509 pour le sujet du certificat. Pour un exemple, reportez-vous à “Utilisation de certificats de clés publiques dans IKE” à la page 137.
`-A` *altname*	Nom ou un pseudonyme en spécifie l'du certificat. *altname* se présente de la manière suivante : *tag=value*. Les balises valides sont `IP`, `DNS`, `email` et `DN`.

Remarque - Les valeurs des options `-D` et `-A` sont des noms qui identifient uniquement le certificat, et non un système, tel que `192.168.13.213`. Il s'agit en réalité de surnoms de certificats, vous devez donc vérifier hors bande que le certificat correct est installé sur les systèmes homologues.

b. La commande sur le système `enigma` se présenterait de la manière suivante :

```
# ikecert certlocal -ks -m 2048 -t rsa-sha512 \
-D "O=exampleco, OU=IT, C=US, CN=enigma" \
-A IP=192.168.116.16
Creating private key.
Certificate added to database.
-----BEGIN X509 CERTIFICATE-----
MIIC1TCCAb2gAwIBAgIEBl5JnjANBgkqhkiG9w0BAQUFADAaMRgwFgYDVQQDEw9T
...
```

```
y85m6LHJYtC6
-----END X509 CERTIFICATE-----
```

2. **Enregistrez le certificat et envoyez-le au système distant.**

   La sortie est une version codée de la portion publique du certificat. Vous pouvez coller en toute sécurité ce certificat dans un e-mail. La partie réceptrice doit vérifier hors bande que le certificat correct est installé, comme indiqué à l'Étape 4.

   a. **Par exemple, vous transmettez la portion publique du certificat `partym` à l'administrateur de `enigma`.**

   ```
   To: admin@enigma.ja.example.com
   From: admin@party.us.example.com
   Message: -----BEGIN X509 CERTIFICATE-----
   MIIC1TCCAb2gAwIBAgIEfdZgKjANBgkqhkiG9w0BAQUFADAaMRgwFgYDVQQDEw9T
   a...+
   zBGi4QkNdI3f
   -----END X509 CERTIFICATE------
   ```

   b. **L'administrateur de `enigma` vous transmet la portion publique du certificat `enigma`.**

   ```
   To: admin@party.us.example.com
   From: admin@enigma.ja.example.com
   Message: ----BEGIN X509 CERTIFICATE-----
   MIIC1TCCAb2gAwIBAgIEBl5JnjANBgkqhkiG9w0BAQUFADAaMRgwFgYDVQQDEw9T
   ...
   y85m6LHJYtC6
   -----END X509 CERTIFICATE-----
   ```

3. **Sur chaque système, ajoutez le certificat reçu à la base de données des clés publiques.**

   a. **Enregistrez l'e-mail de l'administrateur dans un fichier lisible par `root`.**

   b. **Redirigez le fichier vers la commande `ikecert`.**

   ```
   # ikecert certdb -a < /tmp/certificate.eml
   ```

   La commande importe le texte entre les balises `BEGIN` et `END`.

4. **Assurez-vous auprès de l'autre administrateur que ce certificat est bien de lui.**

   Par exemple, vous pouvez téléphoner à l'autre administrateur afin de vérifier que le hachage de son certificat public, que vous possédez, correspond au hachage de son certificat privé, que seul lui possède.

   a. **Listez le certificat enregistré sur `partym`.**

Dans l'exemple suivant, la remarque 1 indique le nom distinctif (ND) du certificat dans l'emplacement 0. Le certificat privé de l'emplacement 0 possède le même hachage (reportez-vous à la remarque 3), ces certificats appartiennent donc à la même paire de certificats. Pour que les certificats publics fonctionnent, vous devez posséder une paire correspondante. La sous-commande `certdb` répertorie la portion publique, tandis que la sous-commande `certlocal` répertorie la portion privée.

```
partym # ikecert certdb -l

Certificate Slot Name: 0   Key Type: rsa
 (Private key in certlocal slot 0)
 Subject Name: <O=exampleco, OU=IT, C=US, CN=partym>     Note 1
 Key Size: 2048
 Public key hash: 80829EC52FC5BA910F4764076C20FDCF

Certificate Slot Name: 1   Key Type: rsa
 (Private key in certlocal slot 1)
 Subject Name: <O=exampleco, OU=IT, C=US, CN=Ada>
 Key Size: 2048
 Public key hash: FEA65C5387BBF3B2C8F16C019FEBC388
partym # ikecert certlocal -l
Local ID Slot Name: 0   Key Type: rsa
 Key Size: 2048
 Public key hash: 80829EC52FC5BA910F4764076C20FDCF     Note 3

Local ID Slot Name: 1   Key Type: rsa-sha512
        Key Size: 2048
        Public key hash: FEA65C5387BBF3B2C8F16C019FEBC388

Local ID Slot Name: 2   Key Type: rsa
 Key Size: 2048
 Public key hash: 2239A6A127F88EE0CB40F7C24A65B818
```

Cette vérification a permis de confirmer que le système `partym` possédait une paire de certificats valide.

b. Vérifiez que le système `enigma` possède le certificat public de `partym`.

Vous pouvez communiquer la valeur de hachage par téléphone.

Comparez les hachages de la remarque 3 sur `partym` indiquée dans l'étape précédente avec la remarque 4 sur `enigma`.

```
enigma # ikecert certdb -l

Certificate Slot Name: 0   Key Type: rsa
        (Private key in certlocal slot 0)
        Subject Name: <O=exampleco, OU=IT, C=US, CN=Ada>
        Key Size: 2048
        Public key hash: 2239A6A127F88EE0CB40F7C24A65B818

Certificate Slot Name: 1   Key Type: rsa
```

```
        (Private key in certlocal slot 1)
        Subject Name: <O=exampleco, OU=IT, C=US, CN=enigma>
        Key Size: 2048
        Public key hash: FEA65C5387BBF3B2C8F16C019FEBC388

Certificate Slot Name: 2   Key Type: rsa
        (Private key in certlocal slot 2)
        Subject Name: <O=exampleco, OU=IT, C=US, CN=partym>
        Key Size: 2048
        Public key hash: 80829EC52FC5BA910F4764076C20FDCF     Note 4
```

Le hachage de clé publique et le nom du sujet du dernier certificat stocké dans la base de données de certificats publics de `enigma` correspond au certificat privé de `partym` issu de l'étape précédente.

5. Sur chaque système, faites confiance aux deux certificats.

Editez le fichier `/etc/inet/ike/config` pour reconnaître les certificats.

L'administrateur du système distant fournit les valeurs des paramètres `cert_trust`, `remote_addr` et `remote_id`.

a. Sur le système `partym` par exemple, le fichier `ike/config` se présente comme suit :

```
# Explicitly trust the self-signed certs
# that we verified out of band. The local certificate
# is implicitly trusted because we have access to the private key.

cert_trust "O=exampleco, OU=IT, C=US, CN=enigma"
# We could also use the Alternate name of the certificate,
# if it was created with one.  In this example, the Alternate Name
# is in the format of an IP address:
# cert_trust "192.168.116.16"

## Parameters that may also show up in rules.

p1_xform
  { auth_method preshared oakley_group 5 auth_alg sha256 encr_alg 3des }
p2_pfs 5

{
 label "US-partym to JA-enigma"
 local_id_type dn
 local_id "O=exampleco, OU=IT, C=US, CN=partym"
 remote_id "O=exampleco, OU=IT, C=US, CN=enigma"

 local_addr  192.168.13.213
# We could explicitly enter the peer's IP address here, but we don't need
# to do this with certificates, so use a wildcard address. The wildcard
# allows the remote device to be mobile or behind a NAT box
 remote_addr 0.0.0.0/0
```

```
 p1_xform
  {auth_method rsa_sig oakley_group 2 auth_alg sha256 encr_alg aes}
}
```

b. Sur le système `enigma`, ajoutez les valeurs de `enigma` des paramètres locaux dans le fichier `ike/config`.

Pour les paramètres distants, utilisez les valeurs de `partym`. Assurez-vous que la valeur du mot-clé `label` est unique sur le système local.

```
…
{
 label "JA-enigma to US-partym"
 local_id_type dn
 local_id "O=exampleco, OU=IT, C=US, CN=enigma"
 remote_id "O=exampleco, OU=IT, C=US, CN=partym"

 local_addr  192.168.116.16
 remote_addr 0.0.0.0/0

 p1_xform
  {auth_method rsa_sig oakley_group 2 auth_alg sha256 encr_alg aes}
}
```

6. Sur les systèmes homologues, activez IKEv1.

```
partym # svcadm enable ipsec/ike:default
enigma # svcadm enable ipsec/ike
```

Étapes suivantes

Si vous n'avez pas terminé d'établir la stratégie IPsec, effectuez de nouveau la procédure IPsec pour activer ou actualiser la stratégie IPsec. Pour obtenir des exemples de la stratégie IPsec de protection de VPN, reportez-vous à “Protection d'un VPN à l'aide d'IPsec” à la page 117. Pour plus d'exemples de la stratégie IPsec, reportez-vous à la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

▼ Configuration du protocole IKEv1 avec des certificats signés par une AC

Avant de commencer

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, “Configuration de la stratégie IPsec à distance via une connexion `ssh`” et à “ Administration à distance de ZFS avec le shell

sécurisé ” du manuel “ Gestion de l’accès au shell sécurisé dans Oracle Solaris 11.2 ” pour des instructions sur la connexion à distance sécurisée.

1. **Utilisez la commande `ikecert certlocal -kc` pour créer une demande de signature de certificat (CSR, Certificate signing request).**

   Pour consulter la description des arguments de la commande, reportez-vous à l'Étape 1 de la section “Configuration d'IKEv1 avec des certificats de clés publiques autosignés” à la page 178.

   ```
   # ikecert certlocal -kc -m keysize -t keytype \
   -D dname -A altname
   ```

   a. **La commande suivante permet par exemple de créer un fichier contenant une CSR sur le système `partym` :**

   ```
   # ikecert certlocal -kc -m 2048 -t rsa-sha384 \
   > -D "C=US, O=PartyCompany\, Inc., OU=US-Partym, CN=Partym" \
   > -A "DN=C=US, O=PartyCompany\, Inc., OU=US-Partym"
   Creating software private keys.
     Writing private key to file /etc/inet/secret/ike.privatekeys/2.
   Enabling external key providers - done.
   Certificate Request:
     Proceeding with the signing operation.
     Certificate request generated successfully (…/publickeys/0)
   Finished successfully.
   -----BEGIN CERTIFICATE REQUEST-----
   MIIByjCCATMCAQAwUzELMAkGA1UEBhMCVVMxHTAbBgNVBAoTFEV4YW1wbGVDb21w
   …
   lcM+tw0ThRrfuJX9t/Qa1R/KxRlMA3zckO80mO9X
   -----END CERTIFICATE REQUEST-----
   ```

   b. **La commande suivante permet de créer une CSR sur le système `enigma` :**

   ```
   # ikecert certlocal -kc -m 2048 -t rsa-sha384 \
   > -D "C=JA, O=EnigmaCo\, Inc., OU=JA-Enigmax, CN=Enigmax" \
   > -A "DN=C=JA, O=EnigmaCo\, Inc., OU=JA-Enigmax"
   Creating software private keys.
   …
   Finished successfully.
   -----BEGIN CERTIFICATE REQUEST-----
   MIIBuDCCASECAQAwSTELMAkGA1UEBhMCVVMxFTATBgNVBAoTDFBhcnR5Q29tcGFu
   …
   8qlqdjaStLGfhDOO
   -----END CERTIFICATE REQUEST-----
   ```

2. **CA CSR à un soumettre le.**

   L'AC peut vous indiquer comment soumettre la CSR. Dans la plupart des cas, celle-ci s'effectue en remplissant un formulaire directement sur le site Web du fournisseur. Dans ce formulaire, vous devrez notamment indiquer la preuve de la légitimité de votre demande. Il suffit généralement de coller votre CSR dans le formulaire. Après avoir vérifié votre demande, le

fournisseur émet vous certificats signés. Pour plus d'informations, reportez-vous à “Utilisation de certificats de clés publiques dans IKE” à la page 137.

3. **Ajoutez chaque certificat à votre système.**

   L'option `-a` de la commande `ikecert certdb -a` ajoute l'objet collé à la base de données de certificats correspondante de votre système. Pour plus d'informations, reportez-vous à la section “IKE avec certificats de clés publiques” à la page 136.

   a. **Connexion en tant qu'administrateur.**

      Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”. Si vous administrez à distance, reportez-vous à l'Exemple 7-1, “Configuration de la stratégie IPsec à distance via une connexion `ssh`” et à “ Administration à distance de ZFS avec le shell sécurisé ” du manuel “ Gestion de l’accès au shell sécurisé dans Oracle Solaris 11.2 ” pour des instructions sur la connexion à distance sécurisée.

   b. **Clé publique et la ajouter le certificat que vous avez reçu de la CA.**

      ```
      # ikecert certdb -a < /tmp/PKIcert.eml
      ```

   c. **Ajoutez le certificat public d'CA.**

      Vous devrez peut-être également. pour ajouter des certificats intermédiaires

      ```
      # ikecert certdb -a < /tmp/PKIca.eml
      ```

   d. **Si l'AC vous a envoyé une liste des certificats révoqués, ajoutez-la à la base de données `certrldb` :**

      ```
      # ikecert certrldb -a
      Press the Return key
      Paste the CRL
      -----BEGIN CRL-----
      …
      -----END CRL----
      Press the Return key
      Press Control-D
      ```

4. **Utilisez le mot-clé `cert_root` dans le fichier `/etc/inet/ike/config` pour identifier l'AC qui a émis le certificat.**

   Utilisez le nom distinctif (DN) du certificat du CA.

   a. **Sur le système `partym`, par exemple, le fichier `ike/config` peut se présenter de la manière suivante :**

      ```
      # Trusted root cert
      ```

```
# This certificate is from Example CA
# This is the X.509 distinguished name for the CA's cert

cert_root "C=US, O=ExampleCA\, Inc., OU=CA-Example, CN=Example CA"

## Parameters that may also show up in rules.

p1_xform
 { auth_method rsa_sig oakley_group 1 auth_alg sha384 encr_alg aes}
p2_pfs 2

{
 label "US-partym to JA-enigma - Example CA"
 local_id_type dn
 local_id  "C=US, O=PartyCompany, OU=US-Partym, CN=Partym"
 remote_id "C=JA, O=EnigmaCo, OU=JA-Enigmax, CN=Enigmax"

 local_addr  192.168.13.213
 remote_addr 192.168.116.16

 p1_xform
  {auth_method rsa_sig oakley_group 2 auth_alg sha256 encr_alg aes}
}
```

Remarque - Tous les arguments du paramètre `auth_method` doivent se trouver sur la même ligne.

b. Créez un fichier similaire sur le système `enigma`.

Spécifiquement, le fichier `enigma` `ike/config` doit :

- Inclure la même valeur `cert_root`.
- Utiliser les valeurs de `enigma` pour les paramètres locaux.
- Utiliser les valeurs de `partym` pour les paramètres distants.
- Créer une valeur unique pour le mot-clé `label`. Elle doit différer de la valeur `label` du système distant.

```
…
cert_root "C=US, O=ExampleCA\, Inc., OU=CA-Example, CN=Example CA"
…
{
 label "JA-enigma to US-partym - Example CA"
 local_id_type dn
 local_id   "C=JA, O=EnigmaCo, OU=JA-Enigmax, CN=Enigmax"
 remote_id  "C=US, O=PartyCompany, OU=US-Partym, CN=Partym"

 local_addr  192.168.116.16
 remote_addr 192.168.13.213
…
```

5. Fixez les stratégies IKEv1 pour la gestion des certificats révoqués.

Choisissez l'option appropriée :

- **Aucune OCSP disponibles**
  Si le certificat de clé publique fournit un URI pour atteindre le serveur OCSP mais que votre système ne peut pas se connecter à Internet, ajoutez le mot-clé `ignore_ocsp` au fichier `ike/config`.

  ```
  # Trusted root cert
  …
  cert_root "C=US, O=ExampleCA\, Inc., OU=CA-Example,…
  ignore_ocsp
  …
  ```

  Le mot-clé `ignore_ocsp` dit à IKEv1 de partir du principe que le certificat est valide.

- **Pas de CRL disponible**
  Si le certificat de clé publique ne fournit pas de source fiable pour les LCR ou si votre système ne peut pas se connecter à Internet pour les récupérer, ajoutez le mot-clé `ignore_crls` au fichier `ike/config`.

  ```
  # Trusted root cert
  …
  cert_root "C=US, O=ExampleCA\, Inc., OU=CA-Example,…
  ignore_crls
  …
  ```

- **Disponible pour les CRL ou OCSP URI**
  Si l'AC communique un point de distribution central pour les certificats révoqués, vous pouvez modifier le fichier `ike/config` de façon à utiliser les URI.
  Pour des exemples, reportez-vous à “Gestion des certificats révoqués dans IKEv1” à la page 192.

Exemple 10-2 Utilisation de `rsa_encrypt` lors de la configuration d'IKEv1

Lorsque vous utilisez `auth_method rsa_encrypt` dans le fichier `ike/config`, vous devez ajouter le certificat du pair à la base de données `publickeys`.

1. Envoyez ce certificat à l'administrateur du système distant.
   Vous pouvez le coller dans un e-mail.
   Par exemple, l'administrateur de `partym` envoie le message suivant :

   ```
   To: admin@enigma.ja.example.com
   From: admin@party.us.example.com
   Message: -----BEGIN X509 CERTIFICATE-----
   MII…
   ----END X509 CERTIFICATE-----
   ```

L'administrateur de `enigma` envoie le message suivant :

```
To: admin@party.us.example.com
From: admin@enigma.ja.example.com
Message: -----BEGIN X509 CERTIFICATE-----
MII
…
-----END X509 CERTIFICATE-----
```

2. Sur chacun des systèmes, ajoutez à la base de données `publickeys` locale le certificat envoyé par e-mail.

```
# ikecert certdb -a < /tmp/saved.cert.eml
```

En cachant les identités à l'aide du protocole IKE, la méthode d'authentification utilisée en chiffrement RSA prévient les risques d'écoute électronique. Etant donné que la méthode `rsa_encrypt` cache l'identité du pair, IKEv1 ne peut pas récupérer son certificat. La méthode `rsa_encrypt` requiert donc que les pairs IKEv1 connaissent leurs clés publiques respectives.

C'est pourquoi vous devez ajouter le certificat du pair à la base de données `publickeys` lorsque vous utilisez la méthode `auth_method` de `rsa_encrypt` dans le fichier `/etc/inet/ike/config`. La base de données `publickeys` détient alors trois certificats pour chaque couple de systèmes communicants :

- Votre certificat de clé publique
- La chaîne de certificats CA du
- Le certificat de clé publique de l'homologue

Dépannage – la charge utile IKEv1, qui comprend au moins trois certificats, peut devenir trop volumineuse pour être chiffrée par `rsa_encrypt`. L'apparition d'erreurs indiquant que l'autorisation a échoué ou que la charge n'est pas conforme peut signifier que la méthode `rsa_encrypt` est incapable de chiffrer la totalité de la charge. Pour réduire la charge, utilisez une autre méthode (par exemple, `rsa_sig`, qui ne requiert que deux certificats).

Étapes suivantes Si vous n'avez pas terminé d'établir la stratégie IPsec, effectuez de nouveau la procédure IPsec pour activer ou actualiser la stratégie IPsec. Pour obtenir des exemples de la stratégie IPsec de protection de VPN, reportez-vous à “Protection d'un VPN à l'aide d'IPsec” à la page 117. Pour plus d'exemples de la stratégie IPsec, reportez-vous à la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

▼ Génération et stockage de certificats de clés publiques pour IKEv1 dans le matériel

Le processus de génération et de stockage de certificats de clés publiques sur du matériel est similaire au processus de génération et de stockage de certificats de clés publiques sur un système. Dans le premier cas, il convient d'identifier le matériel à l'aide des commandes `ikecert certlocal` et `ikecert certdb`, accompagnées de l'option `-T` et de l'ID de jeton.

Avant de commencer

- Le matériel doit être configuré.
- Excepté si le mot-clé `pkcs11_path` du fichier `/etc/inet/ike/config` pointe sur une autre bibliothèque, le matériel utilise `/usr/lib/libpkcs11.so`. Cette bibliothèque doit être implémentée conformément à la norme suivante : RSA Security Inc. PKCS #11 Cryptographic Token Interface (Cryptoki), c'est-à-dire une bibliothèque PKCS #11.

  Reportez-vous à “Configuration du protocole IKEv1 en vue de l'utilisation d'une carte Sun Crypto Accelerator 6000” à la page 202 pour les instructions de configuration.

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, “Configuration de la stratégie IPsec à distance via une connexion `ssh`” et à “ Administration à distance de ZFS avec le shell sécurisé ” du manuel “ Gestion de l’accès au shell sécurisé dans Oracle Solaris 11.2 ” pour des instructions sur la connexion à distance sécurisée.

1. **Générez un certificat autosigné ou une CSR et spécifiez l'ID de jeton.**

Remarque - Pour RSA, la carte Sun Crypto Accelerator 6000 prend en charge des clés d'une longueur maximum de 2 048 bits. Pour DSA, la longueur maximum des clés est de 1 024 bits.

Procédez de l'une des manières suivantes :

- **Pour un certificat autosigné, utilisez la syntaxe suivante :**

```
# ikecert certlocal -ks -m 2048 -t rsa-sha512 \
> -D "C=US, O=PartyCompany, OU=US-Partym, CN=Partym" \
> -a -T dca0-accel-stor IP=192.168.116.16
Creating hardware private keys.
Enter PIN for PKCS#11 token:      Type user:password
```

  L'argument de l'option `-T` est l'ID de jeton de la carte Sun Crypto Accelerator 6000 connectée.

- **Pour une CSR, utilisez la syntaxe suivante :**

```
# ikecert certlocal -kc -m 2048 -t rsa-sha512 \
> -D "C=US, O=PartyCompany, OU=US-Partym, CN=Partym" \
> -a -T dca0-accel-stor IP=192.168.116.16
Creating hardware private keys.
Enter PIN for PKCS#11 token:      Type user:password
```

Pour obtenir une description des arguments de la commande `ikecert`, reportez-vous à la page de manuel `ikecert`(1M).

2. **Lorsque vous êtes invité à saisir le PIN, entrez le nom de l'utilisateur de la carte Sun Crypto Accelerator 6000 suivi de deux-points et du mot de passe de l'utilisateur.**

   Si la carte Sun Crypto Accelerator 6000 possède un utilisateur `ikemgr` dont le mot de passe est `rgm4tigt`, entrez :

   ```
   Enter PIN for PKCS#11 token: ikemgr:rgm4tigt
   ```

   Remarque - Si vous saisissez les commandes `ikecert` avec l'option `-p`, le jeton PKCS #11 est enregistré sur le disque *en texte clair* et protégé par les permissions `root`. Si vous ne stockez pas le PIN sur le disque, vous devez déverrouiller le jeton à l'aide de la commande `ikeadm` une fois la commande `in.iked` en cours d'exécution.

   Après entrée du mot de passe, le certificat imprime la sortie suivante :

   ```
   Enter PIN for PKCS#11 token: ikemgr:rgm4tigt
   -----BEGIN X509 CERTIFICATE-----
   MIIBuDCCASECAQAwSTELMAkGA1UEBhMCVVMxFTATBgNVBAoTDFBhcnR5Q29tcGFu
   …
   oKUDBbZ9O/pLWYGr
   -----END X509 CERTIFICATE-----
   ```

3. **Envoyez votre certificat à l'autre partie.**

   Procédez de l'une des manières suivantes :

   - **Envoyez le certificat autosigné au système distant.**

     Vous pouvez le coller dans un e-mail.

   - **Envoyez la CSR à une autorité de certification (AC).**

     Pour ce faire, suivez les instructions de l'AC. Pour plus d'informations, reportez-vous à l'Étape 2 de la section “Configuration du protocole IKEv1 avec des certificats signés par une AC” à la page 183.

4. **Sur votre système, modifiez le fichier `/etc/inet/ike/config` pour qu'il reconnaisse les certificats.**

   Procédez de l'une des manières suivantes :

- **Certificat autosigné**

  Utilisez les valeurs fournies par l'administrateur du système distant pour les paramètres `cert_trust`, `remote_id` et `remote_addr`. Sur le système `enigma` par exemple, le fichier `ike/config` se présente comme suit :

```
# Explicitly trust the following self-signed certs
# Use the Subject Alternate Name to identify the cert

cert_trust "192.168.116.16"       Local system's certificate Subject Alt Name
cert_trust "192.168.13.213"       Remote system's certificate Subject Alt name

…
{
 label "JA-enigma to US-partym"
 local_id_type dn
 local_id "C=JA, O=EnigmaCo, OU=JA-Enigmax, CN=Enigmax"
 remote_id "C=US, O=PartyCompany, OU=US-Partym, CN=Partym"

 local_addr  192.168.116.16
 remote_addr 192.168.13.213

 p1_xform
  {auth_method rsa_sig oakley_group 2 auth_alg sha256 encr_alg aes}
}
```

- **Demande de certificat**

  Saisissez le nom fourni par l'AC comme valeur du mot-clé `cert_root`. Par exemple, le fichier `ike/config` sur le système `enigma` peut se présenter comme suit :

```
# Trusted root cert
# This certificate is from Example CA
# This is the X.509 distinguished name for the CA that it issues.

cert_root "C=US, O=ExampleCA\, Inc., OU=CA-Example, CN=Example CA"

…
{
 label "JA-enigma to US-partym - Example CA"
 local_id_type dn
 local_id "C=JA, O=EnigmaCo, OU=JA-Enigmax, CN=Enigmax"
 remote_id  "C=US, O=PartyCompany, OU=US-Partym, CN=Partym"

 local_addr  192.168.116.16
 remote_addr 192.168.13.213

 p1_xform
  {auth_method rsa_sig oakley_group 2 auth_alg sha256 encr_alg aes}
}
```

5. **Placez les certificats de l'autre partie sur le matériel.**

Répondez à la demande de PIN comme vous l'avez fait au cours de l'Étape 2.

Remarque - Vous *devez* ajouter les certificats de clés publiques au matériel connecté qui a généré votre clé privée.

- **Certificat auto-signé.**
  Ajoutez le certificat autosigné du système distant. Dans cet exemple, il est stocké dans le fichier `DCA.ACCEL.STOR.CERT`.

  ```
  # ikecert certdb -a -T dca0-accel-stor < DCA.ACCEL.STOR.CERT
  Enter PIN for PKCS#11 token:      Type user:password
  ```

  Si le certificat autosigné a utilisé `rsa_encrypt` comme valeur du paramètre `auth_method`, ajoutez le certificat de l'homologue au magasin du matériel.

- **Certificats d'une AC.**
  Ajoutez le certificat généré par l'AC suite à votre demande et le certificat de l'organisation. Vous devrez peut-être également. pour ajouter des certificats intermédiaires

  ```
  # ikecert certdb -a -T dca0-accel-stor < DCA.ACCEL.STOR.CERT
  Enter PIN for PKCS#11 token:      Type user:password

  # ikecert certdb -a -T dca0-accel-stor < DCA.ACCEL.STOR.CA.CERT
  Enter PIN for PKCS#11 token:      Type user:password
  ```

  Pour ajouter une liste des certificats révoqués de l'autorité de certification, reportez-vous à “Gestion des certificats révoqués dans IKEv1” à la page 192.

Étapes suivantes Si vous n'avez pas terminé d'établir la stratégie IPsec, effectuez de nouveau la procédure IPsec pour activer ou actualiser la stratégie IPsec. Pour obtenir des exemples de la stratégie IPsec de protection de VPN, reportez-vous à “Protection d'un VPN à l'aide d'IPsec” à la page 117. Pour plus d'exemples de la stratégie IPsec, reportez-vous à la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

▼ Gestion des certificats révoqués dans IKEv1

Les certificats révoqués est compromis s'effectue à l'aide de certificats pour quelque raison que ce soit. Un certificat révoqué en cours d'utilisation peut poser des problèmes de sécurité. Vous avez la possibilité lors de la vérification de la révocation de certificat. Vous pouvez utiliser une liste statique ou vous pouvez vérifier révocations dynamiquement sur le protocole HTTP. Il existe quatre manières de gérer les certificats révoqués.

- Vous pouvez demander à IKEv1 les CRL ou OCSP dont l'identificateur universel de ressources (URI) est intégré dans le certificat. Cette option est indiquée à l'Étape 5.

- Vous pouvez dire à IKEv1 d'accéder aux LCR ou OCSP depuis un URI dont l'adresse est intégrée dans le certificat de clé publique de l'AC.
- Vous pouvez dire à IKEv1 d'accéder aux LCR à partir d'un serveur LDAP dont le nom de répertoire (DN, directory name) est intégré au certificat de clé publique de l'AC.
- Vous pouvez fournir la LCR comme argument de la commande `ikecert certrldb`. Pour consulter un exemple, reportez-vous à l'Exemple 10-3, “Ajout d'une LCR à la base de données `certrldb` locale pour IKEv1”.

Avant de commencer

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

1. Affichez le certificat reçu depuis l'AC.

Pour plus d'information sur les arguments de la commande `ikecert certdb`, reportez-vous à la page de manuel `ikecert`(1M).

Par exemple, le certificat suivant a été émis par la PKI d'une entreprise. (les détails ont été modifiés).

```
# ikecert certdb -lv cert-protect.example.com
Certificate Slot Name: 0   Type: dsa-sha256
   (Private key in certlocal slot )
 Subject Name: <O=Example, CN=cert-protect.example.com>
 Issuer Name: <CN=ExampleCo CO (Cl B), O=Example>
 SerialNumber: 14000D93
   Validity:
      Not Valid Before: 2013 Sep 19th, 21:11:11 GMT
      Not Valid After:  2017 Sep 18th, 21:11:11 GMT
   Public Key Info:
      Public Modulus  (n) (2048 bits): C575A…A5
      Public Exponent (e) (  24 bits): 010001
   Extensions:
      Subject Alternative Names:
             DNS = cert-protect.example.com
      Key Usage: DigitalSignature KeyEncipherment
      [CRITICAL]
   CRL Distribution Points:
      Full Name:
         URI = #Ihttp://www.example.com/pki/pkismica.crl#i
         DN = <CN=ExampleCo CO (Cl B), O=Example>
      CRL Issuer:
      Authority Key ID:
      Key ID:              4F … 6B
      SubjectKeyID:        A5 … FD
      Certificate Policies
      Authority Information Access
```

Notez l'entrée `CRL Distribution Points`.

- L'entrée URI indique que la CRL de cette organisation est disponible sur le Web.
- L'entrée DN indique que la CRL est disponible sur un serveur LDAP. Après que le protocole IKE a accédé à la CRL, celle-ci est mise en cache en vue de futures utilisations.

Pour accéder à la CRL, vous devez tout d'abord accéder à un point de distribution.

2. **Choisissez l'une des méthodes suivantes pour accéder à la CRL depuis un point de distribution central.**

   - **Utilisez l'URI.**

     Ajoutez le mot-clé `use_http` au fichier `/etc/inet/ike/config` de l'hôte. Le fichier `ike/config` se présente comme suit :

     ```
     # Use CRL or OCSP from organization's URI
     use_http
     …
     ```

   - **Utilisez un proxy Web.**

     Ajoutez le mot-clé `proxy` au fichier `ike/config`. Le mot-clé `proxy` adopte une URL comme argument, comme indiqué ci-dessous :

     ```
     # Use web proxy to reach CRLs or OCSP
     proxy "http://proxy1:8080"
     ```

   - **Utilisez un serveur LDAP.**

     Utilisez le nom du serveur LDAP comme argument du mot-clé `ldap-list` dans le fichier `/etc/inet/ike/config` de l'hôte. Le nom du serveur LDAP est fourni par votre organisation. L'entrée dans le fichier `ike/config` se présente comme suit :

     ```
     # Use CRL from organization's LDAP
     ldap-list "ldap1.example.com:389,ldap2.example.com"
     …
     ```

   Le protocole IKE récupère la CRL et la met en cache jusqu'à ce que le certificat expire.

Exemple 10-3 Ajout d'une LCR à la base de données `certrldb` locale pour IKEv1

Si la LCR de l'AC n'est pas disponible à partir d'un point de distribution central, vous pouvez l'ajouter manuellement à la base de données `certrldb` locale. Pour extraire la LCR dans un fichier, suivez les instructions de l'AC, puis ajoutez la LRC à la base de données à l'aide de la commande `ikecert certrldb -a`.

```
# ikecert certrldb -a < ExampleCo.Cert.CRL
```

Configuration d'IKEv1 pour les systèmes portables

Les protocoles IPsec et IKE requièrent un ID unique pour identifier la source et la destination. Pour les systèmes portables hors site ne possédant pas d'adresse IP unique, vous devez utiliser un autre type d'ID permettant d'identifier un système de manière unique (par exemple, `DNS`, `DN` ou `email`).

Il est toujours préférable de configurer les systèmes portables ou hors site possédant une adresse IP unique avec un autre type d'ID. Par exemple, si ces systèmes tentent de se connecter à un site central par l'intermédiaire d'un boîtier NAT, leur adresse unique n'est pas utilisée. Le boîtier NAT leur assigne une adresse IP arbitraire que le système central ne reconnaît pas.

Les clés prépartagées ne sont pas, elles non plus, un moyen d'authentification approprié pour les systèmes portables, car elles requièrent une adresse IP fixe. Les certificats autosignés ou les certificats d'AC permettent par contre aux systèmes portables de communiquer avec le site central.

La liste de tâches suivante décrit les procédures permettant de configurer IKEv1 pour gérer des systèmes qui se connectent à distance à un site central.

TABLEAU 10-2 Liste des tâches pour la configuration d'IKEv1 pour les systèmes portables

Tâche	Description	Pour obtenir des instructions
Etablissement de la communication avec un site central depuis un lieu hors site	Permettez aux systèmes hors site de communiquer avec un site central. Ces systèmes peuvent être portables.	“Configuration d'IKEv1 pour les systèmes hors site” à la page 195
Utilisation d'un certificat public d'une AC et du protocole IKEv1 sur un système central acceptant le trafic des systèmes mobiles.	Configurez un système de passerelle pour accepter le trafic IPsec d'un système ne possédant pas d'adresse IP fixe.	Exemple 10-4, “Configuration d'un ordinateur central utilisant IKEv1 pour qu'il accepte le trafic protégé issu d'un système portable”
Utilisation d'un certificat public d'une AC et du protocole IKEv1 sur un système ne possédant pas d'adresse IP fixe.	Configurez le système portable de manière à protéger son trafic avec le site central (par exemple, le siège de l'entreprise).	Exemple 10-5, “Configuration d'un système situé derrière un NAT à l'aide d'IPsec et d'IKEv1”
Utilisation de certificats autosignés et du protocole IKEv1 sur un système central acceptant le trafic de systèmes portables.	Configurez un système de passerelle avec des certificats autosignés pour accepter le trafic IPsec d'un système portable.	Exemple 10-6, “Acceptation de certificats autosignés émanant d'un système portable”
Utilisation de certificats autosignés et du protocole IKEv1 sur un système ne possédant pas d'adresse IP fixe.	Configurez un système portable avec des certificats autosignés pour protéger son trafic avec un site central.	Exemple 10-7, “Utilisation de certificats autosignés pour contacter un système central”

▼ Configuration d'IKEv1 pour les systèmes hors site

Avant de commencer

Vous devez prendre le rôle `root`. Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”. Si vous administrez à distance, reportez-vous à l'Exemple 7-1,

"Configuration de la stratégie IPsec à distance via une connexion `ssh`" et à " Administration à distance de ZFS avec le shell sécurisé " du manuel " Gestion de l'accès au shell sécurisé dans Oracle Solaris 11.2 " pour des instructions sur la connexion à distance sécurisée.

1. Configurez le système central de manière à ce qu'il reconnaisse les systèmes portables.

a. Configurez le fichier `ipsecinit.conf`.

Le système central nécessite une stratégie autorisant une plage étendue d'adresses IP. Les certificats de la stratégie IKE garantissent ultérieurement la légitimité des systèmes connectés.

```
# /etc/inet/ipsecinit.conf on central
# Keep everyone out unless they use this IPsec policy:
{} ipsec {encr_algs aes encr_auth_algs sha256 sa shared}
```

b. Configurez le fichier de configuration IKEv1.

Le DNS identifie le système central et les certificats permettent d'authentifier le système.

```
## /etc/inet/ike/ike.config on central
# Global parameters
#
# Find CRLs by URI, URL, or LDAP
# Use CRL from organization's URI
use_http
#
# Use web proxy
proxy "http://somecache.domain:port/"
#
# Use LDAP server
ldap_server   "ldap-server1.domain.org,ldap2.domain.org:port"
#
# List CA-signed certificates
cert_root    "C=US, O=Domain Org, CN=Domain STATE"
#
# List self-signed certificates - trust server and enumerated others
#cert_trust    "DNS=central.domain.org"
#cert_trust    "DNS=mobile.domain.org"
#cert_trust    "DN=CN=Domain Org STATE (CLASS), O=Domain Org
#cert_trust    "email=root@central.domain.org"
#cert_trust    "email=user1@mobile.domain.org"
#

# Rule for mobile systems with certificate
{
  label "Mobile systems with certificate"
  local_id_type DNS
# CA's public certificate ensures trust,
# so allow any remote_id and any remote IP address.
  remote_id ""
  remote_addr 0.0.0.0/0
```

```
p2_pfs 5

p1_xform
{auth_method rsa_sig oakley_group 5 encr_alg aes auth_alg sha256 }
}
```

2. Connectez-vous à chacun des systèmes portables et configurez-les de manière à ce qu'ils trouvent le système central.

a. Configurez le fichier `/etc/hosts`.

Le fichier `/etc/hosts` n'a pas besoin d'une adresse pour le système portable, mais peut en fournir une. Il doit contenir une adresse IP publique pour le système central, *central*.

```
# /etc/hosts on mobile
central 192.xxx.xxx.x
```

b. Configurez le fichier `ipsecinit.conf`.

Le système portable doit être capable de trouver le système central à partir de son adresse IP publique. Les deux systèmes doivent avoir la même stratégie IPsec.

```
# /etc/inet/ipsecinit.conf on mobile
# Find central
{raddr 192.xxx.xxx.x} ipsec {encr_algs aes encr_auth_algs sha256 sa shared}
```

c. Configurez le fichier de configuration IKEv1.

L'identificateur ne peut pas être une adresse IP. Pour les systèmes portables, les identificateurs valides sont les suivants :

- `DN=`*ldap-directory-name*
- `DNS=`*domain-name-server-address*
- `email=`*email-address*

Les certificats permettent d'authentifier le système portable, *mobile*.

```
## /etc/inet/ike/ike.config on mobile
# Global parameters
#
# Find CRLs by URI, URL, or LDAP
# Use CRL from organization's URI
use_http
#
# Use web proxy
proxy "http://somecache.domain:port/"
#
# Use LDAP server
ldap_server   "ldap-server1.domain.org,ldap2.domain.org:port"
#
# List CA-signed certificates
```

```
cert_root    "C=US, O=Domain  Org, CN=Domain STATE"
#
# Self-signed certificates - trust me and enumerated others
#cert_trust    "DNS=mobile.domain.org"
#cert_trust    "DNS=central.domain.org"
#cert_trust    "DN=CN=Domain Org STATE (CLASS), O=Domain Org
#cert_trust    "email=user1@domain.org"
#cert_trust    "email=root@central.domain.org"
#
# Rule for off-site systems with root certificate
{
 label "Off-site mobile with certificate"
 local_id_type DNS

# NAT-T can translate local_addr into any public IP address
# central knows me by my DNS

 local_id "mobile.domain.org"
 local_addr 0.0.0.0/0

# Find central and trust the root certificate
 remote_id "central.domain.org"
 remote_addr 192.xxx.xxx.x

p2_pfs 5

p1_xform
{auth_method rsa_sig oakley_group 5 encr_alg aes auth_alg sha256 }
}
```

3. **Activez le service `ike:default`.**

```
# svcadm enable svc:/network/ipsec/ike:default
```

Exemple 10-4 Configuration d'un ordinateur central utilisant IKEv1 pour qu'il accepte le trafic protégé issu d'un système portable

Le protocole IKE peut commencer les négociations derrière un boîtier NAT, mais il est préférable de ne pas faire intervenir de boîtier de ce type. Dans l'exemple ci-dessous, le certificat public d'une CA a été placé sur le système mobile et sur le système central. Le système central accepte les négociations IPsec émanant d'un système situé derrière un boîtier NAT. `main1` est le système acceptant les connexions de systèmes hors site. Pour paramétrer les systèmes hors site, reportez-vous à l'Exemple 10-5, “Configuration d'un système situé derrière un NAT à l'aide d'IPsec et d'IKEv1”.

```
## /etc/hosts on main1
main1 192.168.0.100

## /etc/inet/ipsecinit.conf on main1
# Keep everyone out unless they use this IPsec policy:
{} ipsec {encr_algs aes encr_auth_algs sha256 sa shared}

## /etc/inet/ike/ike.config on main1
```

```
# Global parameters
#
# Find CRLs by URI, URL, or LDAP
# Use CRL from organization's URI
use_http
#
# Use web proxy
proxy "http://cache1.domain.org:8080/"
#
# Use LDAP server
ldap_server   "ldap1.domain.org,ldap2.domain.org:389"
#
# List CA-signed certificate
cert_root "C=US, O=ExampleCA Inc, OU=CA-Example, CN=Example CA"
#
# Rule for off-site systems with root certificate
{
  label "Off-site system with root certificate"
  local_id_type DNS
  local_id "main1.domain.org"
  local_addr 192.168.0.100

# CA's public certificate ensures trust,
# so allow any remote_id and any remote IP address.
  remote_id ""
  remote_addr 0.0.0.0/0

p2_pfs 5

p1_xform
{auth_method rsa_sig oakley_group 5 encr_alg aes auth_alg sha256}
p1_xform
{auth_method rsa_sig oakley_group 5 encr_alg aes auth_alg sha256}
p1_xform
{auth_method rsa_sig oakley_group 5 encr_alg aes auth_alg sha256}
p1_xform
{auth_method rsa_sig oakley_group 5 encr_alg aes auth_alg sha256}
}
```

Exemple 10-5 Configuration d'un système situé derrière un NAT à l'aide d'IPsec et d'IKEv1

Dans l'exemple ci-dessous, le certificat public d'une CA a été placé sur le système mobile et sur le système central. `mobile1` se connecte au siège de l'entreprise depuis son domicile. Le réseau du FAI (fournisseur d'accès Internet) utilise un boîtier NAT pour pouvoir assigner une adresse privée à `mobile1`. Le boîtier NAT convertit l'adresse privée en une adresse IP publique partagée par d'autres noeuds du réseau du FAI. Le siège de l'entreprise ne se trouve pas derrière un boîtier NAT. Pour paramétrer l'ordinateur du siège de l'entreprise, reportez-vous à l'Exemple 10-4, “Configuration d'un ordinateur central utilisant IKEv1 pour qu'il accepte le trafic protégé issu d'un système portable”.

```
## /etc/hosts on mobile1
mobile1 10.1.3.3
```

```
main1 192.168.0.100

## /etc/inet/ipsecinit.conf on mobile1
# Find main1
{raddr 192.168.0.100} ipsec {encr_algs aes encr_auth_algs sha256 sa shared}

## /etc/inet/ike/ike.config on mobile1
# Global parameters
#
# Find CRLs by URI, URL, or LDAP
# Use CRL from organization's URI
use_http
#
# Use web proxy
proxy "http://cache1.domain.org:8080/"
#
# Use LDAP server
ldap_server   "ldap1.domain.org,ldap2.domain.org:389"
#
# List CA-signed certificate
cert_root "C=US, O=ExampleCA Inc, OU=CA-Example, CN=Example CA"
#
# Rule for off-site systems with root certificate
{
  label "Off-site mobile1 with root certificate"
  local_id_type DNS
  local_id "mobile1.domain.org"
  local_addr 0.0.0.0/0

# Find main1 and trust the root certificate
  remote_id "main1.domain.org"
  remote_addr 192.168.0.100

p2_pfs 5

p1_xform
{auth_method rsa_sig oakley_group 5 encr_alg aes auth_alg sha256 }
}
```

Exemple 10-6 Acceptation de certificats autosignés émanant d'un système portable

Dans l'exemple ci-dessous, les certificats autosignés ont été émis par le système portable et le système central, et ont été placés sur les deux systèmes. `main1` est le système acceptant les connexions de systèmes hors site. Pour paramétrer les systèmes hors site, reportez-vous à l'Exemple 10-7, “Utilisation de certificats autosignés pour contacter un système central”.

```
## /etc/hosts on main1
main1 192.168.0.100

## /etc/inet/ipsecinit.conf on main1
# Keep everyone out unless they use this IPsec policy:
{} ipsec {encr_algs aes encr_auth_algs sha256 sa shared}

## /etc/inet/ike/ike.config on main1
```

```
# Global parameters
#
# Self-signed certificates - trust me and enumerated others
cert_trust    "DNS=main1.domain.org"
cert_trust    "jdoe@domain.org"
cert_trust    "user2@domain.org"
cert_trust    "user3@domain.org"
#
# Rule for off-site systems with trusted certificate
{
  label "Off-site systems with trusted certificates"
  local_id_type DNS
  local_id "main1.domain.org"
  local_addr 192.168.0.100

# Trust the self-signed certificates
# so allow any remote_id and any remote IP address.
  remote_id ""
  remote_addr 0.0.0.0/0

p2_pfs 5

p1_xform
{auth_method rsa_sig oakley_group 5 encr_alg aes auth_alg sha256 }
}
```

Exemple 10-7 Utilisation de certificats autosignés pour contacter un système central

Dans l'exemple ci-dessous, `mobile1` se connecte au siège de l'entreprise depuis un domicile privé. Les certificats ont été émis par le système portable et le système central, et ont été placés sur les deux systèmes. Le réseau du FAI utilise un boîtier NAT pour assigner une adresse privée à `mobile1`. Le boîtier NAT convertit l'adresse privée en une adresse IP publique partagée par d'autres noeuds du réseau du FAI. Le siège de l'entreprise ne se trouve pas derrière un boîtier NAT. Pour paramétrer l'ordinateur du siège de l'entreprise, reportez-vous à l'Exemple 10-6, “Acceptation de certificats autosignés émanant d'un système portable”.

```
## /etc/hosts on mobile1
mobile1 10.1.3.3
main1 192.168.0.100

## /etc/inet/ipsecinit.conf on mobile1
# Find main1
{raddr 192.168.0.100} ipsec {encr_algs aes encr_auth_algs sha256 sa shared}

## /etc/inet/ike/ike.config on mobile1
# Global parameters

# Self-signed certificates - trust me and the central system
cert_trust    "jdoe@domain.org"
cert_trust    "DNS=main1.domain.org"
#
# Rule for off-site systems with trusted certificate
{
```

```
  label "Off-site mobile1 with trusted certificate"
  local_id_type email
  local_id "jdoe@domain.org"
  local_addr 0.0.0.0/0

# Find main1 and trust the certificate
  remote_id "main1.domain.org"
  remote_addr 192.168.0.100

p2_pfs 5

p1_xform
{auth_method rsa_sig oakley_group 5 encr_alg aes auth_alg sha256 }
}
```

Étapes suivantes Si vous n'avez pas terminé d'établir la stratégie IPsec, effectuez de nouveau la procédure IPsec pour activer ou actualiser la stratégie IPsec. Pour obtenir des exemples de la stratégie IPsec de protection de VPN, reportez-vous à “Protection d'un VPN à l'aide d'IPsec” à la page 117. Pour plus d'exemples de la stratégie IPsec, reportez-vous à la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

Configuration du protocole IKEv1 en vue de l'utilisation du matériel connecté

Les certificats de clés publiques peuvent également être stockés sur le matériel connecté. La carte Sun Crypto Accelerator 6000 permet de stocker et de décharger les opérations de clés publiques du système.

▼ Configuration du protocole IKEv1 en vue de l'utilisation d'une carte Sun Crypto Accelerator 6000

Avant de commencer La procédure suivante suppose que la carte Sun Crypto Accelerator 6000 est connectée au système. et que le ou les logiciels correspondants ont été installés et configurés. Pour obtenir des instructions, reportez-vous à la documentation de la carte Sun Crypto Accelerator 6000 dans la bibliothèque de produits (http://docs.oracle.com/cd/E19321-01/index.html).

Vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

Si vous administrez à distance, reportez-vous à l'Exemple 7-1, “Configuration de la stratégie IPsec à distance via une connexion `ssh`” et à “ Administration à distance de ZFS avec le shell sécurisé ” du manuel “ Gestion de l’accès au shell sécurisé dans Oracle Solaris 11.2 ” pour des instructions sur la connexion à distance sécurisée.

1. **Vérifiez que la bibliothèque PKCS #11 est liée.**

   IKEv1 utilise les routines de la bibliothèque pour gérer la génération des clés et leur stockage sur la carte Sun Crypto Accelerator 6000.

```
$ ikeadm get stats
…
PKCS#11 library linked in from /usr/lib/libpkcs11.so
$
```

2. **Trouvez l'ID de jeton pour la carte Sun Crypto Accelerator 6000 connectée.**

```
$ ikecert tokens
Available tokens with library "/usr/lib/libpkcs11.so":

"Sun Metaslot                    "
```

   La bibliothèque renvoie un ID de jeton, également appelé nom du keystore, de 32 caractères. Dans l'exemple ci-dessous, vous pouvez utiliser le jeton `Sun Metaslot` avec la commande `ikecert` pour stocker et accélérer les clés IKEv1.

   Pour plus d'informations sur l'utilisation du jeton, reportez-vous à la section “Génération et stockage de certificats de clés publiques pour IKEv1 dans le matériel” à la page 189.

   Les espaces situés à la fin sont automatiquement remplis par la commande `ikecert`.

Exemple 10-8 Découverte et utilisation de jetons metaslot

Les jetons peuvent être stockés sur le disque, sur une carte connectée ou dans le keystore softtoken fourni par la structure cryptographique. L'ID de jeton du keystore de softtoken peut se présenter comme suit :

```
$ ikecert tokens
Available tokens with library "/usr/lib/libpkcs11.so":

"Sun Metaslot                  "
```

Pour créer une phrase de passe pour un keystore de softtoken, reportez-vous à la page de manuel `pktool`(1).

La commande ci-dessous permet d'ajouter un certificat au keystore de softtoken. `Sun.Metaslot.cert` est le fichier contenant le certificat CA.

```
# ikecert certdb -a -T "Sun Metaslot" < Sun.Metaslot.cert
Enter PIN for PKCS#11 token:      Type user:passphrase
```

Étapes suivantes Si vous n'avez pas terminé d'établir la stratégie IPsec, effectuez de nouveau la procédure IPsec pour activer ou actualiser la stratégie IPsec. Pour obtenir des exemples de la stratégie IPsec de protection de VPN, reportez-vous à “Protection d'un VPN à l'aide d'IPsec” à la page 117. Pour plus d'exemples de la stratégie IPsec, reportez-vous à la section “Sécurisation du trafic entre deux serveurs réseau à l'aide d'IPsec” à la page 111.

♦ ♦ ♦ CHAPITRE 11

Dépannage d'IPsec et de sa configuration de gestion des clés

Ce chapitre explique comment résoudre les problèmes liés à IPsec et à ses clés, comment afficher les informations de configuration et comment afficher des informations sur les services de clés manuels, IPsec et IKE.

Ce chapitre contient les informations suivantes :

Dépannage d'IPsec et de sa configuration de gestion des clés

Vous pouvez configurer le système à des fins de dépannage avant ou pendant un problème qui nécessite le dépannage.

Lors du dépannage, vous pouvez exécuter de nombreuses commandes d'un shell de profil en tant qu'administrateur disposant du profil de droits Network IPsec Management. Cependant, pour consulter les journaux vous devez prendre le rôle `root`.

Les invites dans les opérations de dépannage sections indiquer si vous devez disposer de droits permettant de exécution d'une commande.

- Invite `#` – Un utilisateur avec les droits d'administration appropriés ou un rôle ayant ces droits peut exécuter la commande.
- Invite `%` – Un utilisateur standard peut exécuter la commande.

▼ Préparation des systèmes IPsec et IKE pour la résolution de problèmes

Avant d'activer IPsec et ses services de gestion des clés, vous pouvez configurer votre système avec des journaux et des outils qui facilitent le dépannage.

1. **Localisez les journaux des services IPsec et IKEv2.**

   L'option `-L` fournit le chemin d'accès complet aux journaux. Ces journaux contiennent à la fois les messages d'information et les messages d'erreur.

   ```
   % svcs -L policy
   /var/svc/log/network-ipsec-policy:default.log

   % svcs -L ikev2
   /var/svc/log/network-ipsec-ike:ikev2.log
   ```

2. **Configurez un fichier journal de débogage pour IKEv2.**

   Le rôle `root` peut lire ces journaux.

   ```
   % svccfg -s ikev2 listprop | grep debug
   config/debug_level             astring     op
   config/debug_logfile           astring     /var/log/ikev2/in.ikev2d.log
   ```

   Les niveaux de débogage sont décrits à la page de manuel `ikeadm`(1M). Les valeurs `verbose` et `all` sont utiles lors du dépannage.

3. **(Facultatif) Configurez le niveau de débogage.**

   La commande suivante vous permet de définir le niveau de débogage de manière permanente. Pour définir temporairement le niveau de débogage, reportez-vous à l'Exemple 11-3, “Définition d'un nouveau niveau de débogage sur un démon IKE en cours d'exécution”.

   ```
   # svccfg -s ikev2 setprop config/debug_level = all
   ```

   Si le service `ikev2` est activé, il doit être actualisé pour utiliser le nouveau niveau de débogage.

   ```
   # svcadm refresh ikev2
   ```

4. **(Facultatif) Installez le package `wireshark`.**

   L'application Wireshark peut lire la sortie de `snoop`.

   ```
   % pkg info -r wireshark
             Name: diagnostic/wireshark
          Summary: Graphical network protocol analyzer
         Category: Applications/Internet
            State: Not installed
        Publisher: solaris
   ```

```
...
          FMRI: pkg://solaris/diagnostic/wireshark@version
# pkg install diagnostic/wireshark
```

Dépannage de systèmes avant l'exécution d'IPsec et IKE

Vous pouvez vérifier la syntaxe du fichier de configuration IPsec, le fichier de clés IPsec et la validité des certificats dans le keystore avant d'exécuter les services.

1. **Vérifiez la syntaxe du fichier de configuration IPsec.**

```
# ipsecconf -c /etc/inet/ipsecinit.conf
ipsecconf: Invalid pattern on line 5: ukp
ipsecconf: form_ipsec_conf error
ipsecconf: Malformed command (fatal):
{ ukp 58 type 133-137 dir out} pass {}

ipsecconf: 1 policy rule(s) contained errors.
ipsecconf: Fatal error - exiting.
```

Si la sortie affiche une erreur, résolvez le problème et exécutez la commande jusqu'à ce que la vérification a réussi avec succès.

2. **Vérifiez la syntaxe du fichier `ipseckeys`.**

```
# ipseckey -c /etc/inet/secret/ipseckeys
Config file /etc/inet/secret/ipseckeys has insecure permissions,
will be rejected in permanent config.
```

Si la sortie affiche une erreur, corrigez l'erreur, puis que vous actualisez le service.

```
# svcadm refresh ipsec/policy
```

Remarque - Les fichiers de configuration IKE et les clés prépartagées IKE sont validés en exécutant le démon IKE.

3. **Vérifiez la validité des certificats.**

- Pour vérifier la validité de certificats auto-signés dans IKEv2, réalisez l'Étape 4 dans “Configuration d'IKEv2 avec des certificats de clés publiques autosignés” à la page 153.
- Pour vérifier qu'un certificat de clé publique n'est pas révoqué dans IKEv2, suivez la procédure “Définition d'une stratégie de validation de certificats dans IKEv2” à la page 162.
- Pour vérifier la validité de certificats auto-signés dans IKEv1, réalisez l'Étape 4 dans “Configuration d'IKEv1 avec des certificats de clés publiques autosignés” à la page 178.

- Pour vérifier qu'un certificat de clé publique n'est pas révoqué dans IKEv1, suivez la procédure “Gestion des certificats révoqués dans IKEv1” à la page 192.

Étapes suivantes Si votre configuration ne fonctionne pas lorsque vous activez IPsec et ses services de génération de clés, vous devez résoudre les problèmes lorsque les services sont en cours d'exécution.

▼ Dépannage des systèmes lorsqu'IPsec est en cours d'exécution

Sur les systèmes en cours d'exécution qui échangent ou tentent d'échanger des paquets à l'aide d'IKE, la commande `ikeadm` permet d'afficher les statistiques, les règles, les clés prépartagées et d'autres choses. Vous pouvez également utiliser les fichiers journaux et des outils comme sélectionné Wireshark application.

1. **Analysez les éléments suivants :**

   - **Assurez-vous que `policy` et les services de gestion de clés appropriés sont activés.**

     Sur le système test suivant, le service `manual-key` est utilisé pour gérer les clés :

     ```
     % svcs -a  | grep ipsec
     online         Feb_04   svc:/network/ipsec/manual-key:default
     online         Feb_04   svc:/network/ipsec/ipsecalgs:default
     online         Feb_04   svc:/network/ipsec/policy:default
     disabled       Feb_28   svc:/network/ipsec/ike:ikev2
     disabled       Feb_28   svc:/network/ipsec/ike:default
     ```

     Si le service est désactivé, activez-le .

     Vous pouvez utiliser à la fois des services IKE via le programme de traitements simultanés. Vous pouvez également utiliser simultanément de clés manuelles, et cette configuration IKE, il est possible que vous obteniez mais qui ne figurent pas dans oddities dépannage.

   - **Affichez la fin du fichier journal du service IKEv2.**

     ```
     # svcs -xL ikev2
     svc:/network/ipsec/ike:ikev2 (IKEv2 daemon)
      State: disabled since October  10, 2013 10:10:40 PM PDT
     Reason: Disabled by an administrator.
        See: http://support.oracle.com/msg/SMF-8000-05
        See: in.ikev2d(1M)
        See: /var/svc/log/network-ipsec-ike:ikev2.log
     Impact: This service is not running.
        Log:
     Oct 01 13:20:20: (1)  Property "debug_level" set to: "op"
     Oct 01 13:20:20: (1)  Errors and debug messages will be written to:
     ```

```
                        /var/log/ikev2/in.ikev2d.log
[ Oct 10 10:10:10 Method "start" exited with status 0. ]
[ Oct 10 10:10:40 Stopping because service disabled. ]
[ Oct 10 10:10:40 Executing stop method (:kill). ]

   Use: 'svcs -Lv svc:/network/ipsec/ike:ikev2' to view the complete log.
```

- **(Facultatif) Vous pouvez définir une valeur temporaire pour le niveau de débogage du démon en cours d'exécution.**

```
# ikeadm set debug verbose /var/log/ikev2/in.ikev2d.log
Successfully changed debug level from 0x80000000 to 0x6204
Debug categories enabled:
        Operational / Errors
        Config file processing
        Interaction with Audit
        Verbose Operational
```

2. **Vérifiez que la sortie de la commande `ipsecconf` correspond au contenu du fichier de stratégie.**

```
# ipsecconf
#INDEX 14
...
{  laddr 10.133.66.222 raddr 10.133.64.77 }
 ipsec   { encr_algs aes(256) encr_auth_algs sha512 sa shared }
...
{  laddr 10.134.66.122 raddr 10.132.55.55 }
 ipsec   { encr_algs aes(256) encr_auth_algs sha512 sa shared }

# cat /etc/inet/ipsecinit.conf
...
{  laddr 10.133.66.222 raddr 10.133.64.77 }
 ipsec   { encr_algs aes(256) encr_auth_algs sha512 sa shared }

{  laddr 10.134.66.122 raddr 10.132.55.55 }
 ipsec   { encr_algs aes(256) encr_auth_algs sha512 sa shared }
```

Remarque - Les adresses à caractère générique peuvent cacher une correspondance, alors vérifiez que toutes les adresses spécifiques du fichier `ipsecinit.conf` sont dans la plage des adresses à caractère générique de la sortie d'`ipsecconf`.

Si aucun résultat n'est imprimé pour la commande`ipsecconf`, assurez-vous que la stratégie service est activé et actualisez le service.

```
% svcs policy
STATE          STIME    FMRI
online         Apr_10   svc:/network/ipsec/policy:default
```

Si la sortie affiche une erreur, modifiez le fichier `ipsecinit.conf` pour corriger l'erreur puis actualisez le service.

3. **Validez la configuration IKEv2.**

   Pour les sorties de configuration qui ont éventuellement besoin d'être réparées, reportez-vous aux exemples suivants : Exemple 11-1, “Correction d'une configuration IKEv2 non valide” et Exemple 11-2, “Correction en cas de message signalant l'absence de règle correspondante”. La sortie dans l'exemple suivant indique que la configuration est correcte.

```
# /usr/lib/inet/in.ikev2d -c
Feb 04 12:08:25: (1)    Reading service properties from smf(5) repository.
Feb 04 12:08:25: (1)    Property "config_file" set to: "/etc/inet/ike/ikev2.config"
Feb 04 12:08:25: (1)    Property "debug_level" set to: "all"
Feb 04 12:08:25: (1)    Warning: debug output being written to stdout.
Feb 04 12:08:25: (1)    Checking IKE rule #1: "Test 104 to 113"
Feb 04 12:08:25: (1)    Configuration file /etc/inet/ike/ikev2.config is valid.
Feb 04 12:08:25: (1)    Pre-shared key file /etc/inet/ike/ikev2.preshared is valid.
```

Remarque - La boîte de dialogue de l'avertissement sur la sortie de débogage ne change pas même une fois que vous avez indiqué un fichier journal des diagnostics. Si vous spécifiez une valeur pour la propriété de service `debug_logfile`, l'avertissement signifie que la sortie de débogage est transmise à ce fichier. Dans le cas contraire, la sortie de débogage, est concédé sous licence au console.

- Dans les lignes `Checking IKE rule`, assurez-vous que les règles IKE connectent les adresses IP correctes. Par exemple, les entrées suivantes pour qu'il y ait correspondance. La valeur `laddr` du fichier `ipsecinit.conf` correspond à la valeur `local_addr` du fichier `ikev2.config` et les adresses distantes correspondent.

```
{  laddr 10.134.64.104 raddr 10.134.66.113 }      /** ipsecinit.conf **/
                ipsec {encr_algs aes encr_auth_algs sha512 sa shared}


 local_addr   10.134.64.104                         /** ikev2.config **/
 remote_addr  10.134.66.113                         /** ikev2.config **/
```

  Si les entrées ne correspondent pas, corrigez-les pour identifier le bon de configuration des adresses IP.

Remarque - Les règles peuvent comporter des adresses à caractère générique, comme par exemple `10.134.0.0/16`, qui couvrent une plage d'adresses. Vérifiez la plage dans des adresses particulières.

- Si la ligne `Pre-shared key file` indique que le fichier n'est pas valide, réparez le fichier.

  Vérifiez si la saisie comporte des erreurs. Dans IKEv2, assurez-vous aussi que la valeur de l'étiquette dans `ikev2.config` correspond à celle du fichier `ikev2.preshared`. Ensuite, si vous utilisez deux local des clés prépartagées, vérifiez que le système effectue le rapprochement entre clé sur une clé prépartagée sur son pair distant et que le système à distance, la clé correspond à la clé local sur le pair.

Si votre configuration ne fonctionne toujours pas, reportez-vous à la section “Dépannage des erreurs sémantiques dans IPsec et IKE” à la page 212

Exemple 11-1 Correction d'une configuration IKEv2 non valide

Dans la sortie suivante, la durée de vie du SA IKE est trop court.

```
# /usr/lib/inet/in.ikev2d -c
...
May 08 08:52:49: (1) WARNING: Problem in rule "Test 104 to 113"
May 08 08:52:49: (1)  HARD lifetime too small (60 < 100)
May 08 08:52:49: (1)   -> Using 100 seconds (minimum)
May 08 08:52:49: (1) Checking IKE rule #1: "config 10.134.13.113 to 10.134.13.104"
...
```

Cette valeur est définie explicitement dans le fichier `ikev2.config`. Pour supprimer l'avertissement, augmentez la valeur du délai à au moins `100` et actualisez le service.

```
# pfedit /etc/inet/ike/ikev2.config
...
## childsa_lifetime_secs   60
childsa_lifetime_secs   100
...
# /usr/lib/inet/in.ikev2d -c
...
# svcadm refresh ikev2
```

Exemple 11-2 Correction en cas de message signalant l'absence de règle correspondante

Dans la sortie suivante, une clé prépartagée est définie mais n'est pas utilisée dans une règle.

```
# /usr/lib/inet/in.ikev2d -c
Feb 4 12:58:31: (1)  Reading service properties from smf(5) repository.
Feb 4 12:58:31: (1)  Property "config_file" set to: "/etc/inet/ike/ikev2.config"
Feb 4 12:58:31: (1)  Property "debug_level" set to: "op"
Feb 4 12:58:31: (1)  Warning: debug output being written to stdout.
Feb 4 12:58:31: (1)  Checking IKE rule #1: "Test 104 to 113"
Feb 4 12:58:31: (1)  Configuration file /etc/inet/ike/ikev2.config is valid.
Feb 4 12:58:31: (1)  No matching IKEv2 rule for pre-shared key ending on line 12
Feb 4 12:58:31: (1)  Pre-shared key file /etc/inet/ike/ikev2.preshared is valid.
```

La sortie indique qu'une seule règle existe.

- Si la règle requiert une clé prépartagée, alors l'étiquette de la clé prépartagée ne correspond pas à l'étiquette de la règle. Réparez l'étiquette de règle `ikev2.config` et l'étiquette de clé `ikev2.preshared` de façon qu'elles correspondent.
- Si la règle utilise un certificat, vous pouvez supprimer ou commenter la clé prépartagée à la ligne 12 du fichier `ikev2.preshared` pour éviter l'affichage du message `No matching`.

Exemple 11-3 Définition d'un nouveau niveau de débogage sur un démon IKE en cours d'exécution

Dans la sortie suivante, la sortie de débogage est définie sur `all` dans le service `ikev2`.

```
# /usr/lib/inet/in.ikev2d -c
Feb 4 12:58:31: (1)  Reading service properties from smf(5) repository.
...
Feb 4 12:58:31: (1)  Property "debug_level" set to: "all"
...
```

Si vous avez réalisé l'Étape 2 dans “Dépannage de systèmes avant l'exécution d'IPsec et IKE” à la page 207 et que la sortie de débogage est toujours `op` au lieu de `all`, utilisez la commande `ikeadm` pour fixer le niveau de débogage du démon IKE en cours d'exécution.

```
# ikeadm set debug_level all
```

Dépannage des erreurs sémantiques dans IPsec et IKE

Si les examens dans “Dépannage des systèmes lorsqu'IPsec est en cours d'exécution” à la page 208 ne résolvent pas le problème, le problème se situe probablement au niveau de la sémantique de configuration plutôt que dans la syntaxe de vos fichiers ou la configuration des services.

- Si les deux instances de service `ike:default` et `ike:ikev2` sont activées, assurez-vous que les règles IKEv2 et IKEv1 ne se chevauchent pas. Règles qui s'appliquent à la, peut rendre la même redondantes aux extrémités de réseaux les SA IPsec et la connectivité est susceptible de générer un manque de dans certaines situations.

  Règle si vous modifiez une règle IKE, reportez-vous à la rubrique dans le noyau.

  ```
  # ikeadm -v[1|2] read rule
  ```

- Si vous exécutez IKEv1, assurez-vous que les mécanismes d'algorithme utilisés dans vos règles sont disponibles sur le système IKEv1 auquel vous vous connectez. Pour visualiser les algorithmes disponibles, exécutez la commande `ikeadm dump` *algorithms* sur le système ne prenant pas en charge IKEv2 :

  ```
  # ikeadm dump groups      Available Diffie-Hellman groups
  # ikeadm dump encralgs    All IKE encryption algorithms
  # ikeadm dump authalgs    All IKE authentication algorithms
  ```

  Corrigez les deux fichiers de stratégie IPsec et IKEv1 afin d'utiliser des algorithmes disponibles sur les deux systèmes. Redémarrez ensuite le service IKEv1 et actualisez le service IPsec.

  ```
  # svcadm restart ike:default; svcadm refresh ipsec/policy
  ```

- Si vous utilisez IKEv1 avec des clés prépartagées et que le système IKEv1 distant est réinitialisé, exécutez la commande `ipseckey flush` sur le système local.
- Si vous utilisez des certificats autosignés, vérifiez auprès de l'autre administrateur que un certificat avec le même n'a pas fait l'objet d'une nouvelle DN créé et que les valeurs de hachage de vos certificats pour qu'il y ait correspondance. Pour les étapes de la vérification, reportez-vous à l'Étape 4 dans “Configuration d'IKEv2 avec des certificats de clés publiques autosignés” à la page 153.

  Si le certificat est mis à jour, importez le nouveau certificat, puis actualisez et redémarrez le service IKEv2.
- Utilisez la commande `ikeadm -v2 dump | get` pour afficher la configuration actuelle d'IKEv2. Pour un résumé de l'utilisation, reportez-vous à “Affichage des informations IKE” à la page 214.
- Utilisez la commande `kstat` pour afficher les statistiques IPsec. Pour plus d'informations, reportez-vous à la page de manuel `kstat`(1M).

```
# kstat -m ipsecesp
# kstat -m ipsecah
# kstat -m ip
```

La sortie `kstat` dans l'exemple suivant signifie qu'il n'y a aucun problème dans le module `ipsecesp`.

```
# kstat -m ipsecesp
module: ipsecesp                        instance: 0
name:   esp_stat                        class:    net
        acquire_requests                18
        bad_auth                        0
        bad_decrypt                     0
        bad_padding                     0
        bytes_expired                   0
        crtime                          4.87974774
        crypto_async                    0
        crypto_failures                 0
        crypto_sync                     172
        good_auth                       86
        keysock_in                      135
        num_aalgs                       9
        num_ealgs                       13
        out_discards                    0
        out_requests                    86
        replay_early_failures           0
        replay_failures                 0
        sa_port_renumbers               0
        snaptime                        5946769.7947628
```

- Utilisez la commande `snoop` pour afficher le trafic non protégé. L'application Wireshark peut lire la sortie de `snoop`. Pour obtenir un exemple de sortie `snoop`, reportez-vous à “Vérification de la protection des paquets par IPsec” à la page 131.

Affichage des informations relatives à IPsec et ses services de génération de clés

Remarque - Pour la plupart des commandes, vous devez vous connecter en tant qu'administrateur disposant du profil de droits Network IPsec Management (gestion IPsec du réseau). Vous devez saisir dans un shell de profil. Pour plus d'informations, reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

Affichage des propriétés d'IPsec et du service de clés manuel

Vous pouvez visualiser le nom du fichier de stratégie IPsec et le fichier qui contient de clés manuelles.

- Pour afficher le nom du fichier de configuration IPsec :

```
% svccfg -s policy listprop config/config_file
config/config_file         astring     /etc/inet/ipsecinit.conf
```

- Pour afficher le nom du fichier contenant les clés manuelles pour IPsec :

```
% svccfg -s manual-key listprop config/config_file
config/config_file         astring     /etc/inet/secret/ipseckeys
```

Affichage des informations IKE

Vous pouvez visualiser les propriétés du service IKE, certains aspects de l'état IKE et de l'objet de démon IKE et la stratégie de validation de certificat. IKE si vous exécutez les deux services, vous pouvez afficher des informations par service ou à la fois pour les services. Ces commandes peuvent vous être utiles lors du test, à des fins de dépannage, et la surveillance.

- Affichage des propriétés des instances des services IKE – La sortie affiche les propriétés confiigurables du service IKEv2 y compris les noms des fichiers de configuration.

Remarque - Passez en revue les pages de manuel`ipsecconf`(1M). `in.ikev2d`(1M) et `in.iked`(1M) pour vous assurer que vous pouvez ou devez modifier une propriété dans le groupe `config` des services IPsec, IKEv2 ou IKEv1. Par exemple, les fichiers de configuration IKEv2 sont créés avec des permissions spéciales et appartiennent à `ikeuser`. Les droits d'accès et propriétaire du fichier ne doivent pas être modifiés.

```
% svccfg -s ipsec/ike:ikev2 listprop config
config                      application
config/allow_keydump       boolean     false
config/config_file         astring     /etc/inet/ike/ikev2.config
config/ignore_errors       boolean     false
config/kmf_policy          astring     /etc/inet/ike/kmf-policy.xml
config/max_child_sas       integer     0
config/max_threads         integer     0
config/min_threads         integer     0
config/preshared_file      astring     /etc/inet/ike/ikev2.preshared
config/response_wait_time  integer     30
config/value_authorization astring     solaris.smf.value.ipsec
config/debug_logfile       astring
config/debug_level         astring     op
```

La sortie dans l'exemple suivant illustre les propriétés configurables du service IKEv1. Ne spécifiez pas l'instance de service `:default`.

```
% svccfg -s ipsec/ike listprop config
config                      application
config/admin_privilege     astring     base
config/config_file         astring     /etc/inet/ike/config
config/debug_level         astring     op
config/debug_logfile       astring     /var/log/in.iked.log
config/ignore_errors       boolean     false
config/value_authorization astring     solaris.smf.value.ipsec
```

- Affichage de l'état actuel du démon IKE – La sortie de l'exemple suivant affiche les arguments de la commande `ikeadm`. Ces arguments afficher l'état en cours du démon.

Remarque - Pour utiliser la commande `ikeadm`, le démon IKE doit être en cours d'exécution.

```
% ikeadm help
...
        get   debug|priv|stats|p1|ikesa|rule|preshared|defaults [identifier]
        dump  p1|ikesa|rule|preshared|certcache|groups|encralgs|authalgs
```

```
        read  rule|preshared [filename]
        help  [get|set|add|del|dump|flush|read|write|token|help]
```

- Affichage de la syntaxe d'un argument particulier de la commande `ikeadm` – Utilisez les sous-commandes `help` pour voir la syntaxe des arguments de la commande. Ainsi,

```
% ikeadm help read
This command reads a new configuration file into
in.iked, discarding the old configuration info.

Sets of data that may be read include:
        rule            all phase 1/ikesa rules
        preshared       all preshared keys

A filename may be provided to specify a source file
other than the default.
```

- Affichage des clés prépartagées – Vous pouvez afficher les clés prépartagées pour IKEv1 et IKEv2.

Remarque - Si vous exécutez une seule version d'IKE, vous pouvez omettre l'option `-v`.

Pour IKEv2 :

```
# ikeadm -v2 dump preshared
```

Pour IKEv1 :

```
# ikeadm set priv keymat
# ikeadm -v1 dump preshared

PSKEY: Rule label: "Test PSK 197 to 56"
PSKEY: Local pre-shared key (80 bytes): 74206272696c6c696720...3/584
PSKEY: Remote pre-shared key (80 bytes): 74206272696c6c696720...3/584

Completed dump of preshared keys
```

- Affichage des AS IKE – La sortie comporte des informations sur les AS, la transformation, les systèmes locaux et distants ainsi que d'autres détails. En cas de communication n'a pas été demandé. par conséquent, aucun EdC existe, il n'existe aucune information à afficher.

```
# ikeadm -v2 dump ikesa
IKESA: SPIs: Local 0xd3db95689459cca4  Remote 0xb5878717f5cfa877
...
XFORM: Encryption alg: aes-cbc(256..256); Authentication alg: hmac-sha512
...
LOCIP: AF_INET: port 500, 10.1.2.3 (example-3).
...
```

```
REMIP: AF_INET: port 500, 10.1.4.5 (ex-2).
...
LIFTM: SA expires in 11459 seconds (3.18 hours)
...
STATS: 0 IKE SA rekeys since initial AUTH.
LOCID: Initiator identity, type FQDN
...
CHILD: ESP Inbound SPI: 0x94841ca3, Outbound SPI 0x074ae1e5
...
Completed dump of IKE SA info
```

- Affichage des règles IKE actives – Une règle IKE répertoriée peut ne pas être en cours d'utilisation, mais disponible.

```
# ikeadm -v2 dump rule

GLOBL: Label 'Test Rule1 for PSK', key manager cookie 1
GLOBL: Local auth method=pre-shared key
GLOBL: Remote auth method=pre-shared key

GLOBL: childsa_pfs=false
GLOBL: authentication_lifetime=86400 seconds (1.00 day)
GLOBL: childsa_lifetime=120 seconds (2.00 minutes)
GLOBL: childsa_softlife=108 seconds (1.80 minute)
GLOBL: childsa_idletime=60 seconds
GLOBL: childsa_lifetime_kb=122880 kilobytes (120.00 MB)
GLOBL: childsa_softlife_kb=110592 kilobytes (108.00 MB)
LOCIP: IP address range(s):
LOCIP: 10.142.245.197
REMIP: IP address range(s):
REMIP: 10.134.64.56
LOCID: Identity descriptors:
LOCID: Includes:
LOCID:       fqdn="gloria@ms.mag"
REMID: Identity descriptors:
REMID: Includes:
REMID:       fqdn="gloria@ms.mag"
XFRMS: Available Transforms:

XF  0: Encryption alg: aes-cbc(128..256); Authentication alg: hmac-sha512
XF  0: PRF: hmac-sha512 ; Diffie-Hellman Group: 2048-bit MODP (group 14)
XF  0: IKE SA lifetime before rekey: 14400 seconds (4.00 hours)

Completed dump of policy rules
```

- Affichage de la stratégie de validation de certificat dans IKEv2 – Vous devez spécifier les valeurs `dbfile` et `policy`.

- Les listes des certificats révoqués, il peut s'avérer nécessaire de façon dynamique que l'administrateur ait à intervenir téléchargé pour ajuster le répondeur de temporisation.

  Dans la sortie de l'exemple suivant, les CRL sont téléchargés à partir de l'URI intégrée au certificat, puis les listes sont mises en mémoire cache. Lorsque la mémoire cache contient un CRL expiré, un nouveau CRL est téléchargé pour remplacer l'ancien.

```
# kmfcfg list dbfile=/etc/inet/ike/kmf-policy.xml policy=default
…
Validation Policy Information:
    Maximum Certificate Revocation Responder Timeout: 10
    Ignore Certificate Revocation Responder Timeout: true
…
    CRL:
        Base filename: [not set]
        Directory: /var/user/ikeuser/crls
        Download and cache CRL: true
        CRL specific proxy override: www-proxy.cagate.example.com:80
        Ignore CRL signature: false
        Ignore CRL validity date: false
IPsec policy bypass on outgoing connections: true
…
```

- Les LCR téléchargées de manière statique demandent des interventions fréquentes de la part de l'administrateur.

  Lorsque l'administrateur définit les entrées de la liste de révocation de certificats sur les valeurs suivantes, il est responsable du téléchargement manuel de ces listes, du remplissage de l'annuaire et de la maintenance des listes de révocation de certificats actuelles :

```
…
        Directory: /var/user/ikeuser/crls
        Download and cache CRL: false
        Proxy: [not set]
…
```

Gestion d'IPsec et de ses services de génération de clés

La stratégie IPsec est activée par défaut, mais des informations de configuration manquent à l'appel.

La gestion des clés n'est pas activée par défaut. Vous pouvez configurer la gestion des clés IKE ou manuelle, ou les deux. Chaque règle IKE indique le service de gestion des clés utilisé. La commande `ikeadm` peut modifier uniquement le démon IKE en cours d'exécution.

Configuration et gestion d'IPsec et de ses services de génération de clés

- Configuration et actualisation d'IPsec, puis affichage de la stratégie :

```
# pfedit /etc/inet/ipsecinit.conf
# ipsecconf -c /etc/inet/ipsecinit.conf
# svcadm refresh ipsec/policy
# ipsecconf -Ln
```

- Configuration et activation des clés manuelles pour IPsec :

```
# pfedit -s /etc/inet/secret/ipseckeys
# svcadm enable ipsec/manual-key
```

- Configuration et d'activation d'IKEv2 :

```
# pfedit /etc/inet/ike/ikev2.config
# /usr/lib/inet/in.ikev2d -c
# svcadm enable ipsec/ike:ikev2
```

- Configuration et d'activation d'IKEv1 :

```
# pfedit /etc/inet/ike/config
# /usr/lib/inet/in.iked -c
# svcadm enable ipsec/ike:default
```

- Vérification qu'IPsec et IKE sont configurés sur un système où les services sont activés :

```
# ipsecconf -Ln
# ikeadm -v2 dump rule
# ikeadm set priv keymat
# ikeadm -v1 dump rule
```

- Modification de la gestion des clés :

  Pour IKEv2 :

```
# pfedit /etc/inet/ike/ikev2.config
# /usr/lib/inet/in.ikev2d -c
# svcadm restart ipsec/ike:ikev2
```

  Pour IKEv1 :

```
# pfedit /etc/inet/ike/config
# /usr/lib/inet/in.iked -c
# svcadm restart ipsec/ike:default
```

  Pour la gestion des clés manuelle :

```
# pfedit -s /etc/inet/secret/ipseckeys
# ipseckey -c /etc/inet/secret/ipseckeys
```

```
# svcadm refresh ipsec/manual-key
```

- Modification des propriétés IPsec et IKE configurables :

  Service IPsec :

  ```
  # svccfg -s ipsec/policy setprop config/property = value
  # svcadm refresh ipsec/policy; svcadm restart ipsec/policy
  ```

  Service IKEv2 :

  ```
  # svccfg -s ike:ikev2 editprop
  # svcadm refresh ipsec/ike:ikev2; svcadm restart ipsec/ike:ikev2
  ```

  Service IKEv1 :

  ```
  # svccfg -s ipsec/ike setprop config/property = value
  # svcadm refresh ipsec/ike:ikev2; svcadm restart ipsec/ike:ikev2
  ```

  Service de clés manuel :

  ```
  # svccfg -s ipsec/manual-key setprop config/property = value
  # svcadm refresh ipsec/manual-key; svcadm restart ipsec/manual-key
  ```

- Configuration de clés prépartagées pour IKEv2 :

  ```
  # pfedit -s /etc/inet/ike/ikev2.preshared
  # /usr/lib/inet/in.ikev2d -c
  # svcadm restart ikev2
  ```

- Configuration de clés prépartagées pour IKEv1 :

  ```
  # pfedit -s /etc/inet/secret/ike.preshared
  # svcadm restart ike
  ```

Gestion des démons IKE en cours d'exécution

Pour plus d'informations, consultez la page de manuel `ikeadm`(1M). Les commandes de cette section ne sont disponibles que lorsque le démon IKEv2 ou IKEv1 est en cours d'exécution.

- Modification du démon IKE en cours d'exécution :

  La sortie suivante affiche les arguments de la commande `ikeadm` qui peuvent modifier l'état actuel du démon. Certains arguments sont spécifiques au démon IKEv2 ou IKEv1.

  ```
  % ikeadm help
  ...
          set   priv level
          set   debug level [filename]
  ```

```
add   rule|preshared {definition}|filename
del   p1|ikesa|rule|preshared identifier
flush p1|ikesa|certcache
write rule|preshared filename
token login|logout PKCS#11-Token-Object
```

- Affichage de la syntaxe d'un argument particulier de la commande `ikeadm` :

```
% ikeadm help add
This command adds items to in.iked's tables.

Objects that may be set include:
        rule            a phase 1 or IKE SA policy rule
        preshared       a preshared key

Objects may be entered on the command-line, as a
series of keywords and tokens contained in curly
braces ('{', '}'); or the name of a file containing
the object definition may be provided.

For security purposes, preshared keys may only be
entered on the command-line if ikeadm is running in
interactive mode.
```

- Modification du démon IKEv2 à l'aide de la commande `ikeadm` :

```
# ikeadm add rule | preshared {definition} | filename
# ikeadm flush ikesa
# ikeadm del ikesa | rule | preshared identifier
# ikeadm set debug level
# ikeadm token login | logout PKCS#11-Token-Object
# ikeadm write rule | preshared filename
```

- Modification du démon IKEv1 à l'aide de la commande `ikeadm` :

```
# ikeadm set debug level
# ikeadm set privlevel
# ikeadm add rule | preshared {definition} | filename
# ikeadm del p1 | rule | preshared identifier
# ikeadm flush p1 | certcache
# ikeadm del rule | preshared id
# ikeadm write rule | preshared filename
```

♦ ♦ ♦ CHAPITRE 12

IPsec et référence de gestion des clés

Ce chapitre contient des informations de référence sur IPsec, IKEv2 et IKEv1.

Pour obtenir les instructions relatives à l'implémentation d'IPsec sur votre réseau, reportez-vous au Chapitre 7, Configuration d'IPsec. Pour une présentation d'IPsec, reportez-vous au Chapitre 6, A propos de l'architecture de sécurité IP.

Pour des instructions sur l'implémentation d'IKE, reportez-vous au Chapitre 9, Configuration d'IKEv2. Pour obtenir une présentation, reportez-vous au Chapitre 8, A propos d'Internet Key Exchange.

Référence IPsec

Services, fichiers et commandes IPsec

Cette section répertorie les services IPsec, les RFC IPsec sélectionnés, ainsi que les fichiers et commandes propres à IPsec.

Services IPsec

L'utilitaire de gestion des services (SMF) fournit les services suivants pour IPsec :

- service `svc:/network/ipsec/policy` – gère la stratégie IPsec. Par défaut, ce service est activé. La valeur de la propriété `config_file` détermine l'emplacement du fichier `ipsecinit.conf`. La valeur initiale sur un système exécutant le profil de configuration réseau DefaultFixed est `/etc/inet/ipsecinit.conf`. Sur les systèmes qui ne sont pas en cours d'exécution, la valeur de la propriété ce profil est vide.

- service `svc:/network/ipsec/ipsecalgs` – gère les algorithmes disponibles pour IPsec. Par défaut, ce service est activé.
- service `svc:/network/ipsec/manual-key` – Active la gestion manuelle des clés. Par défaut, ce service est désactivé. La valeur de la propriété `config_file` détermine l'emplacement du fichier de configuration `ipseckeys`. La valeur initiale est `/etc/inet/secret/ipseckeys`.
- service `svc:/network/ipsec/ike` – gère IKE. Par défaut, ce service est désactivé. Pour les propriétés configurables, reportez-vous à “Service d'IKEv2” à la page 230 et “Service d'IKEv1” à la page 234.

Pour une présentation complète de SMF, reportez-vous au Chapitre 1, “ Introduction à l’utilitaire de gestion des services ” du manuel “ Gestion des services système dans Oracle Solaris 11.2 ”. Reportez-vous aussi aux pages de manuel `smf`(5), `svcadm`(1M) et `svccfg`(1M).

Commande `ipsecconf`

Pour configurer la stratégie IPsec d'un hôte, vous devez exécuter la commande `ipsecconf`. A l'exécution de la commande de configuration de la stratégie, le système crée des entrées de stratégie IPsec dans le noyau. Elles lui permettent de vérifier la stratégie appliquée à tous les paquets IP entrants et sortants. Paquets qui ne sont pas mis en tunnel et retransmis ne sont pas soumis aux vérifications de stratégie ajoutées à l'aide de cette commande. La commande `ipsecconf` gère aussi les entrées IPsec dans la base de données des stratégies de sécurité. Pour consulter les options de stratégie IPsec, reportez-vous à la page de manuel `ipsecconf`(1M).

Vous devez prendre le rôle `root` pour exécuter la commande `ipsecconf`. La commande peut configurer des entrées qui protègent le trafic bidirectionnel. Elle configure également celles qui protègent le trafic unidirectionnel.

Les entrées de stratégie au format d'adresse locale et d'adresse distante peuvent protéger le trafic dans les deux directions à l'aide d'une entrée de stratégie unique. Par exemple, les entrées de modèles `laddr host1` et `raddr host2` protègent le trafic dans les deux directions quand aucune direction n'est spécifiée pour l'hôte nommé. Par conséquent, une seule entrée de stratégie est nécessaire pour chaque hôte.

Les entrées de stratégie ajoutées par le biais de la commande `ipsecconf` ne sont pas conservées après la réinitialisation du système. Pour vous assurer que la stratégie IPsec est active lorsque le système s'initialise, ajoutez l'entrée de stratégie au fichier `/etc/inet/ipsecinit.conf` , puis actualisez ou activez le service `policy`. Pour obtenir des exemples, reportez-vous à la section “Protection du trafic réseau à l'aide d'IPsec” à la page 109.

Fichier de configuration `ipsecinit.conf`

Pour activer la stratégie de sécurité IPsec lorsque vous démarrez Oracle Solaris, vous créez un fichier de configuration pour initialiser IPsec avec vos entrées de stratégie IPsec spécifiques. Le nom par défaut de ce fichier est `/etc/inet/ipsecinit.conf`. Reportez-vous à la page de manuel `ipsecconf`(1M) pour plus d'informations sur les entrées d'une stratégie et leur format. Une fois la stratégie configurée, vous pouvez l'actualiser avec la commande `svcadm refresh ipsec/policy`.

Exemple de fichier `ipsecinit.conf`

Le logiciel Oracle Solaris comprend un exemple de fichier de stratégie IPsec, `ipsecinit.sample`. Vous pouvez l'utiliser comme modèle pour créer votre propre fichier `ipsecinit.conf`. Le fichier `ipsecinit.sample` contient les exemples suivants :

```
...
# In the following simple example, outbound network traffic between the local
# host and a remote host will be encrypted. Inbound network traffic between
# these addresses is required to be encrypted as well.
#
# This example assumes that 10.0.0.1 is the IPv4 address of this host (laddr)
# and 10.0.0.2 is the IPv4 address of the remote host (raddr).
#

{laddr 10.0.0.1 raddr 10.0.0.2} ipsec
 {encr_algs aes encr_auth_algs sha256 sa shared}

# The policy syntax supports IPv4 and IPv6 addresses as well as symbolic names.
# Refer to the ipsecconf(1M) man page for warnings on using symbolic names and
# many more examples, configuration options and supported algorithms.
#
# This example assumes that 10.0.0.1 is the IPv4 address of this host (laddr)
# and 10.0.0.2 is the IPv4 address of the remote host (raddr).
#
# The remote host will also need an IPsec (and IKE) configuration that mirrors
# this one.
#
# The following line will allow ssh(1) traffic to pass without IPsec protection:

{lport 22 dir both} bypass {}

#
# {laddr 10.0.0.1 dir in} drop {}
#
# Uncommenting the above line will drop all network traffic to this host unless
# it matches the rules above. Leaving this rule commented out will allow
# network packets that do not match the above rules to pass up the IP
# network stack. ,,,
```

Considérations de sécurité à propos de `ipsecinit.conf` et `ipsecconf`

La stratégie IPsec ne peut pas être modifiée pour les connexions établies. Un socket dont la stratégie ne peut pas être modifiée est appelé un *socket verrouillé*. Les nouvelles entrées de stratégie ne protègent pas les sockets qui sont déjà verrouillés. Pour plus d'informations, reportez-vous aux pages de manuel `connect`(3SOCKET) et `accept`(3SOCKET) man pages. En cas de doute, redémarrez la connexion. Pour plus d'informations, reportez-vous à la section `SECURITY` de la page de manuel `ipsecconf`(1M).

Commande `ipsecalgs`

La Structure cryptographique fournit les algorithmes d'authentification et de chiffrement à IPsec. La commande `ipsecalgs` permet d'établir la liste des algorithmes pris en charge par chacun des protocoles IPsec. La configuration `ipsecalgs` est stockée dans le fichier `/etc/inet/ipsecalgs`. En général, ce fichier n'a pas besoin d'être modifié et ne doit jamais être modifié directement. Toutefois, si vous devez modifier le fichier, utilisez la commande `ipsecalgs`. Les algorithmes pris en charge sont synchronisés avec le noyau à l'initialisation du système par le service `svc:/network/ipsec/ipsecalgs:default`.

Les protocoles et algorithmes IPsec valides sont décrits par le DOI, ISAKMP, traité dans le document RFC 2407. En particulier, le DOI ISAKMP définit les conventions d'attribution de nom et de numéro des algorithmes IPsec valides et de leurs protocoles, `PROTO_IPSEC_AH` et `PROTO_IPSEC_ESP`. Chaque algorithme est associé à exactement un protocole. Ces définitions DOI ISAKMP figurent dans le fichier `/etc/inet/ipsecalgs`. Les numéros d'algorithme et de protocole sont définis par l'IANA (Internet Assigned Numbers Authority). La commande `ipsecalgs` permet d'allonger la liste des algorithmes IPsec.

Pour plus d'informations sur les algorithmes, reportez-vous à la page de manuel `ipsecalgs`(1M). Pour plus d'informations sur la Structure cryptographique, reportez-vous au Chapitre 1, “ Structure cryptographique ” du manuel “ Gestion du chiffrement et des certificats dans Oracle Solaris 11.2 ”.

Commande `ipseckey`

La commande `ipseckey` avec ses nombreuses options permet de gérer manuellement les clés pour IPsec. La commande `ipseckey` est décrite à la page de manuel `ipseckey`(1M).

Considérations de sécurité pour la commande `ipseckey`

La commande `ipseckey` permet à un rôle auquel a été attribué le profil de droits Network Security ou Network IPsec Management de saisir des informations de clés cryptographiques confidentielles. Un utilisateur malintentionné accédant à ces informations peut compromettre la sécurité du trafic IPsec.

Remarque - Utilisez plutôt que de génération manuelle de clés IKE, dans la mesure du possible.

Pour plus d'informations, reportez-vous à la section `SECURITY` de la page de manuel `ipseckey`(1M).

Commande `kstat`

La commande `kstat` peut afficher des statistiques sur ESP, AH et d'autres données liées à IPsec. Les options liées à IPsec sont répertoriées sous “Dépannage des erreurs sémantiques dans IPsec et IKE” à la page 212. Reportez-vous également à la page de manuel `kstat`(1M).

IPsec et commande `snoop`

La commande `snoop` peut analyser les en-têtes AH et ESP. En raison du chiffrement des données ESP, la commande `snoop` ne détecte pas les en-têtes chiffrés et protégés par ESP. AH ne chiffre pas les données, de manière que le trafic protégé par AH peut être inspecté à l'aide de la commande `snoop`. L'option `-V` de la commande signale l'utilisation d'AH sur un paquet. Pour plus de détails, reportez-vous à la page de manuel `snoop`(1M).

La section `How to Verify That Packets Are Protected With IPsec` contient un exemple détaillé de sortie “Vérification de la protection des paquets par IPsec” à la page 131 sur un paquet protégé.

Des analyseurs de réseau tiers sont également disponibles, notamment le logiciel open-source Wireshark (http://www.wireshark.org/about.html) qui est intégré à cette version.

RFC IPsec

Le groupe IETF (Internet Engineering Task Force) a publié un certain nombre de documents RFC (Request for Comments, demande de commentaires) décrivant l'architecture de sécurité de la couche IP. Pour plus d'informations sur les RFC, reportez-vous au site Web http://www.ietf.org/. Les références de sécurité IP les plus générales sont couvertes par les RFC suivants :

- RFC 2411, "IP Security Document Roadmap", novembre 1998
- RFC 2401, "Security Architecture for the Internet Protocol", novembre 1998
- RFC 2402, "IP Authentication Header", novembre 1998
- RFC 2406, "IP Encapsulating Security Payload (ESP)", novembre 1998
- RFC 2408, "Internet Security Association and Key Management Protocol (ISAKMP)", novembre 1998 ;
- RFC 2407, "The Internet IP Security Domain of Interpretation for ISAKMP", novembre 1998 ;
- RFC 2409, "The Internet Key Exchange (IKEv1)", novembre 1998
- RFC 5996, "Internet Key Exchange Protocol Version 2 (IKEv2)", septembre 2010
- RFC 3554, "On the Use of Stream Control Transmission Protocol (SCTP) with IPsec", juillet 2003

Base de données des associations de sécurité IPsec

Les informations sur les numéros de clés des services de sécurité IPsec sont conservées dans la base de données des associations de sécurité (SADB). Les associations de sécurité (SA) protègent les paquets entrants et sortants.

Le démon `in.iked` et la commande `ipseckey` utilisent l'interface socket `PF_KEY` dans le cadre de la gestion des SADB. Pour plus d'informations sur la gestion des demandes et des messages par les SADB, reportez-vous à la page de manuel `pf_key`(7P).

Gestion des clés dans IPsec

Le protocole IKE (Internet Key Exchange, échange de clé Internet) gère automatiquement les clés pour IPsec. Les AS IPsec peuvent également être gérées manuellement avec la commande `ipseckey`, mais IKE est recommandé. Pour plus d'informations, reportez-vous à la section “Clé de gestion pour les associations de sécurité IPsec” à la page 96.

La fonctionnalité d'utilitaire de gestion des services (SMF) d'Oracle Solaris fournit les services de gestion des clés suivants pour IPsec :

- service `svc:/network/ipsec/ike` – le service SMF pour la gestion automatique des clés. Le service `ike` a deux instances. L'instance `ike:ikev2` exécute le démon `in.ikev2d` (IKEv2) pour assurer la gestion automatique des clés. Le service `ike:default` exécute le démon `in.iked` (IKEv1). Pour une description du protocole IKE, reportez-vous au Chapitre 8, A propos d'Internet Key Exchange. Pour plus d'informations sur les démons, reportez-vous aux pages de manuel `in.ikev2d`(1M) et `in.iked`(1M)

- service `svc:/network/ipsec/manual-key:default` – Le service SMF pour la gestion manuelle des clés. Le service `manual-key` exécute la commande `ipseckey` avec de nombreuses options pour gérer les clés manuellement. La commande `ipseckey` est décrite à la page de manuel `ipseckey`(1M).

Référence d'IKEv2

IKEv2 remplace IKEv1. Pour une comparaison, reportez-vous à “Comparaison d'IKEv2 et IKEv1” à la page 138.

Utilitaires et fichiers d'IKEv2

Vous trouverez, dans le tableau ci-dessous une liste des fichiers de configuration de la stratégie IKEv2, les emplacements de stockage des clés IKEv2 et les différentes commandes et divers services permettant d'implémenter IKEv2. Pour plus d'informations sur les services, reportez-vous au Chapitre 1, “ Introduction à l’utilitaire de gestion des services ” du manuel “ Gestion des services système dans Oracle Solaris 11.2 ”.

TABLEAU 12-1 Nom de service, commandes, configuration et emplacements de stockage des clés d'IKEv2 et périphériques matériels

Fichier, emplacement, commande ou service	Description	Page de manuel
`svc:/network/ipsec/ike:ikev2`	Le service SMF qui gère IKEv2.	`smf`(5)
`/usr/lib/inet/in.ikev2d`	Démon IKE (Internet Key Exchange). Permet la gestion des clés automatique lorsque le service `ike:ikev2` est actif.	`in.ikev2d`(1M)
`/usr/sbin/ikeadm [-v 2]`	Commande d'administration IKE permettant d'afficher et de modifier temporairement la stratégie IKEv2. Vous permet d'afficher les objets administratifs IKEv2, tels que les groupes Diffie-Hellman disponibles.	`ikeadm`(1M)
`/usr/sbin/ikev2cert`	Commande de gestion de la base de données de certificats pour créer et enregistrer les certificats de clé publique en tant que propriétaire de la configuration, `ikeuser`. Appelle la commande `pktool`.	`ikev2cert`(1M) `pktool`(1)
`/etc/inet/ike/ikev2.config`	Fichier de configuration par défaut pour la stratégie IKEv2. Contient les règles du site pour la concordance des demandes IKEv2 entrantes et la préparation des demandes IKEv2 sortantes. Si le fichier existe, le démon `in.ikev2d` démarre lorsque le service `ike:ikev2` est activé. Vous pouvez changer l'emplacement de ce fichier à l'aide de la commande `svccfg`.	`ikev2.config`(4)
`/etc/inet/ike/ikev2.preshared`	Contient les clés secrètes que les deux instances d'IKEv2 qui n'utilisent pas l'authentification par certificats peuvent utiliser pour s'authentifier mutuellement.	`ikev2.preshared`(4)

Fichier, emplacement, commande ou service	Description	Page de manuel
Keystore softtoken	Contient les clés privées et les certificats de clés publiques pour IKEv2, appartenant à `ikeuser`.	`pkcs11_softtoken`(5)

Service d'IKEv2

L'utilitaire de gestion des services (SMF) fournit le service `svc:/network/ipsec/ike:ikev2` pour gérer IKEv2. Par défaut, ce service est désactivé. Avant d'activer ce service, vous devez créer une configuration IKEv2 valide dans le fichier `/etc/inet/ike/ikev2.config`.

Les propriétés suivantes du service `ike:ikev2` sont configurables :

- `config_file` – Spécifie l'emplacement du fichier de configuration IKEv2. La valeur initiale est `/etc/inet/ike/ikev2.config`. Ce fichier a des autorisations spéciales et doit appartenir à `ikeuser`. N'utilisez pas un autre fichier.
- `debug_level` - Détermine le niveau de débogage du démon `in.ikev2d`. La valeur initiale est `op`, ce qui signifie opérationnelle. Pour connaître les valeurs possibles, reportez-vous au tableau sur les niveaux de débogage sous Types d'objet dans la page de manuel `ikeadm`(1M).
- `debug_logfile` – Spécifie l'emplacement du fichier journal pour le débogage d'IKEv2. La valeur initiale est `/var/log/ikev2/in.ikev2d.log`.
- `kmf_policy` – Détermine l'emplacement du fichier journal pour la stratégie de certificats. La valeur par défaut est `/etc/inet/ike/kmf-policy.xml`. Ce fichier a des autorisations spéciales et doit appartenir à `ikeuser`. N'utilisez pas un autre fichier.
- propriété `pkcs11_token/pin` – Détermine le PIN à utiliser pour la connexion au keystore au démarrage du démon IKEv2. Cette valeur doit correspondre à la valeur que vous avez définie pour le jeton à l'aide de la commande `ikev2cert setpin`.
- propriété `pkcs11_token/uri` – Fixe l'URI de PKCS #11 au keystore. Pour utiliser le composant matériel carte emplacement de stockage dans un accélérateur de cryptographie, vous devez fournir cette valeur.

Pour une présentation complète de SMF, reportez-vous au Chapitre 1, “ Introduction à l’utilitaire de gestion des services ” du manuel “ Gestion des services système dans Oracle Solaris 11.2 ”. Voir aussi les pages de manuel `smf`(5), `svcadm`(1M) et `svccfg`(1M).

Démon IKEv2

Le démon `in.ikev2d` automatise la gestion des clés cryptographiques pour IPsec sur les systèmes Oracle Solaris. Il négocie avec un système distant exécutant le même protocole pour

fournir, de manière protégée, des numéros de clé authentifiés destinés aux associations de sécurité (SA). Le démon doit s'exécuter sur tous les systèmes qui sont censés utiliser IPsec pour protéger les communications à l'aide du protocole IKEv2.

Par défaut, le service `svc:/network/ipsec/ike:ikev2` n'est pas activé. Après que vous avez configuré le fichier `/etc/inet/ike/ikev2.config` et activé l'instance du service `ike:ikev2`, SMF démarre le démon `in.ikev2d` lors de l'initialisation du système.

Lorsque le démon IKEv2 est en cours d'exécution, le système s'authentifie auprès de son entité homologue IKEv2 et établit les clés de la session. A un intervalle spécifié dans le fichier de configuration sont remplacés, les clés IKE automatiquement. Le démon `in.ikev2d` est à l'écoute des demandes IKE entrantes émanant du réseau et des demandes de trafic hors bande via le socket `PF_KEY`. Pour plus d'informations, reportez-vous à la page de manuel `pf_key`(7P).

Deux commandes prennent en charge le démon IKEv2. La commande `ikeadm` peut être utilisée pour afficher la stratégie IKE. Pour plus d'informations, reportez-vous à la section “Commande `ikeadm` pour IKEv2” à la page 232. La commande `ikev2cert` permet de visualiser et de gérer les certificats de clés publiques et privées. Pour plus d'informations, reportez-vous à la section “Commande IKEv2 `ikev2cert`” à la page 232.

Fichier de configuration IKEv2

Le fichier de configuratoin IKEv2, `/etc/inet/ike/ikev2.config`, gère les règles utilisées pour négocier les clés pour les extrémités réseau spécifiées protégées dans le fichier de stratégie IPsec, `/etc/inet/ipsecinit.conf`.

La gestion des clés avec IKE inclut des règles et des paramètres globaux. Les règles IKE identifient les systèmes ou réseaux sécurisés par les numéros de clé. Elles spécifient également la méthode d'authentification. Les paramètres globaux incluent des éléments tels que le délai par défaut avant qu'une AS IKEv2 fasse l'objet d'un renouvellement de clés, `ikesa_lifetime_secs` Pour obtenir des exemples de fichiers de configuration IKEv2, reportez-vous à “Configuration d'IKEv2 avec des clés prépartagées” à la page 144. Pour des exemples et une description des entrées de stratégies IKEv2, consultez la page de manuel `ikev2.config`(4).

Les AS IPsec prises en charge par IKEv2 protègent les paquets IP conformément aux stratégies du fichier de configuration IPsec, `/etc/inet/ipsecinit.conf`.

Les considérations de sécurité pour le fichier `ike/ikev2.config` sont similaires à celles du fichier `ipsecinit.conf`. Pour plus d'informations, reportez-vous à la section “Considérations de sécurité à propos de `ipsecinit.conf` et `ipsecconf`” à la page 226.

Commande `ikeadm` pour IKEv2

Lorsque le démon `in.ikev2d` est en cours d'exécution, vous pouvez utiliser la commande `ikeadm [-v2]` pour effectuer les opérations suivantes :

- Afficher les différents aspects de l'état d'IKEv2
- Affiche les objets du démon IKEv2 tels que les règles de stratégie, les clés prépartagées, les groupes Diffie-Hellman disponibles, les algorithmes d'authentification et de chiffrement et les SA IKEv2 actifs existants.

Pour consulter des exemples et une description complète des options de cette commande, reportez-vous à la page de manuel `ikeadm`(1M)

En matière de sécurité, les considérations concernant la commande `ikeadm` sont similaires à celles concernant la commande `ipseckey`. Pour plus d'informations, reportez-vous à la section “Considérations de sécurité pour la commande `ipseckey`” à la page 227.

Fichier de clés prépartagées d'IKEv2

Le fichier `/etc/inet/ike/ikev2.preshared` contient les clés prépartagées utilisées par le service IKEv2. Le fichier appartient à `ikeuser` et il est protégé à `0600`.

Vous devez personnaliser le fichier `ikev2.preshared` par défaut lorsque vous configurez une règle dans le fichier `ike/ikev2.config` qui requiert des clés prépartagées. Etant donné qu'IKEv2 utilise ces clés prépartagées pour authentifier les pairs IKEv2, ce fichier doit être valide avant que le démon `in.ikev2d` commence à lire des règles nécessitant des clés prépartagées.

Commande IKEv2 `ikev2cert`

La commande `ikev2cert` sert à générer, stocker et gérer des certificats et clés publics et privés. Utilisez-la lorsque le fichier `ike/ikev2.config` requiert des certificats de clés publiques. Etant donné qu'IKEv2 utilise ces certificats pour authentifier les pairs IKEv2, ce fichier doit être en place avant que le démon `in.ikev2d` commence à lire des règles nécessitant les certificats.

La commande `ikev2cert` appelle la commande `pktool` en tant que `ikeuser`.

Les commandes `ikev2cert` suivantes gèrent les certificats pour IKEv2. Elles doivent être exécutées par le compte `ikeuser`. Les résultats sont enregistrés dans le keystore softtoken d'PKCS #11

- `ikev2cert setpin` – Génère un code PIN pour l'utilisateur `ikeuser`. Cette PIN est obligatoire lorsque vous utilisez des certificats.
- `ikev2cert gencert` – Génère un certificat autosigné.
- `ikev2cert gencsr` – Génère une demande de signature de certificat (CSR).
- `ikev2cert list` – Répertorie les certificats dans le keystore.
- `ikev2cert export` – Exporte les fichier dans un fichier en vue de l'export.
- `ikev2cert import` – Importe un certificat ou une LCR.

Pour plus d'informations sur la syntaxe des sous-commandes `ikev2cert`, reportez-vous à la page de manuel `pktool`(1). Pour consulter des exemples, reportez-vous à la page de manuel `ikev2cert`(1M). Pour plus d'informations sur le keystore softtoken, reportez-vous à la page de manuel `cryptoadm`(1M).

Référence d'IKEv1

Les sections suivantes fournissent des informations de référence sur IKEv1. IKEv1 est remplacé par IKEv2, qui offre une gestion des clés automatique plus rapide. Pour plus d'informations sur IKEv2, reportez-vous à “Référence d'IKEv2” à la page 229. Pour une comparaison, reportez-vous à “Comparaison d'IKEv2 et IKEv1” à la page 138.

Utilitaires et fichiers d'IKEv1

Vous trouverez, dans le tableau ci-dessous une liste des fichiers de configuration de la stratégie IKEv1, les emplacements de stockage des clés IKEv1 et les différentes commandes et divers services permettant d'implémenter IKEv1. Pour plus d'informations sur les services, reportez-vous au Chapitre 1, “ Introduction à l’utilitaire de gestion des services ” du manuel “ Gestion des services système dans Oracle Solaris 11.2 ”.

TABLEAU 12-2 Nom de service, commandes, configuration et emplacements de stockage des clés d'IKEv1 et périphériques matériels

Fichier, la commande de service, ou l'appareil,	Description	Page de manuel
`svc:/network/ipsec/ike:default`	Le service SMF qui gère IKEv1.	`smf`(5)
`/usr/lib/inet/in.iked`	démon Internet Key Exchange (IKEv1). Permet la gestion des clés automatique lorsque le service `ike` est actif.	`in.iked`(1M)
`/usr/sbin/ikeadm [-v1]`	Commande d'administration IKE permettant d'afficher et de modifier temporairement la stratégie IKE. Permet à l'utilisateur d'afficher les objets administratifs IKE, tels que les algorithmes de phase 1 et les groupes Diffie-Hellman disponibles.	`ikeadm`(1M)

Fichier, la commande de service, ou l'appareil,	Description	Page de manuel
`/usr/sbin/ikecert`	Commande de gestion de la base de données de certificats pour manipuler les bases de données locales contenant les certificats de clé publique. Les bases de données peuvent également être stockées sur le matériel connecté.	`ikecert`(1M)
`/etc/inet/ike/config`	Fichier de configuration par défaut pour la stratégie IKEv1. Contient les règles du site pour la concordance des demandes IKEv1 entrantes et la préparation des demandes IKEv1 sortantes. Si le fichier existe, le démon `in.iked` démarre lorsque le service `ike` est activé. Vous pouvez changer l'emplacement de ce fichier à l'aide de la commande `svccfg`.	`ike.config`(4)
`ike.preshared`	Fichier des clés prépartagées dans le répertoire `/etc/inet/secret`. Contient des clés secrètes destinées à l'authentification pendant la phase 1. Ce fichier s'utilise pour configurer IKEv1 avec des clés prépartagées.	`ike.preshared`(4)
`ike.privatekeys`	Répertoire des clés privées dans le répertoire `/etc/inet/secret`. Contient les clés privées de la biclé.	`ikecert`(1M)
Répertoire `publickeys`	Répertoire dans `/etc/inet/ike` qui contient les clés publiques et les fichiers certificats. Contient la clé publique de la biclé.	`ikecert`(1M)
Répertoire `crls`	Répertoire dans `/etc/inet/ike` qui contient les listes des clés publiques et les certificats révoqués.	`ikecert`(1M)
Carte Sun Crypto Accelerator 6000	Matériel pouvant accélérer les opérations de clés publiques en déchargeant les opérations à partir du système d'exploitation. Les clés publiques, les clés privées et les certificats de clés publiques peuvent également être stockés sur cette carte. La carte Sun Crypto Accelerator 6000 est un périphérique certifié au niveau 3 de FIPS 140-2.	`ikecert`(1M)

Service d'IKEv1

L'utilitaire de gestion des services (SMF) fournit le service `svc:/network/ipsec/ike:default` pour gérer IKEv1. Par défaut, ce service est désactivé. Avant d'activer ce service, vous devez créer un fichier de configuration IKEv1, `/etc/inet/ike/config`.

Les propriétés suivantes du service `ike` sont configurables :

- `config_file` – Spécifie l'emplacement du fichier de configuration IKEv1. La valeur initiale est `/etc/inet/ike/config`.
- `debug_level` – Détermine le niveau de débogage du démon `in.iked`. La valeur initiale est `op`, ce qui signifie opérationnelle. Pour connaître les valeurs possibles, reportez-vous au tableau sur les niveaux de débogage sous Types d'objet dans la page de manuel `ikeadm`(1M).
- `admin_privilege` – Détermine le niveau de privilège du démon `in.iked`. La valeur initiale est `base`. Les autres valeurs sont `modkeys` et `keymat`. Pour plus de détails, reportez-vous à “Commande IKEv1 `ikeadm`” à la page 236.

Pour une présentation complète de SMF, reportez-vous au Chapitre 1, “ Introduction à l’utilitaire de gestion des services ” du manuel “ Gestion des services système dans Oracle Solaris 11.2 ”. Voir aussi les pages de manuel `smf`(5), `svcadm`(1M) et `svccfg`(1M).

Démon IKEv1

Le démon `in.iked` automatise la gestion des AS IPsec, qui comprennent les clés cryptographiques qui protègent les paquets utilisant IPsec. Le démon négocie de façon sécurisée les SA ISAKMP et les SA IPsec avec un système homologue qui exécute le protocole IKEv1.

Par défaut, le service `svc:/network/ipsec/ike:default` n'est pas activé. Après que vous avez configuré le fichier `/etc/inet/ike/config` et activé le service `ike:default`, le démon `in.iked` se lance à l'initialisation du système. Outre le fichier `/etc/inet/ike/config`, d'autres fichiers de configuration sont stockés dans d'autres fichiers et bases de données, ou en tant que propriétés du SMF. Pour plus d'informations, reportez-vous à “Utilitaires et fichiers d'IKEv1” à la page 233 et aux pages de manuel `ike.preshared`(4), `ikecert`(1M) et `in.iked`(1M).

Une fois le service `ike:default` activé, le démon `in.iked` lit les fichiers de configuration et écoute les demandes externes provenant d'un pair IKE et les demandes internes d'AS d'IPsec.

Pour les demandes externes provenant d'un pair IKEv1, la configuration du service `ike:default` détermine la manière dont le démon réagit. Les demandes internes sont acheminées via l'interface `PF_KEY`. Cette interface gère la communication entre la partie noyau d'IPsec, qui stocke les AS IPsec et gère le chiffrement et le déchiffrement de paquets, et le démon de gestion des clés, `in.iked`, qui s'exécute dans l'espace utilisateur. Lorsque le noyau a besoin d'une AS pour protéger un paquet, il envoie un message au démon `in.iked` via l'interface `PF_KEY`. Pour plus d'informations, reportez-vous à la page de manuel `pf_key`(7P).

Deux commandes prennent en charge le démon IKEv1. La commande `ikeadm` fournit une interface de ligne de commande pour le démon en cours d'exécution. La commande `ikecert` gère les bases de données de certificats, `ike.privatekeys` et `publickeys`, sur le disque et le matériel.

Pour plus d'informations sur ces commandes, reportez-vous aux pages de manuel `in.iked`(1M), `ikeadm`(1M) et `ikecert`(1M)

Fichier de configuration IKEv1

Le fichier de configuration IKEv1, `/etc/inet/ike/config`, gère les AS pour les paquets réseaux nécessitant une protection IPsec conformément aux stratégies du fichier de configuration IPsec, `/etc/inet/ipsecinit.conf`.

La gestion des clés avec IKE inclut des règles et des paramètres globaux. Une règle IKEv1 identifie les systèmes qui exécutent un autre démon IKEv1. Elles spécifient également la méthode d'authentification. Les paramètres globaux incluent des éléments tels que le chemin vers un accélérateur matériel connecté. Pour obtenir des exemples de fichiers de stratégie IKEv1, reportez-vous à “Configuration d'IKEv2 avec des clés prépartagées” à la page 144. Pour des exemples et une description des entrées de stratégies IKEv1, consultez la page de manuel `ike.config`(4).

Le fichier `/etc/inet/ike/config` peut inclure le chemin vers une bibliothèque implémentée conformément à la norme suivante : RSA Security Inc. PKCS #11 Cryptographic Token Interface (Cryptoki). IKEv1 utilise cette bibliothèque PKCS #11 pour accéder au matériel d'accélération et de stockage des clés.

Les considérations de sécurité pour le fichier `ike/config` sont similaires à celles du fichier `ipsecinit.conf`. Pour plus d'informations, reportez-vous à la section “Considérations de sécurité à propos de `ipsecinit.conf` et `ipsecconf`” à la page 226.

Commande IKEv1 `ikeadm`

La commande `ikeadm` permet d'effectuer les opérations suivantes :

- Afficher les différents aspects de l'état d'IKE
- modifier les propriétés du démon IKE ;
- Afficher les statistiques concernant la création de SA pendant la phase 1
- Déboguer les échanges du protocole IKE
- Affiche les objets du démon IKE tels que les SA Phase 1, les règles de stratégie, les clés prépartagées, les groupes Diffie-Hellman disponibles, les algorithmes d'authentification et de chiffrement Phase 1, et le cache de certificat

Pour consulter des exemples et une description complète des options de cette commande, reportez-vous à la page de manuel `ikeadm`(1M)

Le niveau de privilège du démon IKE en cours d'exécution détermine les aspects du démon IKE susceptibles d'être affichés et modifiés. Trois niveaux de privilège sont possibles :

niveau `base` — Vous ne pouvez ni afficher ni modifier les clés. Le niveau `base` est le niveau de privilège par défaut.

niveau `keymat` : Ce niveau vous permet d'afficher les clés actuelles à l'aide de la commande `ikeadm`.

niveau `modkeys` : A ce niveau, vous pouvez supprimer, modifier et ajouter des clés prépartagées.

Vous pouvez utiliser la commande `ikeadm` pour modifier temporairement un privilège. Pour une modification permanente, modifiez la propriété `admin_privilege` du service `ike`. Pour la modification temporaire du privilège, reportez-vous à “Gestion des démons IKE en cours d'exécution” à la page 220.

En matière de sécurité, les considérations concernant la commande `ikeadm` sont similaires à celles concernant la commande `ipseckey`. Reportez-vous à “Considérations de sécurité pour la commande `ipseckey`” à la page 227. Pour plus de détails qui sont propres à la commande `ikeadm`, reportez-vous à la page de manuel `ikeadm`(1M).

Fichiers de clés prépartagées IKEv1

Lorsque vous créez manuellement les des clés prépartagées, elles sont enregistrées dans des fichiers dans `/etc/inet/secret`. Le fichier `ike.preshared` contient les clés prépartagées de la phase 1 lorsque vous configurez une règle dans `ike/config` pour utiliser des clés prépartagées. Le fichier `ipseckeys` contient les clés prépartagées utilisées pour protéger les paquets IP. Ces fichiers sont protégés en mode `0600` et le répertoire `secret` en mode `0700`.

Les clés prépartagées étant utilisées pour authentifier la phase 1, le fichier doit être valide avant le démarrage du démon `in.iked`.

Pour consulter des exemples de gestion manuelle des clés IPsec, reportez-vous à “Création manuelle de clés IPsec” à la page 125.

Bases de données de clés publiques et commandes IKEv1

La commande `ikecert` gère les clés publiques et privées du système, ses certificats publics et les bases de données de LCR statiques. Utilisez-la lorsque le fichier de configuration d'IKEv1 requiert des certificats de clés publiques. Ces bases de données étant utilisées par IKEv1 pour authentifier la phase 1 de l'échange, elles doivent être alimentées avant l'activation du démon `in.iked`. Trois sous-commandes permettent de gérer chacune des trois bases de données : `certlocal`, `certdb` et `certrldb`.

Si une carte Sun Crypto Accelerator 6000 est connectée au système, la commande `ikecert` utilise une bibliothèque PKCS #11 pour accéder au stockage matériel de clés et certificats.

Pour plus d'informations, consultez la page de manuel `ikecert`(1M) Pour plus d'informations sur le metaslot et sur le fichier keystore de clés softtoken, reportez-vous à la page de manuel `cryptoadm`(1M).

Commande IKEv1 `ikecert tokens`

L'argument `tokens` répertorie les ID de jetons disponibles. Les ID de jetons permettent aux commandes `ikecert certlocal` et `ikecert certdb` de générer des certificats de clés publiques et des CSR. Les clés et certificats peuvent également être stockées sur une carte Sun Crypto Accelerator 6000 connectée. La commande `ikecert`utilise la bibliothèque PKCS #11 pour accéder au keystore matériel.

Commande IKEv1 `ikecert certlocal`

La sous-commande `certlocal`gère la base de données des clés privées. Les options de cette sous-commande permettent d'ajouter, d'afficher et de supprimer des clés privées. Cette sous-commande permet également de créer un certificat autosigné ou une CSR. L'option `-ks` crée un certificat autosigné L'option `-kc` crée une CSR. Les clés sont stockées sur le système, dans le répertoire `/etc/inet/secret/ike.privatekeys`, ou, avec l'option `-T`, sur un composant matériel connecté.

Lorsque vous créez une clé privée, les options de la commande `ikecert certlocal` doivent avoir des entrées connexes dans le fichier `ike/config`. Le tableau ci-dessous détaille les correspondances entre les options `ikecert` et les entrées `ike/config`.

TABLEAU 12-3 Correspondances entre les options `ikecert` et les entrées `ike/config` dans IKEv1

Option `ikecert`	Entrée `ike/config`	Description
`-A` *subject-alternate-name*	`cert_trust` *subject-alternate-name*	Pseudonyme identifiant le certificat de manière unique. Il peut s'agir d'une adresse IP, d'une adresse e-mail ou d'un nom de domaine.
`-D` *X.509-distinguished-name*	*X.509-distinguished-name*	Nom complet de l'autorité de certification, incluant le pays (C), le nom de l'organisation (ON), l'unité d'organisation (OU) et le nom commun (CN).
`-t dsa-sha1 | dsa-sha256`	`auth_method dsa_sig`	Méthode d'authentification légèrement plus lente que RSA.
`-t rsa-md5` et `-t rsa-sha1 | rsa-sha256 | rsa-sha384 | rsa-sha512`	`auth_method rsa_sig`	Méthode d'authentification légèrement plus rapide que la méthode DSA. La clé publique RSA doit être suffisamment importante pour chiffrer la charge utile la plus lourde. Les charges les plus lourdes sont habituellement les données d'identité (par exemple, le nom distinctif X.509).
`-t rsa-md5` et	`auth_method rsa_encrypt`	Le chiffrement RSA met les identités d'IKE à l'abri des écoutes électroniques, mais implique que les pairs IKE connaissent leurs clés publiques respectives.

Option `ikecert`	Entrée `ike/config`	Description
`-t rsa-sha1 | rsa-sha256 | rsa-sha384 | rsa-sha512`		

Si vous émettez une CSR à l'aide de la commande `ikecert certlocal -kc`, vous envoyez la sortie de la commande à l'autorité de certification (AC). Si votre entreprise possède sa propre PKI (Public Key Infrastructure), vous envoyez cette sortie à votre administrateur de PKI. L'AC ou votre administrateur de PKI crée ensuite les certificats. Ceux qui vous sont remis sont entrés dans la sous-commande `certdb`. La liste des certificats révoqués (LCR) que l'AC vous envoie est entrée dans la sous-commande `certrldb`.

Commande IKEv1 `ikecert certdb`

la sous-commande `certdb` gère la base de données des clés publiques. Les options de cette sous-commande vous permettent d'ajouter, d'afficher et de supprimer des certificats et des clés publiques. Cette sous-commande accepte l'entrée de certificats générés par la commande `ikecert certlocal -ks` sur un système distant. Pour plus d'informations sur cette procédure, reportez-vous à la section “Configuration d'IKEv1 avec des certificats de clés publiques autosignés” à la page 178. Cette commande accepte également l'entrée de certificats émanant d'une AC. Pour plus d'informations sur cette procédure, reportez-vous à la section “Configuration du protocole IKEv1 avec des certificats signés par une AC” à la page 183.

Les certificats et les clés publiques sont stockés sur le système dans le répertoire `/etc/inet/ike/publickeys`. L'option `-T` permet de stocker les certificats, les clés privées et les clés publiques sur les composants matériels connectés.

Commande IKEv1 `ikecert certrldb`

La sous-commande `certrldb` gère la base de données des listes de certificats révoqués (LCR), `/etc/inet/ike/crls`. Cette base de données LCR met à jour les listes de révocation des clés publiques. Les certificats qui ne sont plus valides figurent dans ces listes. Lorsqu'une AC vous fait parvenir une LCR, vous pouvez l'installer dans cette base de données à l'aide de la commande `ikecert certrldb`. Pour plus d'informations sur cette procédure, reportez-vous à la section “Gestion des certificats révoqués dans IKEv1” à la page 192.

Répertoire IKEv1 `/etc/inet/ike/publickeys`

Le répertoire `/etc/inet/ike/publickeys` contient la partie publique d'une paire de clés publique-privée et son certificat, dans des fichier aussi appelés *emplacements*. Ce répertoire est protégé en mode `0755`. et peut être alimenté à l'aide de la commande `ikecert certdb`.

L'option `-T` permet de stocker les clés sur une carte Sun Crypto Accelerator 6000 plutôt que dans le répertoire `publickeys`.

les emplacements contiennent, sous forme codée, le nom distinctif X.509 d'un certificat généré sur un autre système. Si vous utilisez des certificats autosignés, vous devez indiquer le certificat que l'administrateur du système distant vous a envoyé comme entrée de commande. Si vous utilisez des certificats d'une CA, vous installez deux certificats signés d'une CA dans la base de données. Vous installez un certificat basé sur la CSR envoyée à l'AC. Vous installez également un certificat de la CA.

IKEv1 `/etc/inet/secret/ike.privatekeys`

Le répertoire `/etc/inet/secret/ike.privatekeys` contient des fichiers de clés privées qui font partie d'une paire de clés publique-privée. Ce répertoire est protégé en mode `0700`. La commande `ikecert certlocal` permet d'alimenter le répertoire `ike.privatekeys`. Les clés privées sont effectives uniquement lors de l'installation de leur clé publique homologue, de certificats autosignés ou de certificats CA. Les clés publiques homologues sont stockées dans le répertoire `/etc/inet/ike/publickeys` ou sur du matériel pris en charge.

Répertoire IKEv1 `/etc/inet/ike/crls`

Le répertoire `/etc/inet/ike/crls` contient les fichiers des listes de certificats révoqués (LCR). Chaque fichier correspond à un fichier de certificat public du répertoire `/etc/inet/ike/publickeys`. Les AC fournissent les LCR correspondant à leurs certificats. La commande `ikecert certrldb` permet d'alimenter la base de données.

Glossaire de la sécurité réseau

3DES Voir Triple-DES.

adresse de diffusion Adresses réseau IPv4 avec la partie hôte de l'adresse ne comportant que des zéros (10.50.0.0) ou des valeurs à un bit (10.50.255.255). Un paquet envoyé à une adresse de diffusion à partir d'un ordinateur situé sur le réseau local est transmis à tous les ordinateurs reliés au réseau.

adresse de multidiffusion Adresse IPv6 identifiant un groupe d'interfaces d'une manière particulière. Un paquet envoyé à une adresse de multidiffusion est diffusé à toutes les interfaces du groupe. La fonctionnalité de l'adresse de multidiffusion IPv6 est similaire à celle de l'adresse de diffusion IPv4.

adresse lien-local Dans IPv6, désignation utilisée en guise d'adresse d'une liaison simple lors d'une configuration d'adresse automatique, par exemple. Par défaut, l'adresse lien-local est créée à partir de l'adresse MAC du système.

AES Standard de chiffrement avancé (Advanced Encryption Standard). Technique de chiffrement de données symétrique par blocs. Le gouvernement des Etats-Unis a adopté la variante Rijndael de l'algorithme comme norme de chiffrement en octobre 2000. AES remplace le chiffrement Standard de chiffrement de données (DES) comme norme administrative.

algorithme Diffie-Hellman Egalement appelé cryptographie par clé publique. Un protocole d'accord de clés cryptographiques asymétriques a été mis au point par Diffie et Hellman en 1976. Ce protocole permet à deux utilisateurs d'échanger une clé secrète via un moyen non sécurisé sans secrets préalables. Diffie-Hellman est utilisé par le protocole IKE.

ancre sécurisée Dans de certificats X.509, le certificat root de l'autorité de certification. Les certificats du certificat root au certificat final établissent une chaîne de confiance.

association de sécurité (AS) SA, Security Association. Association définissant les propriétés en matière de sécurité entre un premier hôte et un deuxième hôte.

attaque par réflexion Attaque consistant à envoyer à distance des paquets de demande d'écho ICMP à une adresse de diffusion IP ou à plusieurs adresses de diffusion dans le but de congestionner le réseau ou de provoquer de graves interruptions de service.

attaque par rejeu Dans IPsec, attaque impliquant la capture d'un paquet par un intrus. Le paquet stocké remplace ou réplique l'original par la suite. Pour se protéger contre ce type d'attaque, il suffit que le paquet contienne un champ qui s'incrémente pendant la durée de vie de la clé secrète assurant la sécurité du paquet.

autorité de certification (AC) Organisation ou société « tiers de confiance » publiant des certificats numériques utilisés pour créer des signatures numériques et des biclés. L'Autorité de certification (CA) garantit l'identité d'une personne ayant reçu un certificat unique.

base de données des stratégies de sécurité SPD, Security Policy Database. Base de données définissant le niveau de protection à appliquer à un paquet. La base de données SPD filtre le trafic IP afin de déterminer s'il est nécessaire de rejeter un paquet, de le transmettre en clair ou de le protéger avec IPsec.

Blowfish Algorithme de chiffrement par bloc symétrique de longueur de clé variable (entre 32 et 448 bits). Son créateur, Bruce Schneier, affirme que Blowfish est optimisé pour les applications pour lesquelles la clé n'a pas besoin d'être régulièrement modifiée.

chaîne de confiance Dans les certificats X.509, la garantie par l'autorité de certification que depuis l'ancre sécurisée au certificat de l'utilisateur, tous les certificats présentent une chaîne d'authentification ininterrompue.

charge utile Données transportées dans un paquet. La charge utile n'inclut pas les informations d'en-tête nécessaires pour amener le paquet à destination.

couche liaison Couche située immédiatement sous IPv4/IPv6.

cryptographie asymétrique Système de chiffrement dans lequel l'expéditeur et le destinataire du message utilisent différentes clés pour chiffrer et déchiffrer le message. Les clés asymétriques servent à établir un canal sécurisé pour le chiffrement par clé symétrique. L'algorithme Diffie-Hellman est un exemple de protocole de clé asymétrique. Voir aussi cryptographie symétrique.

cryptographie par clé publique Système cryptographique utilisant deux clés différentes : une clé publique connue de tous et une clé privée présentée exclusivement au destinataire du message. IKE fournit des clés publiques à IPsec.

cryptographie symétrique Système de chiffrement dans lequel l'expéditeur et le destinataire d'un message partagent une clé unique commune. Cette clé commune sert au chiffrement et au déchiffrement du message. Les clés symétriques permettent de chiffrer l'ensemble de la transmission de données dans IPsec. AES est un exemple de cryptographie par clé symétrique.

détection de routeur Processus selon lequel les hôtes localisent des routeurs résidant sur une liaison directe.

DOI Un DOI (Domain of Interpretation, domaine d'interprétation) définit les formats de données, les types d'échange du trafic réseau ainsi que les conventions d'appellation des informations liées à la sécurité. Les stratégies de sécurité, les algorithmes et les modes cryptographiques sont toutes des informations ayant trait à la sécurité.

DSA algorithme de signature numérique (Digital Signature Algorithm) Algorithme de clé publique dont la longueur de clé varie de 512 à 4 096 bits. La norme du gouvernement américain, DSS, atteint 1 024 bits. L'algorithme DSA repose sur l'algorithme SHA-1 en entrée.

ECDSA Algorithme de signature numérique de courbe elliptique. Algorithme de clé publique basé sur une courbe elliptique mathématique. La taille d'une clé ECDSA est de beaucoup plus petite que celle d'une clé publique DSA nécessaire pour générer une signature de la même longueur.

en-tête d'authentification En-tête d'extension assurant l'authentification et l'intégrité des paquets IP, mais pas leur confidentialité.

en-tête de paquet Voir en-tête IP.

en-tête IP Vingt octets de données identifiant de manière unique un paquet Internet. L'en-tête inclut l'adresse source et l'adresse de destination du paquet. Une partie facultative de l'en-tête permet d'insérer des octets supplémentaires.

encapsulation Processus selon lequel un en-tête et une charge utile sont placés dans le premier paquet, puis insérés dans la charge utile du deuxième paquet.

encapsulation IP-in-IP Mécanisme de mise en tunnel des paquets IP au sein de paquets IP.

étiquette 1. Un mot-clé d'une règle IKEv2 dont la valeur doit correspondre à la valeur du mot-clé `label` dans un fichier de clés prépartagées si `auth_method` est `preshared`.

2. Mot-clé utilisé lors de la création d'un certificat IKEv2. Cette valeur est pratique permettant de localiser toutes les parties du certificat (clé privée, la clé publique et dans le certificat de clé publique keystore).

(3. MAC A d'accès obligatoire) bon indicateur de contrôle le niveau de confidentialité d'un objet ou d'un business process. Confidentiel et Top Secret sont des exemples d'étiquettes. Les transmissions réseau étiquetées contiennent des étiquettes MAC.

4. Mot-clé d'une règle IKEv1 dont la valeur est utilisée pour obtenir la règle.

filtrage de paquets Fonction de pare-feu pouvant être configurée pour autoriser ou interdire le transit de paquets particuliers via un pare-feu.

filtre de paquets dynamique Voir filtre de paquets sans état.

filtre de paquets sans état filtrage de paquets permettant de contrôler l'état des connexions actives et d'identifier, à l'aide des informations obtenues, les paquets du réseau autorisés à franchir le pare-feu. En assurant le suivi et la coordination des requêtes et des réponses, un filtre de paquets sans état a la possibilité d'écarter une réponse non satisfaisante.

gestion des clés La façon dont vous gérez les associations de sécurité.

HMAC Méthode de hachage à clé pour l'authentification de messages. HMAC est un algorithme d'authentification à clé secrète. HMAC est utilisé avec une fonction de repère cryptographique

répétitive, telle que MD5 ou SHA-1, combinée avec une clé secrète partagée. La puissance cryptographique de HMAC dépend des propriétés de la fonction de repère sous-jacente.

hôte à multihébergement Système doté de plusieurs interfaces physiques et qui ne transfère pas les paquets. Un hôte à multihébergement peut exécuter des protocoles de routage.

IKE Internet Key Exchange, échange de clé Internet. IKE automatise la mise en service de matériel d'identification authentifié pour les associations de sécurité IPsec.

index du paramètre de sécurité SPI, Security Parameter Index. Nombre entier indiquant la rangée de la SADB qui permettra au récepteur de décrypter un paquet reçu.

interface physique Mode de raccordement d'un système à une liaison. Ce mode de raccordement est souvent mis en oeuvre sous la forme d'un pilote de périphérique et d'une NIC. Certaines cartes d'interface réseau (`igb`, par exemple) peuvent disposer de plusieurs points de connexion.

interface réseau virtuelle (VNIC) Pseudo-interface offrant une connectivité réseau virtuelle qu'elle soit ou non configurée sur une interface réseau physique. Des conteneurs, tels que des zones IP exclusives, sont configurés sur des VNIC afin de former un réseau virtuel.

Internet Protocol (IP, protocole Internet) Méthode ou protocole utilisé pour envoyer les données d'un ordinateur à l'autre via Internet.

IP Voir Internet Protocol (IP, protocole Internet), IPv4, IPv6.

IPsec Sécurité IP. Architecture de sécurité assurant la protection des paquets IP.

IPv4 Protocole Internet, version 4. IPv4 est parfois appelé IP. Cette version prend en charge un espace d'adressage à 32 bits.

IPv6 Protocole Internet, version 6. IPv6 prend en charge un espace d'adressage à 128 bits.

liaison IP Utilitaire ou moyen de communication à l'aide duquel les noeuds peuvent communiquer dans la couche liaison. La couche liaison se trouve immédiatement sous IPv4/IPv6. Les réseaux Ethernet (simple ou reliés par un pont) ou les réseaux ATM sont des exemples de liaisons IP. Une liaison IP est définie par un ou plusieurs numéros ou préfixes de masque de sous-réseau IPv4. Un même numéro ou préfixe de masque de sous-réseau ne peut pas être attribué à plusieurs liaisons IP. Dans le système ATM LANE, une liaison IP est un LAN à émulation simple. Lorsque vous utilisez le système ARP, la portée du protocole ARP correspond à une liaison IP simple.

liste des certificats révoqués (CRL) Liste des certificats de clés publiques ayant fait l'objet d'une révocation par une CA. Les listes de certificats révoqués sont stockées dans la base de données des listes de certificats révoqués, gérée par IKE.

MAC MAC garantit l'intégrité des données et authentifie leur origine. MAC ne protège aucunement contre l'écoute frauduleuse des informations échangées.

marqueur 1. Module de l'architecture diffserv et IPQoS attribuant une valeur au champ DS d'un paquet IP. Cette valeur indique la manière dont est traité le paquet. Dans l'implémentation IPQoS, le module marqueur est `dscpmk`.

2. Module dans l'implémentation IPQoS qui marque l'indicateur de réseau local virtuel d'un paquet Ethernet par une valeur de priorité utilisateur. La valeur de priorité utilisateur indique comment les paquets sont transmis sur un réseau comportant des périphériques VLAN. Ce module est appelé `dlcosmk`.

MD5 Fonction de hachage cryptographique répétitive utilisée pour authentifier les messages, y compris les signatures numériques. Elle a été développée en 1991 par Rivest.

NAT Voir Traduction d'adresse réseau (NAT, Network Address Translation).

NIC Network Interface Card, carte d'interface réseau. Carte réseau jouant le rôle d'interface d'un réseau. Certaines NIC ont plusieurs interfaces physiques. C'est le cas des cartes `igb`.

nom distinctif (ND) Méthode normalisée pour représenter à l'aide de chaînes d'informations partagées. LDAP des noms distinctifs (DN) et dans sont utilisées dans les certificats X.509, ainsi que dans d'autres technologies. Pour plus d'informations, reportez-vous à A String Representation of Distinguished Names (http://www.ietf.org/rfc/rfc1779.txt)

nom du keystore Nom qu'un administrateur attribue à une zone de stockage, ou keystore, sur une NIC. Le nom du keystore est également appelé jeton ou ID de jeton.

paquet Voir paquet IP.

paquet Groupe d'informations transmis sous forme d'une unité sur les lignes de communications. Un paquet contient un en-tête IP et une charge utile.

paquet de demande d'écho ICMP Paquet transmis à une machine sur Internet en vue de solliciter une réponse. De tels paquets sont communément appelés paquets "ping".

paquet IP Paquet d'informations transporté par IP. Un paquet IP contient un en-tête et des données. L'en-tête inclut les adresses de la source et de la destination du paquet. Les autres champs de l'en-tête permettent d'identifier et de recombiner les données avec les paquets associés lorsqu'ils arrivent à destination.

pare-feu Dispositif ou logiciel prévu pour isoler le réseau privé ou le réseau intranet d'une organisation d'Internet, afin de le protéger contre d'éventuelles intrusions. Un pare-feu peut inclure le filtrage de paquets, des serveurs proxy et les valeurs NAT (Network Address Translation, translation d'adresse réseau).

périphérique VLAN Interface réseau permettant de renvoyer le trafic vers le niveau Ethernet (liaison de données) de la pile de protocole IP.

PFS Perfect Forward Secrecy, secret rigoureux des transmission .Avec la fonction PFS, la clé visant à protéger la transmission des données n'est pas utilisée pour dériver d'autres clés. Il en est de même pour la source de la clé. Par conséquent, PFS peut empêcher le déchiffrement du trafic précédemment enregistré.

PFS s'applique à l'échange de clés authentifiées uniquement. Voir aussi algorithme Diffie-Hellman.

pile IP TCP/IP est souvent appelé une "pile". Ce terme fait référence aux couches (TCP, IP et parfois d'autres) par lesquelles transitent toutes les données aux extrémités client et serveur d'un échange de données.

PKI Infrastructure à clé publique. Système de certificats numériques, d'autorités de certification et d'autres autorités d'enregistrement prévu pour vérifier et authentifier la validité de chaque partie impliquée dans une transaction Internet.

protocole ESP Extension de l'en-tête assurant l'intégrité et la confidentialité des paquets. ESP est l'un des cinq composants de l'architecture de sécurité IP (IPsec).

publication du routeur Processus selon lequel les routeurs annoncent leur présence (avec divers paramètres de connexion et paramètres Internet) de façon périodique ou en réponse à un message de sollicitation d'un routeur.

reniflage Action d'espionner les communications des réseaux informatiques. Cette technique est fréquemment employée avec des programmes automatisés pour extirper hors ligne des informations telles que des mots de passe en clair.

réseau virtuel Regroupement de ressources et fonctionnalités réseau logicielles et matérielles gérées en tant qu'entité logicielle unique. Un réseau virtuel *interne* regroupe les ressources réseau sur un seul système, parfois appelé "réseau en boîte".

routeur Système généralement composé de plusieurs interfaces ayant pour fonction d'exécuter des protocoles de routage et de transférer des paquets. Vous pouvez configurer un système à une seule interface en guise de routeur à condition que le système se trouve à l'extrémité d'une liaison PPP.

RSA Méthode permettant d'obtenir des signatures numériques et des systèmes de cryptographie par clé publique. Cette méthode datant de 1978 a été décrite par trois développeurs (Rivest, Shamir et Adleman).

SADB Security Associations Database, base de données des associations de sécurité. Table définissant les clés cryptographiques et les algorithmes cryptographiques. Les clés et les algorithmes ont pour intérêt de sécuriser la transmission des données.

serveur proxy Serveur faisant l'interface entre une application client (telle qu'un navigateur Web) et un autre serveur. Ce type de serveur permet de filtrer les demandes afin d'interdire l'accès à certains sites Web, par exemple.

SHA-1 Secure Hashing Algorithm, algorithme de hachage sécurisé. L'algorithme s'applique à toute longueur d'entrée inférieure à 2^{64} afin d'obtenir une synthèse des messages. L'algorithme SHA-1 sert d'entrée à l'algorithme DSA.

signature numérique Code numérique associé à un message électronique qui identifie l'expéditeur de manière unique.

sollicitation du routeur Processus selon lequel les hôtes demandent à des routeurs de générer immédiatement des publications du routeur, et non pas lors de la prochaine exécution programmée.

Standard de chiffrement de données (DES) Standard de chiffrement de données (Data Encryption Standard). Méthode de chiffrement à clé symétrique développée en 1975 et standardisée par l'ANSI en 1981 comme ANSI X.3.92. Le DES utilise une clé de 56 bits.

stream control transport protocol (SCTP) SCTP, Stream Control Transport Protocol. Protocole de la couche transport assurant des communications orientées connexion sous une forme similaire au protocole TCP. De plus, SCTP gère le multihébergement (une des extrémités de la connexion peut être associée à plusieurs adresses IP).

TCP/IP Transmission Control Protocol/Internet Protocol, protocole de contrôle de la transmission/ protocole Internet. TCP/IP est le langage de communication ou protocole de base sur Internet. Il peut également servir de protocole de communication sur un réseau privé (intranet ou extranet).

Traduction d'adresse réseau (NAT, Network Address Translation) Traduction d'une adresse IP utilisée au sein d'un réseau sous une adresse IP différente connue au sein d'un autre réseau. Cette technique sert à limiter le nombre d'adresses IP globales nécessaires.

Triple-DES Triple-Data Encryption, triple chiffrement des données. Méthode de chiffrement par clé symétrique. Elle nécessite une clé de 168 bits. L'abréviation de Triple-DES est 3DES.

tunnel Chemin suivi par un paquet pendant son encapsulation. Voir encapsulation.

Dans IPsec, un tunnel configuré est une interface point à point. Le tunnel permet l'encapsulation d'un paquet IP dans un autre paquet IP.

tunnel bidirectionnel Tunnel pouvant transmettre des paquets dans les deux directions.

usurpation Action d'accéder par intrusion à un ordinateur en envoyant un message avec une adresse IP provenant prétendument d'un hôte de confiance. Pour ce faire, un pirate doit d'abord utiliser différentes techniques pour identifier l'adresse IP d'un hôte de confiance, puis modifier les en-têtes de paquets pour donner l'impression que les paquets proviennent de cet hôte.

valeur de hachage Nombre généré à partir d'une chaîne de texte. Les fonctions de hachage garantissent que les messages transmis n'ont pas été sabotés. MD5 et SHA-1 sont des exemples de fonctions de hachage unidirectionnel.

VPN Virtual Private Network, réseau privé virtuel. Réseau logique sécurisé utilisant des tunnels dans un réseau public tel qu'Internet.

Index

B

C

D

E

F

G

H

I

M

N

O

P

R

S

T

U

V

W

X

Z