Configuration et dépannage du chiffrement de couche réseau Cisco IPSec et ISAKMP - Partie 2

Contenu

Introduction

La Première partie de ce rapport technique couvre des informations générales en matière de Cryptage de réseau-couche et la configuration de base du Chiffrement de réseau-couche. Cette partie du document couvre la Sécurité IP (IPSec) et le Internet Security Association and Key Management Protocol (ISAKMP).

IPSec a été introduit dans la version de logiciel 11.3T de Cisco IOS®. Il fournit un mécanisme pour la transmission de données sécurisée et se compose d'ISAKMP/Oakley et d'IPSec.

Conditions préalables

Conditions requises

Aucune spécification déterminée n'est requise pour ce document.

Composants utilisés

Les informations de ce document sont basées sur les versions de logiciel et matériel suivantes :

- Version du logiciel Cisco IOS 11.3(T) et plus tard

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

Conventions

Pour plus d'informations sur les conventions des documents, référez-vous aux Conventions utilisées pour les conseils techniques de Cisco.

L'information générale et configuration de cryptage de couche réseau

Définitions

Cette section définit les termes connexes utilisés dans tout ce document.

- **Authentification :** La propriété de savoir que les données reçues sont envoyées réellement par l'expéditeur réclamé.
- **Confidentialité :** La propriété de la communication de sorte que les destinataires destinés sachent ce qui est envoyé mais les interlocuteurs fortuits ne peut pas déterminer ce qui est envoyé.
- **Norme de chiffrement de données (DES) :** Le DES utilise une méthode principale symétrique, également connue sous le nom de méthode principale secrète. Ceci signifie que si un bloc de données est chiffré avec la clé, le bloc chiffré doit être déchiffré avec la même clé, ainsi l'unité de chiffrement et le decrypter doivent utiliser la même clé. Quoique la méthode de cryptage soit connue et bien éditée, la méthode d'attaque connue de meilleur publiquement est par la force brutale. Des clés doivent être testées contre les blocs chiffrés pour voir si elles peuvent correctement les résoudre. Pendant que les processeurs deviennent plus puissants, la vie naturelle du DES s'approche de son extrémité. Par exemple, un effort coordonné utilisant la capacité de traitement supplémentaire des milliers d'ordinateurs à travers l'Internet peut trouver la clé 56-bit à un message encodé par DES en 21 jours.Le DES est validé tous les cinq ans par l'agence de Sécurité nationale des USA (NSA) pour rencontrer les buts du gouvernement des USA. L'approbation en cours expire en 1998 et le NSA a indiqué qu'ils re-ne certifieront pas le DES. Se déplacent au delà du DES, là d'autres algorithmes de chiffrement qui également n'ont aucune faiblesse connue autres que des attaques de force brutale. Pour information les informations complémentaires, voir les PAP DES 46-2 par le National Institute of Standards and Technology (NIST) .
- **Déchiffrement :** L'application inverse d'un algorithme de chiffrement aux données cryptées, restaurant de ce fait ces données sur son état d'origine et décrypté.
- **Algorithme de DSS et de signature numérique (DSA) :** Le DSA a été édité par le NIST dans le Norme de signature numérique (DSS), qui est une partie du projet de la pierre angulaire du gouvernement des États-Unis. Le DSS a été sélectionné par le NIST, en coopération avec le NSA, pour être le niveau numérique d'authentification du gouvernement des États-Unis. La norme a été émise en mai 19, 1994.
- **Cryptage :** L'application d'un algorithme spécifique aux données afin de modifier l'apparence

des données la rendant incompréhensible à ceux qui ne sont pas autorisées pour voir les informations.

- **Intégrité :** La propriété de s'assurer que des données sont transmises de la source à la destination sans modification non détectée.
- **Non-répudiation :** La propriété d'un récepteur pouvant montrer que l'expéditeur de quelques données a en fait envoyé les données quoique l'expéditeur pourrait plus tard désirer refuser jamais pour avoir envoyé ces données.
- **Cryptographie à clé publique :** Le chiffrement traditionnel est basé sur l'expéditeur et le récepteur d'un message connaissant et utilisant la même clé secrète. L'expéditeur emploie la clé secrète pour chiffrer le message, et le récepteur emploie la même clé secrète pour déchiffrer le message. Cette méthode est connue en tant que le « secret-key » ou « chiffrement symétrique. » Le problème principal obtient l'expéditeur et le récepteur pour convenir sur la clé secrète sans n'importe qui d'autre qui trouve. S'ils sont dans des emplacements physiques distincts, ils doivent faire confiance à un messager, ou un système téléphonique, ou un autre support de transmission pour empêcher la divulgation de la clé secrète étant communiquée. N'importe qui qui surprend ou intercepte la clé en transit peut plus tard lire, modifier, et modifier tous les messages chiffrés ou authentifiés utilisant cette clé. La production, la transmission, et la mémoire des clés s'appelle la gestion des clés ; tous les systèmes cryptographiques doivent traiter des questions de gestion des clés. Puisque toutes les clés dans un système cryptographique de secret-key doivent demeurer secrètes, le chiffrement de secret-key a souvent la difficulté fournissant à la gestion des clés sécurisée, particulièrement dans les systèmes ouverts un grand nombre d'utilisateurs.Le concept de la cryptographie à clé publique a été introduit en 1976 par Whitfield Diffie et Martin Hellman afin de résoudre le problème de gestion des clés. Dans leur concept, chaque personne obtient une paire de clés, une appelée la clé publique et l'autre appelée la clé privée. La clé publique de chaque personne est éditée tandis que la clé privée est maintenue secrète. Le besoin de l'expéditeur et du récepteur de partager les informations secrètes est éliminé et toutes les transmissions impliquent seulement des clés publiques, et aucune clé privée n'est jamais transmise ou est partagée. N'est plus il nécessaire pour faire confiance à la voie de quelques transmissions pour être sécurisé contre l'écoute illicite ou la trahison. La seule condition requise est que des clés publiques sont associées avec leurs utilisateurs d'une manière (authentifiée) de confiance (par exemple, dans un répertoire de confiance). N'importe qui peut envoyer un message confidentiel simplement à l'aide de l'information publique, mais le message peut seulement être déchiffré avec une clé privée, qui est uniquement en possession du destinataire destiné. En outre, la cryptographie à clé publique peut être aussi bien utilisée non seulement pour l'intimité (cryptage), mais pour l'authentification (signatures numériques).
- **Signatures numériques de clé publique :** Pour signer un message, une personne exécute un calcul impliquant leur clé privée et le message elle-même. La sortie s'appelle la signature numérique et est reliée au message, qui est alors envoyé. Une deuxième personne vérifie la signature en exécutant un calcul impliquant le message, la signature prétendue, et de la première la clé publique personne. Si le résultat se tient correctement dans une relation mathématique simple, la signature est vérifiée en tant qu'étant véritable. Autrement, la signature peut être frauduleuse ou le message pourrait avoir été modifié.
- **Chiffrement à clé public :** Quand une personne souhaite envoyer un message secret à une autre personne, les premières la clé publique de deuxième personne de personne consultations dans un répertoire, l'emploie pour chiffrer le message et l'envoie. La deuxième personne emploie alors leur clé privée pour déchiffrer le message et pour le lire. Personne qui

écoute dedans ne peut déchiffrer le message. N'importe qui peut envoyer un message crypté à la deuxième personne mais seulement la deuxième personne peut le lire. Clairement, une condition requise est que personne ne peut figurer la clé privée de la clé publique correspondante.
- **Analyse du trafic :** L'analyse de l'écoulement du trafic réseau afin de déduire les informations qui sont utiles à un adversaire. Les exemples d'une telle informations sont fréquence de transmission, les identités des interlocuteurs de conversation, des tailles des paquets, des identifiants d'écoulement utilisés, et ainsi de suite.

IPSec et ISAKMP

La présente partie du document couvre IPSec et ISAKMP.

IPSec a été introduit dans la version du logiciel Cisco IOS 11.3T. Il fournit un mécanisme pour la transmission de données sécurisée et se compose d'ISAKMP/Oakley et d'IPSec.

Protocole IPsec

Le protocole IPsec (RFC 1825) fournit le cryptage de couche de réseau IP et définit un nouvel ensemble d'en-têtes à ajouter aux datagrammes IP. Ces nouvelles en-têtes sont placées après l'en-tête IP et avant le protocole de la couche 4 (typiquement TCP ou UDP). Ils fournissent des informations pour sécuriser la charge utile du paquet IP, comme décrit ci-dessous :

L'En-tête d'authentification (AH) et le Protocole ESP (Encapsulating Security Payload) peuvent être utilisés indépendamment ou ensemble, bien que pour la plupart des applications juste un d'entre elles soit suffisant. Pour chacun des deux protocoles, IPSec ne définit pas les algorithmes spécifiques de Sécurité pour l'utiliser, mais plutôt, fournit un cadre ouvert pour mettre en application les algorithmes industriellement compatibles. Au commencement, la plupart des réalisations d'IPSec prennent en charge le MD5 de RSA Data Security ou l'Algorithme de hachage sûr (SHA) comme défini par le gouvernement des USA pour l'intégrité et l'authentification. Le DES est actuellement l'algorithme de chiffrement en vrac le plus généralement offert, bien que les RFC soient disponibles que définissiez comment utiliser beaucoup d'autres systèmes de cryptage, y compris l'IDÉE, le Blowfish, et le RC4.

- **OH** (voir le RFC 1826)OH est un mécanisme pour fournir l'intégrité et l'authentification fortes pour des datagrammes IP. Il peut également fournir la non-répudiation, selon laquelle l'algorithme de chiffrement est utilisé et comment introduisant est exécuté. Par exemple, l'utilisation d'un algorithme asymétrique de signature numérique, tel que la RSA, a pu fournir la non-répudiation. La confidentialité et la protection contre l'analyse du trafic ne sont pas assurées par OH. Les utilisateurs qui ont besoin de confidentialité devraient envisager d'utiliser l'ESP IP, au lieu ou en même temps que du OH. OH peut apparaître après toutes les autres en-têtes qui sont examinées à chaque saut, et avant n'importe quelles autres en-têtes qui ne sont pas examinées à un saut intermédiaire. L'en-tête d'ipv4 ou d'IPv6 juste avant OH contiendra la valeur 51 dans son prochain domaine d'en-tête (ou Protocol).
- **L'ESP** (voir le RFC 1827)L'ESP peut apparaître n'importe où après l'en-tête IP et avant le protocole de la couche transport final. L'Internet Assigned Numbers Authority a assigné Protocol le numéro 50 en ESP. L'en-tête juste avant une en-tête de l'ESP contient toujours la valeur 50 dans son prochain domaine d'en-tête (IPv6) ou de Protocol (ipv4). L'ESP se compose d'une en-tête non chiffré suivie des données cryptées. Les données cryptées

incluent les champs d'en-tête protégés de l'ESP et les données d'utilisateur protégées, qui sont un datagramme IP entier ou une trame de protocole de couche supérieure (telle que le TCP ou l'UDP).Des recherches de l'ESP IP pour fournir la confidentialité et l'intégrité par des données de chiffrage pour être protégées et plaçantes les données cryptées dans la partie données de l'IP EN PARTICULIER selon les exigences de la sécurité de l'utilisateur, ce mécanisme peuvent être utilisées pour chiffrer un segment de couche transport (tel que le TCP, l'UDP, l'ICMP, l'IGMP) ou un datagramme IP entier. Encapsuler les données protégées est nécessaire de fournir la confidentialité pour le datagramme d'origine entier. L'utilisation de cette spécification augmentera les coûts de traitement de protocole IP dans les systèmes participants et augmentera également la latence de transmissions. La latence accrue est principalement due au cryptage et au déchiffrement exigés pour chaque datagramme IP contenant l'ESP.En ESP de tunnel mode, le datagramme IP d'origine est placé dans la partie chiffrée de l'ESP et cette trame entière de l'ESP est placée dans un datagramme ayant les en-têtes décryptées IP. Les informations dans les en-têtes décryptées IP sont utilisées pour conduire le datagramme sécurisé de l'origine à la destination. Une en-tête décryptée de Routage IP pourrait être incluse entre l'en-tête IP et l'ESP.Ce mode permet à un périphérique de réseau, tel qu'un routeur, pour agir en tant que proxy d'IPSec. C'est-à-dire, le routeur exécute le cryptage au nom des hôtes. Le routeur de la source chiffre des paquets et en avant eux le long du tunnel d'IPSec. Le routeur de la destination déchiffre le datagramme IP d'origine et en avant lui en fonction au système de destination. Le principal avantage du tunnel mode est que les systèmes d'extrémité n'ont pas besoin d'être modifiés pour apprécier les avantages de la sécurité IP. Le tunnel mode se protège également contre l'analyse du trafic ; avec le tunnel mode un attaquant peut seulement déterminer les périphériques du tunnel et pas la source vraie et la destination des paquets percés un tunnel, même si ils sont identiques que les périphériques du tunnel. Comme défini par l'IETF, le mode de transport d'IPSec peut seulement être utilisé quand la source et les systèmes de destination comprennent IPSec. Dans la plupart des cas, vous déployez IPSec avec le tunnel mode. Faire te permet ainsi pour implémenter IPSec en architecture de réseau sans ne modifier le système d'exploitation ou aucune application sur vos PC, serveurs, et hôtes.En ESP de mode de transport, l'en-tête de l'ESP est insérée dans le datagramme IP immédiatement avant l'en-tête de protocole de la couche transport (telle que le TCP, l'UDP, ou l'ICMP). En ce mode, la bande passante est économisée parce qu'il n'y a aucune en-tête chiffrée IP ou options IP.Seulement la charge utile d'IP est chiffrée, et les en-têtes IP d'original sont laissées intactes. Ce mode a l'avantage d'ajouter seulement quelques octets à chaque paquet. Il permet également à des périphériques sur le réseau public pour voir la source et la destination définitives du paquet. Cette capacité te permet pour activer l'offre spéciale traitant (par exemple, qualité de service) dans le réseau intermédiaire basé sur les informations sur l'en-tête IP. Cependant, l'en-tête de la couche 4 sera chiffrée, limitant l'examen du paquet. Malheureusement, en passant l'en-tête IP en clair, le mode de transport permet à un attaquant pour exécuter une certaine analyse du trafic. Par exemple, un attaquant pourrait voir quand un Président a envoyé beaucoup de paquets à un autre Président. Cependant, l'attaquant saurait seulement que des paquets IP ont été envoyés ; l'attaquant ne pourrait pas déterminer s'ils étaient un courrier électronique ou une application différente.

ISAKMP/Oakley

Tandis qu'IPSec est le protocole réel qui protège les datagrammes IP, l'ISAKMP est le protocole qui négocie la stratégie et fournit un cadre commun pour générer les clés que les pairs d'IPSec

partagent. Il ne spécifie aucun détail de gestion des clés ou d'échange de clé et n'est lié à aucune technique de génération de clés. L'intérieur de l'ISAKMP, Cisco utilise Oakley pour le protocole d'échange de clés. Oakley te permet pour choisir entre cinq groupes « réputés ». Groupe 1 (de supports de Cisco IOS un bit 768 principal) et groupe 2 (un bit 1024 principal). Le prenez en charge pour le groupe 5 (un bit 1536 principal) a été introduit dans le Logiciel Cisco IOS version 12.1(3)T.

ISAKMP/Oakley crée authentifié, sécurise le tunnel entre deux entités, puis négocie l'association de sécurité pour IPSec. Ce processus exige que les deux entités s'authentifient entre eux et établissent des clés partagées.

Les deux interlocuteurs doivent être authentifiés entre eux. ISAKMP/Oakley prend en charge de plusieurs méthodes d'authentification. Les deux entités doivent convenir sur un protocole d'authentification commun par un procédé de négociation utilisant des signatures RSA, des nonces chiffrés par RSA, ou des clés pré-partagées.

Les deux interlocuteurs doivent avoir une clé de session partagée afin de chiffrer le tunnel ISAKMP/Oakley. Le protocole de Diffie-Hellman est utilisé pour convenir sur une clé de session commune. L'échange est authentifié comme décrit ci-dessus pour garder contre des attaques « homme-dans-le-moyennes ».

Ces deux étapes, authentification et échanges de clé, créent l'association de session ISAKMP/Oakley (SA), qui est un tunnel sécurisé entre les deux périphériques. Un côté du tunnel offre un ensemble d'algorithmes ; l'autre côté doit alors recevoir une des offres ou rejeter la connexion entière. Quand les deux côtés ont convenu sur avec quels algorithmes aux utiliser, ils doivent dériver l'élément de clé pour les utiliser pour IPSec OH, l'ESP, ou chacun des deux.

IPSec utilise une clé partagée différente qu'ISAKMP/Oakley. La clé partagée par IPSec peut être dérivée à l'aide de Diffie-Hellman de nouveau pour assurer le perfect forward secrecy, ou en régénérant le secret partagé dérivé de l'échange d'origine de Diffie-Hellman qui a généré ISAKMP/Oakley SA en le hachant avec les nombres pseudo-aléatoires (nonces). La première méthode fournit la sécurité accrue mais est plus lente. Dans la plupart des réalisations, une combinaison des deux méthodes est utilisée. C'est-à-dire, Diffie-Hellman est utilisé pour le premier échange clé, et puis les ordres locaux de stratégie quand utiliser Diffie-Hellman ou simplement une clé régénèrent. Après que ce soit complet, IPSec SA est établi.

Les signatures RSA et les nonces chiffrés par RSA exigent la clé publique du pair distant et elles exigent également du pair distant d'avoir votre clé publique locale. Des clés publiques sont permutées dans l'ISAKMP sous forme de Certificats. Ces Certificats sont obtenus par l'inscription dans l'Autorité de certification (CA). Actuellement, s'il n'y a aucun certificat dans le routeur, l'ISAKMP ne négocie pas les signatures RSA de suite de protection.

Les Routeurs de Cisco ne créent pas des Certificats. Les Routeurs créent des clés, et demandent des Certificats pour ces clés. Les Certificats, qui lient les clés des Routeurs à leurs identités, sont créés et signés par des autorités de certification. C'est une fonction d'administration, et l'autorité de certification a besoin toujours d'un certain tri de vérification que les utilisateurs sont qui ils disent qu'ils sont. Ceci signifie que vous ne pouvez pas simplement créer de nouveaux Certificats à la volée.

Les ordinateurs de communication permutent les Certificats de préexistence qu'ils ont obtenus des autorités de certification. Les Certificats eux-mêmes sont l'information publique, mais les clés privées correspondantes doivent être à la disposition de quiconque qui veut employer un certificat pour prouver l'identité. Mais ils doivent également être maintenus secrets de quiconque qui ne

devrait pas pouvoir utiliser cette identité.

Un certificat peut identifier un utilisateur ou un ordinateur. Il dépend de l'implémentation. La plupart des systèmes tôt emploient probablement un certificat pour identifier un ordinateur. Si un certificat identifie un utilisateur, la clé privée correspondant à ce certificat doit être enregistrée de telle manière qu'un autre utilisateur sur le même ordinateur ne puisse pas l'utiliser. Cela signifie généralement qu'ou la clé est maintenue chiffrée, ou que la clé est maintenue dans une carte à puce. Le cas de chiffrer-clé est susceptible d'être plus commun dans des réalisations tôt. Dans l'un ou l'autre de cas, l'utilisateur généralement doit écrire un mot de passe toutes les fois qu'une clé est lancée.

Remarque: ISAKMP/Oakley utilise le port UDP 500 pour la négociation. OH contient 51 dans le domaine de Protocol et l'ESP contient 50 dans le domaine de Protocol. Assurez-vous que vous ne filtrez pas ces derniers.

Pour plus d'informations sur la terminologie utilisée dans cet état technique, référez-vous à la section de définitions.

Configuration de chiffrement de réseau-couche de Cisco IOS pour IPSec et ISAKMP

Les configurations Cisco IOS fonctionnantes d'échantillon dans ce document ont été livré directement des routeurs de laboratoire. La seule modification apportée à eux était la suppression des configurations d'interface indépendantes. Tout les contenu ici est provenu librement des ressources disponibles sur l'Internet ou dans la section Informations connexes à la fin de ce document.

Échantillon 1 : Clés pré-partagées d'ISAKMP

L'authentification par l'intermédiaire des clés pré-partagées est une alternative principale non publique. Suivre cette méthode, chaque pair partage une clé secrète qui a été hors bande permuté et configuré dans le routeur. La capacité pour que chaque côté explique la connaissance de ce secret (sans la mentionner explicitement) authentifie l'échange. Cette méthode est adéquate pour de petites installations mais a des problèmes d'évolution. Une clé pré-partagée de « sharedkey » est utilisée ci-dessous. Si les hôtes partagent les clés pré-partagées basées sur adresse, ils doivent utiliser leur identité d'adresse, qui est le par défaut en logiciel de Cisco IOS, ainsi elle n'affiche pas dans la configuration :

```
crypto isakmp identity address
```

Remarque: Il y a des situations où l'ISAKMP ne peut pas établir la stratégie et les clés pour IPSec. S'il n'y a aucun certificat défini dans le routeur et il y a seulement des méthodes d'authentification clé clé publiques dans la stratégie ISAKMP, ou s'il n'y a aucun certificat et aucune clés pré-partagées pour le pair (partagé directement par adresse ou par une adresse Internet qui a été configurée avec cette adresse), alors l'ISAKMP ne peut pas être en pourparlers avec le pair et IPSec ne fonctionne pas.

Le graphique suivant représente le schéma de réseau pour cette configuration.

Voici les configurations pour deux Routeurs (Cisco 2511 et Cisco 2516) authentification faisante dos à dos d'IPSec et d'ISAKMP basée sur une clé pré-partagée. Des lignes de commentaire sont

indiquées par un point d'exclamation comme premier caractère et sont ignorées si entré dans le routeur. Dans la configuration ci-dessous, les commentaires précèdent certaines lignes de configuration afin de les décrire.

Configuration de Cisco 2511

```
cl-2513-2A#write terminal Building configuration...
Current configuration: ! version 11.3 service timestamps
debug uptime service timestamps log uptime no service
password-encryption ! hostname cl-2513-2A ! !---
Override the default policy and use !--- preshared keys
for authentication. crypto isakmp policy 1
authentication pre-share group 2 ! !--- Define our
secret shared key so !--- you do not have to use RSA
keys. crypto isakmp key sharedkey address 20.20.20.20 !
!--- These are the authentication and encryption !---
settings defined for "auth2", !--- which is later
applied to the crypto map. crypto ipsec transform-set
auth2 esp-des esp-sha-hmac ! !--- The crypto map where
you define your peer, !--- transform auth2, and your
access list. crypto map test 10 ipsec-isakmp set peer
20.20.20.20 set transform-set auth2 match address 133 !
interface Ethernet0 ip address 50.50.50.50 255.255.255.0
! interface Serial0 ip address 20.20.20.21 255.255.255.0
no ip route-cache no ip mroute-cache !--- Nothing
happens unless you apply !--- the crypto map to an
interface. crypto map test ! ip route 0.0.0.0 0.0.0.0
20.20.20.20 ! !--- This is the access list referenced !-
-- in the crypto map; never use "any". !--- You are
encrypting traffic between !--- the remote Ethernet
LANs. access-list 133 permit ip 50.50.50.0 0.0.0.255
60.60.60.0 0.0.0.255 ! line con 0 line aux 0 line vty 0
4 login ! end
```

Configuration de Cisco 2516

```
cl-2513-2B#show run Building configuration... Current
configuration: ! version 11.3 service timestamps debug
uptime service timestamps log uptime no service
password-encryption ! hostname cl-2513-2B ! ip subnet-
zero ! !--- Override the default policy and use !---
preshared keys for authentication. crypto isakmp policy
1 authentication pre-share group 2 !--- Define the
secret shared key so you !--- do not have to use RSA
keys. crypto isakmp key sharedkey address 20.20.20.21 !-
-- These are the authentication and encryption !---
settings defined for "auth2," !--- which is later
applied to the crypto map. crypto ipsec transform-set
auth2 esp-des esp-sha-hmac !--- The crypto map where you
define the peer, !--- transform auth2, and the access
list. crypto map test 10 ipsec-isakmp set peer
20.20.20.21 set transform-set auth2 match address 144 !
interface Ethernet0 ip address 60.60.60.60 255.255.255.0
no ip directed-broadcast ! !--- Nothing happens unless
you apply !--- the crypto map to an interface. interface
Serial0 ip address 20.20.20.20 255.255.255.0 no ip
directed-broadcast no ip route-cache no ip mroute-cache
clockrate 800000 crypto map test ! ip classless ip route
0.0.0.0 0.0.0.0 20.20.20.21 ! !--- This is the access
list referenced !--- in the crypto map; never use "any".
!--- You are encrypting traffic between !--- the remote
Ethernet LANs. access-list 144 permit ip 60.60.60.0
0.0.0.255 50.50.50.0 0.0.0.255 ! line con 0 transport
input none line aux 0 line vty 0 4 login ! end
```

Ce qui suit est **met au point la** sortie de commande.

```
--------------- Preshare with RSA key defined
(need to remove RSA keys) -----

*Mar  1 00:14:48.579: ISAKMP (10): incorrect policy settings.
 Unable to initiate.
*Mar  1 00:14:48.587: ISAKMP (11): incorrect policy settings.
 Unable to initiate......

--------------- Preshare, wrong hostname ---------------

ISAKMP: no pre-shared key based on hostname wan-2511.cisco.com!
%CRYPTO-6-IKMP_MODE_FAILURE: Processing of Aggressive mode
 failed with peer at
20.20.20.21
--------------- Preshare, incompatable policy -------------
wan2511#
*Mar  1 00:33:34.839: ISAKMP (17): processing SA payload. message ID = 0
*Mar  1 00:33:34.843: ISAKMP (17): Checking ISAKMP transform 1
 against priority 1 policy
*Mar  1 00:33:34.843: ISAKMP:      encryption DES-CBC
*Mar  1 00:33:34.843: ISAKMP:      hash SHA
*Mar  1 00:33:34.847: ISAKMP:      default group 2
*Mar  1 00:33:34.847: ISAKMP:      auth pre-share
*Mar  1 00:33:34.847: ISAKMP:      life type in seconds
*Mar  1 00:33:34.851: ISAKMP:      life duration (basic) of 240
*Mar  1 00:33:34.851: ISAKMP (17): atts are acceptable.
 Next payload is 0
*Mar  1 00:33:43.735: ISAKMP (17): processing KE payload.
 message ID = 0
*Mar  1 00:33:54.307: ISAKMP (17): processing NONCE payload.
 message ID = 0
*Mar  1 00:33:54.311: ISAKMP (17): processing ID payload.
 message ID = 0
*Mar  1 00:33:54.331: ISAKMP (17): SKEYID state generated
*Mar  1 00:34:04.867: ISAKMP (17): processing HASH payload.
 message ID = 0
*Mar  1 00:34:04.879: ISAKMP (17): SA has been authenticated
*Mar  1 00:34:06.151: ISAKMP (17): processing SA payload.
 message ID = -1357683133
*Mar  1 00:34:06.155: ISAKMP (17): Checking IPSec proposal 1
*Mar  1 00:34:06.155: ISAKMP: transform 1, AH_MD5_HMAC
*Mar  1 00:34:06.159: ISAKMP:   attributes in transform:
*Mar  1 00:34:06.159: ISAKMP:      encaps is 1
*Mar  1 00:34:06.159: ISAKMP:      SA life type in seconds
*Mar  1 00:34:06.163: ISAKMP:      SA life duration (basic) of 3600
*Mar  1 00:34:06.163: ISAKMP:      SA life type in kilobytes
*Mar  1 00:34:06.163: ISAKMP:      SA life duration (VPI) of
  0x0 0x46 0x50 0x0
*Mar  1 00:34:06.167: ISAKMP (17): atts not acceptable.
 Next payload is 0
*Mar  1 00:34:06.171: ISAKMP (17): Checking IPSec proposal 1
*Mar  1 00:34:06.171: ISAKMP: transform 1, ESP_DES
*Mar  1 00:34:06.171: ISAKMP:   attributes in transform:
*Mar  1 00:34:06.175: ISAKMP:      encaps is 1
*Mar  1 00:34:06.175: ISAKMP:      SA life type in seconds
*Mar  1 00:34:06.175: ISAKMP:      SA life duration (basic) of 3600
*Mar  1 00:34:06.179: ISAKMP:      SA life type in kilobytes
*Mar  1 00:34:06.179: ISAKMP:      SA life duration (VPI) of
  0x0 0x46 0x50 0x0
*Mar  1 00:34:06.183: ISAKMP:      HMAC algorithm is SHA
*Mar  1 00:34:06.183: ISAKMP (17): atts are acceptable.
```

```
*Mar  1 00:34:06.187: ISAKMP (17): SA not acceptable!
%CRYPTO-6-IKMP_MODE_FAILURE: Processing of Quick mode failed
with peer at 20.20.20.20
wan2511#

----------------- preshare, debug isakmp -------------------

wan2511#
*Mar  1 00:06:54.179: ISAKMP (1): processing SA payload.
 message ID = 0
*Mar  1 00:06:54.179: ISAKMP (1): Checking ISAKMP transform 1
 against priority 1 policy
*Mar  1 00:06:54.183: ISAKMP:      encryption DES-CBC
*Mar  1 00:06:54.183: ISAKMP:      hash SHA
*Mar  1 00:06:54.183: ISAKMP:      default group 2
*Mar  1 00:06:54.187: ISAKMP:      auth pre-share
*Mar  1 00:06:54.187: ISAKMP:      life type in seconds
*Mar  1 00:06:54.187: ISAKMP:      life duration (basic) of 240
*Mar  1 00:06:54.191: ISAKMP (1): atts are acceptable.
 Next payload is 0
*Mar  1 00:07:02.955: ISAKMP (1): processing KE payload.
 message ID = 0
*Mar  1 00:07:13.411: ISAKMP (1): processing NONCE payload.
 message ID = 0
*Mar  1 00:07:13.415: ISAKMP (1): processing ID payload.
 message ID = 0
*Mar  1 00:07:13.435: ISAKMP (1): SKEYID state generated
*Mar  1 00:07:23.903: ISAKMP (1): processing HASH payload.
 message ID = 0
*Mar  1 00:07:23.915: ISAKMP (1): SA has been authenticated
*Mar  1 00:07:25.187: ISAKMP (1): processing SA payload.
 message ID = 1435594195
*Mar  1 00:07:25.187: ISAKMP (1): Checking IPSec proposal 1
*Mar  1 00:07:25.191: ISAKMP: transform 1, AH_SHA_HMAC
*Mar  1 00:07:25.191: ISAKMP:   attributes in transform:
*Mar  1 00:07:25.191: ISAKMP:      encaps is 1
*Mar  1 00:07:25.195: ISAKMP:      SA life type in seconds
*Mar  1 00:07:25.195: ISAKMP:      SA life duration (basic) of 3600
*Mar  1 00:07:25.195: ISAKMP:      SA life type in kilobytes
*Mar  1 00:07:25.199: ISAKMP:      SA life duration (VPI) of
  0x0 0x46 0x50 0x0
*Mar  1 00:07:25.203: ISAKMP (1): atts are acceptable.
*Mar  1 00:07:25.203: ISAKMP (1): Checking IPSec proposal 1
*Mar  1 00:07:25.207: ISAKMP: transform 1, ESP_DES
*Mar  1 00:07:25.207: ISAKMP:   attributes in transform:
*Mar  1 00:07:25.207: ISAKMP:      encaps is 1
*Mar  1 00:07:25.211: ISAKMP:      SA life type in seconds
*Mar  1 00:07:25.211: ISAKMP:      SA life duration (basic) of 3600
*Mar  1 00:07:25.211: ISAKMP:      SA life type in kilobytes
*Mar  1 00:07:25.215: ISAKMP:      SA life duration (VPI) of
  0x0 0x46 0x50 0x0
*Mar  1 00:07:25.215: ISAKMP:      HMAC algorithm is SHA
*Mar  1 00:07:25.219: ISAKMP (1): atts are acceptable.
*Mar  1 00:07:25.223: ISAKMP (1): processing NONCE payload.
 message ID = 1435594195
*Mar  1 00:07:25.227: ISAKMP (1): processing ID payload.
 message ID = 1435594195
*Mar  1 00:07:25.227: ISAKMP (1): processing ID payload.
 message ID = 1435594195
*Mar  1 00:07:25.639: ISAKMP (1): Creating IPSec SAs
*Mar  1 00:07:25.643:         inbound SA from 20.20.20.20
     to 20.20.20.21
        (proxy 60.60.60.0      to 50.50.50.0     )
*Mar  1 00:07:25.647:         has spi 85067251 and
```

```
 conn_id 3 and flags 4
*Mar  1 00:07:25.647:         lifetime of 3600 seconds
*Mar  1 00:07:25.647:         lifetime of 4608000 kilobytes
*Mar  1 00:07:25.651:         outbound SA from 20.20.20.21
     to 20.20.20.20
        (proxy 50.50.50.0      to 60.60.60.0     )
*Mar  1 00:07:25.655:         has spi 57872298 and
 conn_id 4 and flags 4
*Mar  1 00:07:25.655:         lifetime of 3600 seconds
*Mar  1 00:07:25.655:         lifetime of 4608000 kilobytes
*Mar  1 00:07:25.659: ISAKMP (1): Creating IPSec SAs
*Mar  1 00:07:25.659:         inbound SA from 20.20.20.20
   to 20.20.20.21
        (proxy 60.60.60.0      to 50.50.50.0     )
*Mar  1 00:07:25.663:         has spi 538316566 and
 conn_id 5 and flags 4
*Mar  1 00:07:25.663:         lifetime of 3600 seconds
*Mar  1 00:07:25.667:         lifetime of 4608000 kilobytes
*Mar  1 00:07:25.667:         outbound SA from 20.20.20.21
 to 20.20.20.20
        (proxy 50.50.50.0      to 60.60.60.0     )
*Mar  1 00:07:25.671:         has spi 356000275 and
 conn_id 6 and flags 4
*Mar  1 00:07:25.671:         lifetime of 3600 seconds
*Mar  1 00:07:25.675:         lifetime of 4608000 kilobytes
wan2511#

----------------- preshare debug ipsec -------------------
wan2511#
*Mar  1 00:05:26.947: IPSEC(validate_proposal_request):
proposal part #1,
  (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20,
    dest_proxy= 50.50.50.0/0.0.0.0/0/0,
    src_proxy= 60.60.60.0/0.0.0.16/0/0,
    protocol= AH, transform= ah-sha-hmac ,
    lifedur= 0s and 0kb,
    spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
*Mar  1 00:05:26.955: IPSEC(validate_proposal_request):
proposal part #2,
  (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20,
    dest_proxy= 50.50.50.0/0.0.0.0/0/0,
    src_proxy= 60.60.60.0/0.0.0.16/0/0,
    protocol= ESP, transform= esp-des esp-sha-hmac ,
    lifedur= 0s and 0kb,
    spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
*Mar  1 00:05:26.967: IPSEC(key_engine): got a queue event...
*Mar  1 00:05:26.971: IPSEC(spi_response): getting
 spi 203563166 for SA
        from 20.20.20.20     to 20.20.20.21     for prot 2
*Mar  1 00:05:26.975: IPSEC(spi_response): getting
 spi 194838793 for SA
        from 20.20.20.20     to 20.20.20.21     for prot 3
*Mar  1 00:05:27.379: IPSEC(key_engine): got a queue event...
*Mar  1 00:05:27.379: IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20,
    dest_proxy= 50.50.50.0/255.255.255.0/0/0,
    src_proxy= 60.60.60.0/255.255.255.0/0/0,
    protocol= AH, transform= ah-sha-hmac ,
    lifedur= 3600s and 4608000kb,
    spi= 0xC22209E(203563166), conn_id= 3, keysize= 0, flags= 0x4
*Mar  1 00:05:27.387: IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 20.20.20.21, dest= 20.20.20.20,
    src_proxy= 50.50.50.0/255.255.255.0/0/0,
    dest_proxy= 60.60.60.0/255.255.255.0/0/0,
```

```
    protocol= AH, transform= ah-sha-hmac ,
    lifedur= 3600s and 4608000kb,
    spi= 0x15E010D(22937869), conn_id= 4, keysize= 0, flags= 0x4
*Mar  1 00:05:27.395: IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20,
    dest_proxy= 50.50.50.0/255.255.255.0/0/0,
    src_proxy= 60.60.60.0/255.255.255.0/0/0,
    protocol= ESP, transform= esp-des esp-sha-hmac ,
    lifedur= 3600s and 4608000kb,
    spi= 0xB9D0109(194838793), conn_id= 5, keysize= 0, flags= 0x4
*Mar  1 00:05:27.403: IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 20.20.20.21, dest= 20.20.20.20,
    src_proxy= 50.50.50.0/255.255.255.0/0/0,
    dest_proxy= 60.60.60.0/255.255.255.0/0/0,
    protocol= ESP, transform= esp-des esp-sha-hmac ,
    lifedur= 3600s and 4608000kb,
    spi= 0xDED0AB4(233638580), conn_id= 6, keysize= 0, flags= 0x4
*Mar  1 00:05:27.415: IPSEC(create_sa): sa created,
  (sa) sa_dest= 20.20.20.21, sa_prot= 51,
    sa_spi= 0xC22209E(203563166),
    sa_trans= ah-sha-hmac , sa_conn_id= 3
*Mar  1 00:05:27.419: IPSEC(create_sa): sa created,
  (sa) sa_dest= 20.20.20.20, sa_prot= 51,
    sa_spi= 0x15E010D(22937869),
    sa_trans= ah-sha-hmac , sa_conn_id= 4
*Mar  1 00:05:27.423: IPSEC(create_sa): sa created,
  (sa) sa_dest= 20.20.20.21, sa_prot= 50,
    sa_spi= 0xB9D0109(194838793),
    sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
*Mar  1 00:05:27.427: IPSEC(create_sa): sa created,
  (sa) sa_dest= 20.20.20.20, sa_prot= 50,
    sa_spi= 0xDED0AB4(233638580),
    sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
wan2511#

--------------- Preshare, good connection ------
wan2511#
*Mar  1 00:09:45.095: ISAKMP (1): processing SA payload.
 message ID = 0
*Mar  1 00:09:45.099: ISAKMP (1): Checking ISAKMP transform
 1 against priority 1 policy
*Mar  1 00:09:45.099: ISAKMP:      encryption DES-CBC
*Mar  1 00:09:45.103: ISAKMP:      hash SHA
*Mar  1 00:09:45.103: ISAKMP:      default group 2
*Mar  1 00:09:45.103: ISAKMP:      auth pre-share
*Mar  1 00:09:45.107: ISAKMP:      life type in seconds
*Mar  1 00:09:45.107: ISAKMP:      life duration (basic) of 240
*Mar  1 00:09:45.107: ISAKMP (1): atts are acceptable.
Next payload is 0
*Mar  1 00:09:53.867: ISAKMP (1): processing KE payload.
 message ID = 0
*Mar  1 00:10:04.323: ISAKMP (1): processing NONCE payload.
 message ID = 0
*Mar  1 00:10:04.327: ISAKMP (1): processing ID payload.
 message ID = 0
*Mar  1 00:10:04.347: ISAKMP (1): SKEYID state generated
*Mar  1 00:10:15.103: ISAKMP (1): processing HASH payload.
 message ID = 0
*Mar  1 00:10:15.115: ISAKMP (1): SA has been authenticated
*Mar  1 00:10:16.391: ISAKMP (1): processing SA payload.
 message ID = 800032287
*Mar  1 00:10:16.391: ISAKMP (1): Checking IPSec proposal 1
*Mar  1 00:10:16.395: ISAKMP: transform 1, AH_SHA_HMAC
*Mar  1 00:10:16.395: ISAKMP:   attributes in transform:
```

```
*Mar  1 00:10:16.395: ISAKMP:      encaps is 1
*Mar  1 00:10:16.399: ISAKMP:      SA life type in seconds
*Mar  1 00:10:16.399: ISAKMP:      SA life duration (basic) of 3600
*Mar  1 00:10:16.399: ISAKMP:      SA life type in kilobytes
*Mar  1 00:10:16.403: ISAKMP:      SA life duration (VPI) of
  0x0 0x46 0x50 0x0
*Mar  1 00:10:16.407: ISAKMP (1): atts are acceptable.
*Mar  1 00:10:16.407: ISAKMP (1): Checking IPSec proposal 1
*Mar  1 00:10:16.411: ISAKMP: transform 1, ESP_DES
*Mar  1 00:10:16.411: ISAKMP:   attributes in transform:
*Mar  1 00:10:16.411: ISAKMP:      encaps is 1
*Mar  1 00:10:16.415: ISAKMP:      SA life type in seconds
*Mar  1 00:10:16.415: ISAKMP:      SA life duration (basic) of 3600
*Mar  1 00:10:16.415: ISAKMP:      SA life type in kilobytes
*Mar  1 00:10:16.419: ISAKMP:      SA life duration (VPI) of
  0x0 0x46 0x50 0x0
*Mar  1 00:10:16.419: ISAKMP:      HMAC algorithm is SHA
*Mar  1 00:10:16.423: ISAKMP (1): atts are acceptable.
*Mar  1 00:10:16.427: IPSEC(validate_proposal_request):
 proposal part #1,
  (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20,
    dest_proxy= 50.50.50.0/0.0.0.0/0/0,
    src_proxy= 60.60.60.0/0.0.0.16/0/0,
    protocol= AH, transform= ah-sha-hmac ,
    lifedur= 0s and 0kb,
    spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
*Mar  1 00:10:16.435: IPSEC(validate_proposal_request):
 proposal part #2,
  (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20,
    dest_proxy= 50.50.50.0/0.0.0.0/0/0,
    src_proxy= 60.60.60.0/0.0.0.16/0/0,
    protocol= ESP, transform= esp-des esp-sha-hmac ,
    lifedur= 0s and 0kb,
    spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4
*Mar  1 00:10:16.443: ISAKMP (1): processing NONCE payload.
 message ID = 800032287
*Mar  1 00:10:16.443: ISAKMP (1): processing ID payload.
 message ID = 800032287
*Mar  1 00:10:16.447: ISAKMP (1): processing ID payload.
 message ID = 800032287
*Mar  1 00:10:16.451: IPSEC(key_engine): got a queue event...
*Mar  1 00:10:16.455: IPSEC(spi_response): getting
spi 16457800 for SA
        from 20.20.20.20     to 20.20.20.21     for prot 2
*Mar  1 00:10:16.459: IPSEC(spi_response): getting
spi 305534655 for SA
        from 20.20.20.20     to 20.20.20.21     for prot 3
*Mar  1 00:10:17.095: ISAKMP (1): Creating IPSec SAs
*Mar  1 00:10:17.095:         inbound SA from 20.20.20.20
     to 20.20.20.21
        (proxy 60.60.60.0      to 50.50.50.0     )
*Mar  1 00:10:17.099:         has spi 16457800 and conn_id 3
 and flags 4
*Mar  1 00:10:17.103:         lifetime of 3600 seconds
*Mar  1 00:10:17.103:         lifetime of 4608000 kilobytes
*Mar  1 00:10:17.103:         outbound SA from 20.20.20.21
     to 20.20.20.20
        (proxy 50.50.50.0      to 60.60.60.0     )
*Mar  1 00:10:17.107:         has spi 507120385 and conn_id 4
 and flags 4
*Mar  1 00:10:17.111:         lifetime of 3600 seconds
*Mar  1 00:10:17.111:         lifetime of 4608000 kilobytes
*Mar  1 00:10:17.115: ISAKMP (1): Creating IPSec SAs
*Mar  1 00:10:17.115:         inbound SA from 20.20.20.20
```

```
to 20.20.20.21
        (proxy 60.60.60.0      to 50.50.50.0     )
*Mar  1 00:10:17.119:         has spi 305534655 and
conn_id 5 and flags 4
*Mar  1 00:10:17.119:         lifetime of 3600 seconds
*Mar  1 00:10:17.123:         lifetime of 4608000 kilobytes
*Mar  1 00:10:17.123:         outbound SA from 20.20.20.21
     to 20.20.20.20
        (proxy 50.50.50.0      to 60.60.60.0     )
*Mar  1 00:10:17.127:         has spi 554175376 and
conn_id 6 and flags 4
*Mar  1 00:10:17.127:         lifetime of 3600 seconds
*Mar  1 00:10:17.131:         lifetime of 4608000 kilobytes
*Mar  1 00:10:17.139: IPSEC(key_engine): got a queue event...
*Mar  1 00:10:17.143: IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20,
    dest_proxy= 50.50.50.0/255.255.255.0/0/0,
    src_proxy= 60.60.60.0/255.255.255.0/0/0,
    protocol= AH, transform= ah-sha-hmac ,
    lifedur= 3600s and 4608000kb,
    spi= 0xFB2048(16457800), conn_id= 3, keysize= 0,
 flags= 0x4
*Mar  1 00:10:17.151: IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 20.20.20.21, dest= 20.20.20.20,
    src_proxy= 50.50.50.0/255.255.255.0/0/0,
    dest_proxy= 60.60.60.0/255.255.255.0/0/0,
    protocol= AH, transform= ah-sha-hmac ,
    lifedur= 3600s and 4608000kb,
    spi= 0x1E3A0B01(507120385), conn_id= 4, keysize= 0,
 flags= 0x4
*Mar  1 00:10:17.159: IPSEC(initialize_sas): ,
  (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20,
    dest_proxy= 50.50.50.0/255.255.255.0/0/0,
    src_proxy= 60.60.60.0/255.255.255.0/0/0,
    protocol= ESP, transform= esp-des esp-sha-hmac ,
    lifedur= 3600s and 4608000kb,
    spi= 0x123616BF(305534655), conn_id= 5, keysize= 0,
 flags= 0x4
*Mar  1 00:10:17.167: IPSEC(initialize_sas): ,
  (key eng. msg.) SRC= 20.20.20.21, dest= 20.20.20.20,
    src_proxy= 50.50.50.0/255.255.255.0/0/0,
    dest_proxy= 60.60.60.0/255.255.255.0/0/0,
    protocol= ESP, transform= esp-des esp-sha-hmac ,
    lifedur= 3600s and 4608000kb,
    spi= 0x21080B90(554175376), conn_id= 6, keysize= 0,
 flags= 0x4
*Mar  1 00:10:17.175: IPSEC(create_sa): sa created,
  (sa) sa_dest= 20.20.20.21, sa_prot= 51,
    sa_spi= 0xFB2048(16457800),
    sa_trans= ah-sha-hmac , sa_conn_id= 3
*Mar  1 00:10:17.179: IPSEC(create_sa): sa created,
  (sa) sa_dest= 20.20.20.20, sa_prot= 51,
    sa_spi= 0x1E3A0B01(507120385),
    sa_trans= ah-sha-hmac , sa_conn_id= 4
*Mar  1 00:10:17.183: IPSEC(create_sa): sa created,
  (sa) sa_dest= 20.20.20.21, sa_prot= 50,
    sa_spi= 0x123616BF(305534655),
    sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5
*Mar  1 00:10:17.187: IPSEC(create_sa): sa created,
  (sa) sa_dest= 20.20.20.20, sa_prot= 50,
    sa_spi= 0x21080B90(554175376),
    sa_trans= esp-des esp-sha-hmac , sa_conn_id= 6
*Mar  1 00:10:36.583: ISADB: reaper checking SA, conn_id = 1
wan2511#
```

Échantillon 2 : ISAKMP : Authentification RSA-chiffrée

Dans ce scénario, une clé secrète partagée n'est pas créée. Chaque routeur génère sa propre clé RSA. Alors chaque routeur doit configurer la clé publique RSA du pair. C'est un processus manuel et a des limites évidentes d'évolution. En d'autres termes, un routeur doit avoir une clé publique RSA pour chaque pair avec lequel elle souhaite avoir une association de sécurité.

Le document suivant représente le schéma de réseau pour cette configuration d'échantillon.

Dans cet exemple, chaque routeur génère une paire de clés RSA (vous ne voyez jamais la clé privée RSA que vous générez) et configure la clé publique RSA des pairs de distant.

```
wan2511(config)#crypto key generate rsa The name for the keys will be: wan2511.cisco.com Choose
the size of the key modulus in the range of 360 to 2048 for your General Purpose Keys. Choosing
a key modulus greater than 512 may take a few minutes. How many bits in the modulus [512]:
Generating RSA keys ... [OK] wan2511(config)#^Z wan2511# wan2511#show crypto key mypubkey rsa %
Key pair was generated at: 00:09:04 UTC Mar 1 1993 Key name: wan2511.cisco.com Usage: General
Purpose Key Key Data: 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00E9007B E5CD7DC8
6E1C0423 92044254 92C972AD 0CCE9796 86797EAA B6C4EFF0 0F0A5378 6AFAE43B 3A2BD92F 98039DAC
08741E82 5D9053C4 D9CFABC1 AB54E0E2 BB020301 0001 wan2511# wan2511(config)#crypto key pubkey-
chain rsa wan2511(config-pubkey-chain)#named-key wan2516.cisco.com wan2511(config-pubkey-
key)#key-string Enter a public key as a hexidecimal number .... wan2511(config-pubkey)#$86F70D
01010105 00034B00 30480241 00DC3DDC 59885F14 wan2511(config-pubkey)#$D918DE FC7ADB76 B0B9DD1A
ABAF4884 009E758C 4064C699 wan2511(config-pubkey)#$220CB9 31E267F8 0259C640 F8DE4169 1F020301
0001 wan2511(config-pubkey)#quit wan2511(config-pubkey-key)#^Z wan2511# wan2511#show crypto key
pubkey-chain rsa Key name: wan2516.cisco.com Key usage: general purpose Key source: manually
entered Key data: 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00DC3DDC 59885F14
1AB30DCB 794AB5C7 82D918DE FC7ADB76 B0B9DD1A ABAF4884 009E758C 4064C699 3BC9D17E C47581DC
50220CB9 31E267F8 0259C640 F8DE4169 1F020301 0001 wan2511# wan2511#write terminal Building
configuration... Current configuration: ! version 11.3 service timestamps debug datetime msec no
service password-encryption ! hostname wan2511 ! enable password ww ! no ip domain-lookup ip
host wan2516.cisco.com 20.20.20.20 ip domain-name cisco.com ! crypto isakmp policy 1
authentication rsa-encr group 2 lifetime 240 crypto isakmp identity hostname ! crypto ipsec
transform-set auth2 ah-sha-hmac esp-des esp-sha-hmac ! crypto map test 10 ipsec-isakmp set peer
20.20.20.20 set transform-set auth2 match address 133 ! crypto key pubkey-chain rsa named-key
wan2516.cisco.com key-string 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00DC3DDC
59885F14 1AB30DCB 794AB5C7 82D918DE FC7ADB76 B0B9DD1A ABAF4884 009E758C 4064C699 3BC9D17E
C47581DC 50220CB9 31E267F8 0259C640 F8DE4169 1F020301 0001 quit ! interface Ethernet0 ip address
50.50.50.50 255.255.255.0 ! interface Serial0 ip address 20.20.20.21 255.255.255.0 encapsulation
ppp no ip mroute-cache crypto map test ! interface Serial1 no ip address shutdown ! ip classless
ip route 0.0.0.0 0.0.0.0 10.11.19.254 ip route 60.0.0.0 255.0.0.0 20.20.20.20 access-list 133
permit ip 50.50.50.0 0.0.0.255 60.60.60.0 0.0.0.255 ! line con 0 exec-timeout 0 0 password ww
login line 1 6 modem InOut transport input all speed 115200 flowcontrol hardware line 7 16
autoselect ppp modem InOut transport input all speed 115200 flowcontrol hardware line aux 0
login local modem InOut transport input all flowcontrol hardware line vty 0 4 password ww login
! end wan2511# ----------------- wan2516(config)#crypto key generate rsa The name for the keys
will be: wan2516.cisco.com Choose the size of the key modulus in the range of 360 to 2048 for
your General Purpose Keys. Choosing a key modulus greater than 512 may take a few minutes. How
many bits in the modulus [512]: Generating RSA keys ... [OK] wan2516#show crypto key mypubkey
rsa % Key pair was generated at: 00:06:35 UTC Mar 1 1993 Key name: wan2516.cisco.com Usage:
General Purpose Key Key Data: 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00DC3DDC
59885F14 1AB30DCB 794AB5C7 82D918DE FC7ADB76 B0B9DD1A ABAF4884 009E758C 4064C699 3BC9D17E
C47581DC 50220CB9 31E267F8 0259C640 F8DE4169 1F020301 0001 wan2516# -------
wan2516(config)#crypto key exchange ? dss Exchange DSS keys ------- wan2516(config)#crypto key
pubkey-chain rsa wan2516(config-pubkey-chain)#named-key wan2511.cisco.com wan2516(config-pubkey-
key)#key-string Enter a public key as a hexidecimal number .... wan2516(config-pubkey)#$86F70D
01010105 00034B00 30480241 00E9007B E5CD7DC8 wan2516(config-pubkey)#$C972AD 0CCE9796 86797EAA
B6C4EFF0 0F0A5378 6AFAE43B wan2516(config-pubkey)#$741E82 5D9053C4 D9CFABC1 AB54E0E2 BB020301
0001 wan2516(config-pubkey)#quit wan2516(config-pubkey-key)#^Z wan2516#show crypto key pubkey
rsa Key name: wan2511.cisco.com Key usage: general purpose Key source: manually entered Key
data: 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00E9007B E5CD7DC8 6E1C0423 92044254
```

92C972AD 0CCE9796 86797EAA B6C4EFF0 0F0A5378 6AFAE43B 3A2BD92F 98039DAC 08741E82 5D9053C4 D9CFABC1 AB54E0E2 BB020301 0001 wan2516# ---------- wan2516#**write terminal** Building configuration... Current configuration: ! version 11.3 no service pad service timestamps debug datetime msec no service password-encryption service udp-small-servers service tcp-small-servers ! hostname wan2516 ! enable password ww ! no ip domain-lookup ip host wan2511.cisco.com 20.20.20.21 ip domain-name cisco.com ! crypto isakmp policy 1 authentication rsa-encr group 2 lifetime 240 crypto isakmp identity hostname ! crypto ipsec transform-set auth2 ah-sha-hmac esp-des esp-sha-hmac ! crypto map test 10 ipsec-isakmp set peer 20.20.20.21 set transform-set auth2 match address 144 ! crypto key pubkey-chain rsa named-key wan2511.cisco.com key-string 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00E9007B E5CD7DC8 6E1C0423 92044254 92C972AD 0CCE9796 86797EAA B6C4EFF0 0F0A5378 6AFAE43B 3A2BD92F 98039DAC 08741E82 5D9053C4 D9CFABC1 AB54E0E2 BB020301 0001 quit ! hub ether 0 1 link-test auto-polarity ! interface Loopback0 ip address 70.70.70.1 255.255.255.0 no ip route-cache no ip mroute-cache ! interface Ethernet0 ip address 60.60.60.60 255.255.255.0 ! interface Serial0 ip address 20.20.20.20 255.255.255.0 encapsulation ppp clockrate 2000000 crypto map test ! interface Serial1 no ip address no ip route-cache no ip mroute-cache shutdown ! interface BRI0 no ip address no ip route-cache no ip mroute-cache shutdown ! ip default-gateway 20.20.20.21 ip classless ip route 0.0.0.0 0.0.0.0 20.20.20.21 access-list 144 permit ip 60.60.60.0 0.0.0.255 50.50.50.0 0.0.0.255 ! line con 0 exec-timeout 0 0 password ww login line aux 0 password ww login modem InOut transport input all flowcontrol hardware line vty 0 4 password ww login ! end wan2516# ---------- RSA-enc missing RSA Keys --------- *Mar 1 00:02:51.147: ISAKMP: No cert, and no keys (public or pre-shared) with remote peer 20.20.20.21 *Mar 1 00:02:51.151: ISAKMP: No cert, and no keys (public or pre-shared) with remote peer 20.20.20.21 ---------- RSA-enc good connection ---------- wan2511# *Mar 1 00:21:46.375: ISAKMP (1): processing SA payload. message ID = 0 *Mar 1 00:21:46.379: ISAKMP (1): Checking ISAKMP transform 1 against priority 1 policy *Mar 1 00:21:46.379: ISAKMP: encryption DES-CBC *Mar 1 00:21:46.379: ISAKMP: hash SHA *Mar 1 00:21:46.383: ISAKMP: default group 2 *Mar 1 00:21:46.383: ISAKMP: auth RSA encr *Mar 1 00:21:46.383: ISAKMP: life type in seconds *Mar 1 00:21:46.387: ISAKMP: life duration (basic) of 240 *Mar 1 00:21:46.387: ISAKMP (1): atts are acceptable. Next payload is 0 *Mar 1 00:21:46.391: Crypto engine 0: generate alg param *Mar 1 00:21:55.159: CRYPTO_ENGINE: Dh phase 1 status: 0 *Mar 1 00:21:55.163: CRYPTO: DH gen phase 1 status for conn_id 1 slot 0:OK *Mar 1 00:21:55.167: ISAKMP (1): Unable to get router cert to find DN! *Mar 1 00:21:55.171: ISAKMP (1): SA is doing RSA encryption authentication *Mar 1 00:22:04.351: ISAKMP (1): processing KE payload. message ID = 0 *Mar 1 00:22:04.351: Crypto engine 0: generate alg param *Mar 1 00:22:14.767: CRYPTO: DH gen phase 2 status for conn_id 1 slot 0:OK *Mar 1 00:22:14.771: ISAKMP (1): processing ID payload. message ID = 0 *Mar 1 00:22:14.775: Crypto engine 0: RSA decrypt with private key *Mar 1 00:22:15.967: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:16.167: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:16.367: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:16.579: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:16.787: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:16.987: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:17.215: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:17.431: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:17.539: CRYPTO: RSA private decrypt finished with status=OK *Mar 1 00:22:17.543: ISAKMP (1): processing NONCE payload. message ID = 0 *Mar 1 00:22:17.543: Crypto engine 0: RSA decrypt with private key *Mar 1 00:22:18.735: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:18.947: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:19.155: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:19.359: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:19.567: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:19.767: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:19.975: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:20.223: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:20.335: CRYPTO: RSA private decrypt finished with status=OK *Mar 1 00:22:20.347: Crypto engine 0: create ISAKMP SKEYID for conn id 1 *Mar 1 00:22:20.363: ISAKMP (1): SKEYID state generated *Mar 1 00:22:20.367: Crypto engine 0: RSA encrypt with public key *Mar 1 00:22:20.567: CRYPTO: RSA public encrypt finished with status=OK *Mar 1 00:22:20.571: Crypto engine 0: RSA encrypt with public key *Mar 1 00:22:20.767: CRYPTO: RSA public encrypt finished with status=OK *Mar 1 00:22:20.775: ISAKMP (1): processing KE payload. message ID = 0 *Mar 1 00:22:20.775: ISAKMP (1): processing ID payload. message ID = 0 *Mar 1 00:22:20.779: Crypto engine 0: RSA decrypt with private key *Mar 1 00:22:21.959: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:22.187: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:22.399: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:22.599: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:22.811: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:23.019: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:23.223: CRYPTO_ENGINE: key

process suspended and continued *Mar 1 00:22:23.471: CRYPTO_ENGINE: key process suspended and continued *Mar 1 00:22:23.583: CRYPTO: RSA private decrypt finished with status=OK *Mar 1 00:22:23.583: ISAKMP (1): processing NONCE payload. message ID = 0 %CRYPTO-6-IKMP_AUTH_FAIL: Authentication method 4 failed with host 20.20.20.20 %CRYPTO-6-IKMP_MODE_FAILURE: Processing of Main mode failed with peer at 20.20.20.20 *Mar 1 00:22:36.955: ISAKMP (1): processing HASH payload. message ID = 0 *Mar 1 00:22:36.959: generate hmac context for conn id 1 *Mar 1 00:22:36.971: ISAKMP (1): SA has been authenticated *Mar 1 00:22:36.975: generate hmac context for conn id 1 *Mar 1 00:22:37.311: generate hmac context for conn id 1 *Mar 1 00:22:37.319: ISAKMP (1): processing SA payload. message ID = -114148384 *Mar 1 00:22:37.319: ISAKMP (1): Checking IPSec proposal 1 *Mar 1 00:22:37.323: ISAKMP: transform 1, AH_SHA_HMAC *Mar 1 00:22:37.323: ISAKMP: attributes in transform: *Mar 1 00:22:37.327: ISAKMP: encaps is 1 *Mar 1 00:22:37.327: ISAKMP: SA life type in seconds *Mar 1 00:22:37.327: ISAKMP: SA life duration (basic) of 3600 *Mar 1 00:22:37.331: ISAKMP: SA life type in kilobytes *Mar 1 00:22:37.331: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50 0x0 *Mar 1 00:22:37.335: ISAKMP (1): atts are acceptable. *Mar 1 00:22:37.335: ISAKMP (1): Checking IPSec proposal 1 *Mar 1 00:22:37.339: ISAKMP: transform 1, ESP_DES *Mar 1 00:22:37.339: ISAKMP: attributes in transform: *Mar 1 00:22:37.339: ISAKMP: encaps is 1 *Mar 1 00:22:37.343: ISAKMP: SA life type in seconds *Mar 1 00:22:37.343: ISAKMP: SA life duration (basic) of 3600 *Mar 1 00:22:37.347: ISAKMP: SA life type in kilobytes *Mar 1 00:22:37.347: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50 0x0 *Mar 1 00:22:37.351: ISAKMP: HMAC algorithm is SHA *Mar 1 00:22:37.351: ISAKMP (1): atts are acceptable. *Mar 1 00:22:37.355: IPSEC(validate_proposal_request): proposal part #1, (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20, dest_proxy= 50.50.50.0/0.0.0.0/0/0, src_proxy= 60.60.60.0/0.0.0.16/0/0, protocol= AH, transform= ah-sha-hmac , lifedur= 0s and 0kb, spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4 *Mar 1 00:22:37.363: IPSEC(validate_proposal_request): proposal part #2, (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20, dest_proxy= 50.50.50.0/0.0.0.0/0/0, src_proxy= 60.60.60.0/0.0.0.16/0/0, protocol= ESP, transform= esp-des esp-sha-hmac , lifedur= 0s and 0kb, spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4 *Mar 1 00:22:37.371: ISAKMP (1): processing NONCE payload. message ID = -114148384 *Mar 1 00:22:37.375: ISAKMP (1): processing ID payload. message ID = -114148384 *Mar 1 00:22:37.375: ISAKMP (1): processing ID payload. message ID = -114148384 *Mar 1 00:22:37.379: IPSEC(key_engine): got a queue event... *Mar 1 00:22:37.383: IPSEC(spi_response): getting spi 531040311 for SA from 20.20.20.20 to 20.20.20.21 for prot 2 *Mar 1 00:22:37.387: IPSEC(spi_response): getting spi 220210147 for SA from 20.20.20.20 to 20.20.20.21 for prot 3 *Mar 1 00:22:37.639: generate hmac context for conn id 1 *Mar 1 00:22:37.931: generate hmac context for conn id 1 *Mar 1 00:22:37.975: ISAKMP (1): Creating IPSec SAs *Mar 1 00:22:37.975: inbound SA from 20.20.20.20 to 20.20.20.21 (proxy 60.60.60.0 to 50.50.50.0) *Mar 1 00:22:37.979: has spi 531040311 and conn_id 2 and flags 4 *Mar 1 00:22:37.979: lifetime of 3600 seconds *Mar 1 00:22:37.983: lifetime of 4608000 kilobytes *Mar 1 00:22:37.983: outbound SA from 20.20.20.21 to 20.20.20.20 (proxy 50.50.50.0 to 60.60.60.0) *Mar 1 00:22:37.987: has spi 125043658 and conn_id 3 and flags 4 *Mar 1 00:22:37.987: lifetime of 3600 seconds *Mar 1 00:22:37.991: lifetime of 4608000 kilobytes *Mar 1 00:22:37.991: ISAKMP (1): Creating IPSec SAs *Mar 1 00:22:37.991: inbound SA from 20.20.20.20 to 20.20.20.21 (proxy 60.60.60.0 to 50.50.50.0) *Mar 1 00:22:37.995: has spi 220210147 and conn_id 4 and flags 4 *Mar 1 00:22:37.999: lifetime of 3600 seconds *Mar 1 00:22:37.999: lifetime of 4608000 kilobytes *Mar 1 00:22:38.003: outbound SA from 20.20.20.21 to 20.20.20.20 (proxy 50.50.50.0 to 60.60.60.0) *Mar 1 00:22:38.003: has spi 299247102 and conn_id 5 and flags 4 *Mar 1 00:22:38.007: lifetime of 3600 seconds *Mar 1 00:22:38.007: lifetime of 4608000 kilobytes *Mar 1 00:22:38.011: IPSEC(key_engine): got a queue event... *Mar 1 00:22:38.015: IPSEC(initialize_sas): , (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20, dest_proxy= 50.50.50.0/255.255.255.0/0/0, src_proxy= 60.60.60.0/255.255.255.0/0/0, protocol= AH, transform= ah-sha-hmac , lifedur= 3600s and 4608000kb, spi= 0x1FA70837(531040311), conn_id= 2, keysize= 0, flags= 0x4 *Mar 1 00:22:38.023: IPSEC(initialize_sas): , (key eng. msg.) SRC= 20.20.20.21, dest= 20.20.20.20, src_proxy= 50.50.50.0/255.255.255.0/0/0, dest_proxy= 60.60.60.0/255.255.255.0/0/0, protocol= AH, transform= ah-sha-hmac , lifedur= 3600s and 4608000kb, spi= 0x77403CA(125043658), conn_id= 3, keysize= 0, flags= 0x4 *Mar 1 00:22:38.031: IPSEC(initialize_sas): , (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20, dest_proxy= 50.50.50.0/255.255.255.0/0/0, src_proxy= 60.60.60.0/255.255.255.0/0/0, protocol= ESP, transform= esp-des esp-sha-hmac , lifedur= 3600s and 4608000kb, spi= 0xD2023E3(220210147), conn_id= 4, keysize= 0, flags= 0x4 *Mar 1 00:22:38.039: IPSEC(initialize_sas): , (key eng. msg.) SRC= 20.20.20.21, dest= 20.20.20.20, src_proxy= 50.50.50.0/255.255.255.0/0/0, dest_proxy= 60.60.60.0/255.255.255.0/0/0, protocol= ESP, transform= esp-des esp-sha-hmac , lifedur= 3600s and 4608000kb, spi= 0x11D625FE(299247102), conn_id= 5, keysize= 0, flags= 0x4 *Mar 1 00:22:38.047: IPSEC(create_sa): sa created, (sa) sa_dest= 20.20.20.21, sa_prot= 51, sa_spi= 0x1FA70837(531040311), sa_trans= ah-sha-hmac , sa_conn_id= 2 *Mar 1 00:22:38.051: IPSEC(create_sa): sa created, (sa) sa_dest= 20.20.20.20, sa_prot= 51, sa_spi=

0x77403CA(125043658), sa_trans= ah-sha-hmac , sa_conn_id= 3 *Mar 1 00:22:38.055: IPSEC(create_sa): sa created, (sa) sa_dest= 20.20.20.21, sa_prot= 50, sa_spi= 0xD2023E3(220210147), sa_trans= esp-des esp-sha-hmac , sa_conn_id= 4 *Mar 1 00:22:38.063: IPSEC(create_sa): sa created, (sa) sa_dest= 20.20.20.20, sa_prot= 50, sa_spi= 0x11D625FE(299247102), sa_trans= esp-des esp-sha-hmac , sa_conn_id= 5 wan2511# ---------- RSA-ENC ISAKMP debugs good connection --- wan2511# *Mar 1 00:27:23.279: ISAKMP (6): processing SA payload. message ID = 0 *Mar 1 00:27:23.279: ISAKMP (6): Checking ISAKMP transform 1 against priority 1 policy *Mar 1 00:27:23.283: ISAKMP: encryption DES-CBC *Mar 1 00:27:23.283: ISAKMP: hash SHA *Mar 1 00:27:23.283: ISAKMP: default group 2 *Mar 1 00:27:23.287: ISAKMP: auth RSA encr *Mar 1 00:27:23.287: ISAKMP: life type in seconds *Mar 1 00:27:23.287: ISAKMP: life duration (basic) of 240 *Mar 1 00:27:23.291: ISAKMP (6): atts are acceptable. Next payload is 0 *Mar 1 00:27:32.055: ISAKMP (6): Unable to get router cert to find DN! *Mar 1 00:27:32.055: ISAKMP (6): SA is doing RSA encryption authentication *Mar 1 00:27:41.183: ISAKMP (6): processing KE payload. message ID = 0 *Mar 1 00:27:51.779: ISAKMP (6): processing ID payload. message ID = 0 *Mar 1 00:27:54.507: ISAKMP (6): processing NONCE payload. message ID = 0 *Mar 1 00:27:57.239: ISAKMP (6): SKEYID state generated *Mar 1 00:27:57.627: ISAKMP (6): processing KE payload. message ID = 0 *Mar 1 00:27:57.631: ISAKMP (6): processing ID payload. message ID = 0 *Mar 1 00:28:00.371: ISAKMP (6): processing NONCE payload. message ID = 0 %CRYPTO-6-IKMP_AUTH_FAIL: Authentication method 4 failed with host 20.20.20.20 %CRYPTO-6-IKMP_MODE_FAILURE: Processing of Main mode failed with peer at 20.20.20.20 *Mar 1 00:28:13.587: ISAKMP (6): processing HASH payload. message ID = 0 *Mar 1 00:28:13.599: ISAKMP (6): SA has been authenticated *Mar 1 00:28:13.939: ISAKMP (6): processing SA payload. message ID = -161552401 *Mar 1 00:28:13.943: ISAKMP (6): Checking IPSec proposal 1 *Mar 1 00:28:13.943: ISAKMP: transform 1, AH_SHA_HMAC *Mar 1 00:28:13.943: ISAKMP: attributes in transform: *Mar 1 00:28:13.947: ISAKMP: encaps is 1 *Mar 1 00:28:13.947: ISAKMP: SA life type in seconds *Mar 1 00:28:13.947: ISAKMP: SA life duration (basic) of 3600 *Mar 1 00:28:13.951: ISAKMP: SA life type in kilobytes *Mar 1 00:28:13.951: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50 0x0 *Mar 1 00:28:13.955: ISAKMP (6): atts are acceptable. *Mar 1 00:28:13.959: ISAKMP (6): Checking IPSec proposal 1 *Mar 1 00:28:13.959: ISAKMP: transform 1, ESP_DES *Mar 1 00:28:13.959: ISAKMP: attributes in transform: *Mar 1 00:28:13.963: ISAKMP: encaps is 1 *Mar 1 00:28:13.963: ISAKMP: SA life type in seconds *Mar 1 00:28:13.963: ISAKMP: SA life duration (basic) of 3600 *Mar 1 00:28:13.967: ISAKMP: SA life type in kilobytes *Mar 1 00:28:13.967: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50 0x0 *Mar 1 00:28:13.971: ISAKMP: HMAC algorithm is SHA *Mar 1 00:28:13.971: ISAKMP (6): atts are acceptable. *Mar 1 00:28:13.975: ISAKMP (6): processing NONCE payload. message ID = -161552401 *Mar 1 00:28:13.979: ISAKMP (6): processing ID payload. message ID = -161552401 *Mar 1 00:28:13.979: ISAKMP (6): processing ID payload. message ID = -161552401 *Mar 1 00:28:14.391: ISAKMP (6): Creating IPSec SAs *Mar 1 00:28:14.391: inbound SA from 20.20.20.20 to 20.20.20.21 (proxy 60.60.60.0 to 50.50.50.0) *Mar 1 00:28:14.395: has spi 437593758 and conn_id 7 and flags 4 *Mar 1 00:28:14.399: lifetime of 3600 seconds *Mar 1 00:28:14.399: lifetime of 4608000 kilobytes *Mar 1 00:28:14.403: outbound SA from 20.20.20.21 to 20.20.20.20 (proxy 50.50.50.0 to 60.60.60.0) *Mar 1 00:28:14.403: has spi 411835612 and conn_id 8 and flags 4 *Mar 1 00:28:14.407: lifetime of 3600 seconds *Mar 1 00:28:14.407: lifetime of 4608000 kilobytes *Mar 1 00:28:14.411: ISAKMP (6): Creating IPSec SAs *Mar 1 00:28:14.411: inbound SA from 20.20.20.20 to 20.20.20.21 (proxy 60.60.60.0 to 50.50.50.0) *Mar 1 00:28:14.415: has spi 216990519 and conn_id 9 and flags 4 *Mar 1 00:28:14.415: lifetime of 3600 seconds *Mar 1 00:28:14.419: lifetime of 4608000 kilobytes *Mar 1 00:28:14.419: outbound SA from 20.20.20.21 to 20.20.20.20 (proxy 50.50.50.0 to 60.60.60.0) *Mar 1 00:28:14.423: has spi 108733569 and conn_id 10 and flags 4 *Mar 1 00:28:14.423: lifetime of 3600 seconds *Mar 1 00:28:14.427: lifetime of 4608000 kilobytes wan2511# ------------------------- RSA-enc IPSEC debug ------- wan2511# *Mar 1 00:30:32.155: ISAKMP (11): Unable to get router cert to find DN! wan2511#**show debug** Cryptographic Subsystem: Crypto IPSEC debugging is on wan2511# wan2511# wan2511# %CRYPTO-6-IKMP_AUTH_FAIL: Authentication method 4 failed with host 20.20.20.20 %CRYPTO-6-IKMP_MODE_FAILURE: Processing of Main mode failed with peer at 20.20.20.20 *Mar 1 00:31:13.931: IPSEC(validate_proposal_request): proposal part #1, (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20, dest_proxy= 50.50.50.0/0.0.0.0/0/0, src_proxy= 60.60.60.0/0.0.0.16/0/0, protocol= AH, transform= ah-sha-hmac , lifedur= 0s and 0kb, spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4 *Mar 1 00:31:13.935: IPSEC(validate_proposal_request): proposal part #2, (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20, dest_proxy= 50.50.50.0/0.0.0.0/0/0, src_proxy= 60.60.60.0/0.0.0.16/0/0, protocol= ESP, transform= esp-des esp-sha-hmac , lifedur= 0s and 0kb, spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4 *Mar 1 00:31:13.947: IPSEC(key_engine): got a queue event... *Mar 1 00:31:13.951: IPSEC(spi_response): getting spi 436869446 for SA from 20.20.20.20 to 20.20.20.21 for prot 2 *Mar 1 00:31:13.955: IPSEC(spi_response): getting spi 285609740 for SA from 20.20.20.20 to 20.20.20.21 for prot 3 *Mar 1 00:31:14.367: IPSEC(key_engine): got a queue event... *Mar 1 00:31:14.367: IPSEC(initialize_sas): , (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20,

```
dest_proxy= 50.50.50.0/255.255.255.0/0/0, src_proxy= 60.60.60.0/255.255.255.0/0/0, protocol= AH,
transform= ah-sha-hmac , lifedur= 3600s and 4608000kb, spi= 0x1A0A1946(436869446), conn_id= 12,
keysize= 0, flags= 0x4 *Mar 1 00:31:14.375: IPSEC(initialize_sas): , (key eng. msg.) SRC=
20.20.20.21, dest= 20.20.20.20, src_proxy= 50.50.50.0/255.255.255.0/0/0, dest_proxy=
60.60.60.0/255.255.255.0/0/0, protocol= AH, transform= ah-sha-hmac , lifedur= 3600s and
4608000kb, spi= 0x2C40706(46401286), conn_id= 13, keysize= 0, flags= 0x4 *Mar 1 00:31:14.383:
IPSEC(initialize_sas): , (key eng. msg.) dest= 20.20.20.21, SRC= 20.20.20.20, dest_proxy=
50.50.50.0/255.255.255.0/0/0, src_proxy= 60.60.60.0/255.255.255.0/0/0, protocol= ESP, transform=
esp-des esp-sha-hmac , lifedur= 3600s and 4608000kb, spi= 0x11060F0C(285609740), conn_id= 14,
keysize= 0, flags= 0x4 *Mar 1 00:31:14.391: IPSEC(initialize_sas): , (key eng. msg.) SRC=
20.20.20.21, dest= 20.20.20.20, src_proxy= 50.50.50.0/255.255.255.0/0/0, dest_proxy=
60.60.60.0/255.255.255.0/0/0, protocol= ESP, transform= esp-des esp-sha-hmac , lifedur= 3600s
and 4608000kb, spi= 0x12881335(310907701), conn_id= 15, keysize= 0, flags= 0x4 *Mar 1
00:31:14.399: IPSEC(create_sa): sa created, (sa) sa_dest= 20.20.20.21, sa_prot= 51, sa_spi=
0x1A0A1946(436869446), sa_trans= ah-sha-hmac , sa_conn_id= 12 *Mar 1 00:31:14.407:
IPSEC(create_sa): sa created, (sa) sa_dest= 20.20.20.20, sa_prot= 51, sa_spi=
0x2C40706(46401286), sa_trans= ah-sha-hmac , sa_conn_id= 13 *Mar 1 00:31:14.411:
IPSEC(create_sa): sa created, (sa) sa_dest= 20.20.20.21, sa_prot= 50, sa_spi=
0x11060F0C(285609740), sa_trans= esp-des esp-sha-hmac , sa_conn_id= 14 *Mar 1 00:31:14.415:
IPSEC(create_sa): sa created, (sa) sa_dest= 20.20.20.20, sa_prot= 50, sa_spi=
0x12881335(310907701), sa_trans= esp-des esp-sha-hmac , sa_conn_id= 15 wan2511#
```

Échantillon 3 : ISAKMP : RSA-SIG Authentication/CA

Cet exemple utilise les signatures RSA, qui exigent l'utilisation d'un serveur CA. Chaque pair obtient des Certificats du serveur CA (c'est habituellement un poste de travail qui est configuré pour délivrer des Certificats). Quand les deux pairs ont les Certificats CA valides, ils permutent automatiquement des clés publiques RSA les uns avec les autres en tant qu'élément de la négociation ISAKMP. Tout ce qui est exigé dans ce scénario est pour que chaque pair s'inscrive à un CA et a obtenu un certificat. Un pair ne doit plus maintenir des clés publiques RSA de tous les pairs dans un réseau.

En outre, notez qu'une stratégie ISAKMP n'est pas spécifiée parce que vous utilisez la stratégie par défaut, qui est affichée ci-dessous :

```
lab-isdn1#show crypto isakmp policy Default protection suite encryption algorithm: DES - Data
Encryption Standard (56 bit keys). hash algorithm: Secure Hash Standard authentication method:
Rivest-Shamir-Adleman Signature Diffie-Hellman group: #1 (768 bit) lifetime: 86400 seconds, no
volume limit
```

D'abord, définissez l'adresse Internet du serveur CA, et générez la clé RSA.

```
test1-isdn(config)#ip host cert-author 10.19.54.46 test1-isdn(config)#crypto key gen rsa usage
The name for the keys will be: test1-isdn.cisco.com Choose the size of the key modulus in the
range of 360 to 2048 for your Signature Keys. Choosing a key modulus greater than 512 may take a
few minutes. How many bits in the modulus [512]: Generating RSA keys ... [OK] Choose the size of
the key modulus in the range of 360 to 2048 for your Encryption Keys. Choosing a key modulus
greater than 512 may take a few minutes. How many bits in the modulus [512]: Generating RSA keys
... [OK]
```

Ensuite, la configuration CA est définie avec une balise appelée le "test1-RNIS-ultra" et définit l'URL de nom CA. Puis, authentifiez avec le serveur CA et obtenez un certificat. En conclusion, continuez à vérifier pour vous veiller pour avoir reçu les Certificats « disponibles » pour l'usage.

```
test1-isdn(config)#crypto ca identity test1-isdn-ultra test1-isdn(ca-identity)#enrollment url
http://cert-author test1-isdn(ca-identity)#crl optional test1-isdn(ca-identity)#exit ----------
---------------------- test1-isdn(config)#crypto ca authenticate test1-isdn-ultra Certificate
has the following attributes: Fingerprint: 71CA5A98 78828EF8 4987BA95 57830E5F % Do you accept
this certificate? [yes/no]: yes Apr 3 14:08:56.329: CRYPTO_PKI: http connection opened Apr 3
14:08:56.595: CRYPTO__PKI: All enrollment requests completed. Apr 3 14:08:56.599: CRYPTO_PKI:
transaction GetCACert completed Apr 3 14:08:56.599: CRYPTO_PKI: CA certificate received test1-
isdn(config)# ---------------------------------- test1-isdn(config)#crypto ca enroll test1-
```

isdn-ultra % Start certificate enrollment .. % Create a challenge password. You will need to verbally provide this password to the CA Administrator in order to revoke your certificate. For security reasons your password will not be saved in the configuration. Please make a note of it. Password: Re-enter password: % The subject name in the certificate will be: test1-isdn.cisco.com % Include the router serial number in the subject name? [yes/no]: yes % The serial number in the certificate will be: 04922418 % Include an IP address in the subject name? [yes/no]: yes Interface: bri0 Request certificate from CA? [yes/no]: yes % Certificate request sent to Certificate Authority % The certificate request fingerprint will be displayed. % The 'show crypto ca certificate' command will also show the fingerprint. ---------- status: pending ---------- test1-isdn#**show crypto ca certificate** CA Certificate Status: Available Certificate Serial Number: 3051DF7169BEE31B821DFE4B3A338E5F Key Usage: Not Set Certificate Subject Name Name: test1-isdn.cisco.com IP Address: 10.18.117.189 Serial Number: 04922418 Status: Pending Key Usage: Signature Fingerprint: B1566229 472B1DDB 01A072C0 8202A985 00000000 Certificate Subject Name Name: test1-isdn.cisco.com IP Address: 10.18.117.189 Serial Number: 04922418 Status: Pending Key Usage: Encryption Fingerprint: 1EA39C07 D1B26FC7 7AD08BF4 ACA3AABD 00000000 ---------- status: available ---------- test1-isdn#**show crypto ca certificate** Certificate Subject Name Name: test1-isdn.cisco.com Serial Number: 04922418 Status: Available Certificate Serial Number: 1BAFCBCA71F0434B59D192FAFB37D376 Key Usage: Encryption CA Certificate Status: Available Certificate Serial Number: 3051DF7169BEE31B821DFE4B3A338E5F Key Usage: Not Set Certificate Subject Name Name: test1-isdn.cisco.com Serial Number: 04922418 Status: Available Certificate Serial Number: 4B39EE2866814279CBA7534496DE1D99 Key Usage: Signature test1-isdn#

Le graphique suivant représente le schéma de réseau pour cette configuration d'échantillon.

La configuration d'échantillon ci-dessous est prise de deux Cisco 1600 Routeurs qui ont précédemment obtenu des Certificats CA (comme affiché ci-dessus) et ont l'intention de faire l'ISAKMP avec « RSA-Sig » comme stratégie d'authentification. Seulement le trafic entre les deux réseaux locaux distants d'Ethernets est chiffré.

lab-isdn1#**write terminal** Building configuration... Current configuration: ! version 11.3 service timestamps debug datetime msec no service password-encryption service udp-small-servers service tcp-small-servers ! hostname lab-isdn1 ! enable secret 5 1VdPY$uA/BIVeEm9UAFEm.PPJFc. ! username lab-isdn password 0 cisco ip host ciscoca-ultra 171.69.54.46 ip host lab-isdn 12.12.12.12 ip domain-name cisco.com ip name-server 171.68.10.70 ip name-server 171.68.122.99 isdn switch-type basic-ni1 ! crypto ipsec transform-set mypolicy ah-sha-hmac esp-des esp-sha-hmac ! crypto map test 10 ipsec-isakmp set peer 12.12.12.12 set transform-set mypolicy match address 144 ! crypto ca identity bubba enrollment url http://ciscoca-ultra crl optional crypto ca certificate chain bubba certificate 3E1ED472BDA2CE0163FB6B0B004E5EEE 308201BC 30820166 A0030201 0202103E 1ED472BD A2CE0163 FB6B0B00 4E5EEE30 0D06092A 864886F7 0D010104 05003042 31163014 06035504 0A130D43 6973636F 20537973 74656D73 3110300E 06035504 0B130744 65767465 73743116 30140603 55040313 0D434953 434F4341 2D554C54 5241301E 170D3938 30343038 30303030 30305A17 0D393930 34303832 33353935 395A303B 31273025 06092A86 4886F70D 01090216 18737461 6E6E6F75 732D6973 646E312E 63697363 6F2E636F 6D311030 0E060355 04051307 35363739 39383730 5C300D06 092A8648 86F70D01 01010500 034B0030 48024100 D2D125FF BBFC6E56 93CB4385 5473C165 BC7CCAF6 45C35BED 554BAA0B 119AFA6F 0853F574 5E0B8492 2E39B5FA 84C4DD05 C19AA625 8184395C 6CBC7FA4 614F6177 02030100 01A33F30 3D300B06 03551D0F 04040302 05203023 0603551D 11041C30 1A821873 74616E6E 6F75732D 6973646E 312E6369 73636F2E 636F6D30 09060355 1D130402 3000300D 06092A86 4886F70D 01010405 00034100 04AF83B8 FE95F5D9 9C07C105 F1E88F1A 9320CE7D 0FA540CF 44C77829 FC85C94B 8CB4CA32 85FF9655 8E47AC9A B9D6BF1A 0C4846DE 5CB07C8E A32038EC 8AFD161A quit certificate ca 3051DF7169BEE31B821DFE4B3A338E5F 30820182 3082012C A0030201 02021030 51DF7169 BEE31B82 1DFE4B3A 338E5F30 0D06092A 864886F7 0D010104 05003042 31163014 06035504 0A130D43 6973636F 20537973 74656D73 3110300E 06035504 0B130744 65767465 73743116 30140603 55040313 0D434953 434F4341 2D554C54 5241301E 170D3937 31323032 30313036 32385A17 0D393831 32303230 31303632 385A3042 31163014 06035504 0A130D43 6973636F 20537973 74656D73 3110300E 06035504 0B130744 65767465 73743116 30140603 55040313 0D434953 434F4341 2D554C54 5241305C 300D0609 2A864886 F70D0101 01050003 4B003048 024100C1 B69D7BF6 34E4EE28 A84E0DC6 FCA4DEA8 04D89E50 C5EBE862 39D51890 D0D4B732 678BDBF2 80801430 E5E56E7C C126E2DD DBE9695A DF8E5BA7 E67BAE87 29375302 03010001 300D0609 2A864886 F70D0101 04050003 410035AA 82B5A406 32489413 A7FF9A9A E349E5B4 74615E05 058BA3CE 7C5F00B4 019552A5 E892D2A3 86763A1F 2852297F C68EECE1 F41E9A7B 2F38D02A B1D2F817 3F7B quit certificate 503968D890F7D409475B7280162754D2 308201BC 30820166 A0030201 02021050 3968D890 F7D40947 5B728016 2754D230 0D06092A 864886F7 0D010104 05003042 31163014 06035504 0A130D43 6973636F 20537973 74656D73 3110300E 06035504 0B130744 65767465

73743116 30140603 55040313 0D434953 434F4341 2D554C54 5241301E 170D3938 30343038 30303030
30305A17 0D393930 34303832 33353935 395A303B 31273025 06092A86 4886F70D 01090216 18737461
6E6E6F75 732D6973 646E312E 63697363 6F2E636F 6D311030 0E060355 04051307 35363739 39383730
5C300D06 092A8648 86F70D01 01010500 034B0030 48024100 BECE2D8C B32E6B09 0ADE0D46 AF8D4A1F
37850034 35D0C729 3BF91518 0C9E4CF8 1A6A43AE E4F04687 B8E2859D 33D5CE04 2E5DDEA6 3DA54A31
2AD4255A 756014CB 02030100 01A33F30 3D300B06 03551D0F 04040302 07803023 0603551D 11041C30
1A821873 74616E6E 6F75732D 6973646E 312E6369 73636F2E 636F6D30 09060355 1D130402 3000300D
06092A86 4886F70D 01010405 00034100 B3AF6E71 CBD9AEDD A4711B71 6897F2CE D669A23A EE47B92B
B2BE942A 422DF4A5 7ACB9433 BD17EC7A BB3721EC E7D1175F 5C62BC58 C409F805 19691FBD FD925138 quit !
interface Ethernet0 ip address 40.40.40.40 255.255.255.0 no ip mroute-cache ! interface BRI0 ip
address 12.12.12.13 255.255.255.0 encapsulation ppp no ip mroute-cache dialer idle-timeout 99999
dialer map ip 12.12.12.12 name lab-isdn 4724171 dialer hold-queue 40 dialer-group 1 isdn spid1
919472411800 4724118 isdn spid2 919472411901 4724119 ppp authentication chap crypto map test !
ip classless ip route 0.0.0.0 0.0.0.0 12.12.12.12 access-list 144 permit ip 40.40.40.0 0.0.0.255
20.20.20.0 0.0.0.255 dialer-list 1 protocol ip permit ! line con 0 exec-timeout 0 0 line vty 0 4
password ww login ! end lab-isdn1# ------------------ lab-isdn#**write terminal** Building
configuration... Current configuration: ! version 11.3 service timestamps debug datetime msec no
service password-encryption service udp-small-servers service tcp-small-servers ! hostname lab-
isdn ! enable secret 5 1oNe1$wDbhBdcN6x9Y5gfuMjqh10 ! username lab-isdn1 password 0 cisco ip
host ciscoca-ultra 171.69.54.46 ip host lab-isdn1 12.12.12.13 ip domain-name cisco.com ip name-
server 171.68.10.70 ip name-server 171.68.122.99 isdn switch-type basic-ni1 ! crypto ipsec
transform-set mypolicy ah-sha-hmac esp-des esp-sha-hmac ! crypto map test 10 ipsec-isakmp set
peer 12.12.12.13 set transform-set mypolicy match address 133 ! crypto ca identity lab
enrollment url http://ciscoca-ultra crl optional crypto ca certificate chain lab certificate
44FC6C531FC3446927E4EE307A806B20 308201E0 3082018A A0030201 02021044 FC6C531F C3446927 E4EE307A
806B2030 0D06092A 864886F7 0D010104 05003042 31163014 06035504 0A130D43 6973636F 20537973
74656D73 3110300E 06035504 0B130744 65767465 73743116 30140603 55040313 0D434953 434F4341
2D554C54 5241301E 170D3938 30343038 30303030 30305A17 0D393930 34303832 33353935 395A305A
31263024 06092A86 4886F70D 01090216 17737461 6E6E6F75 732D6973 646E2E63 6973636F 2E636F6D
311E301C 060A2B06 0104012A 020B0201 130E3137 312E3638 2E313137 2E313839 3110300E 06035504
05130735 36373939 3139305C 300D0609 2A864886 F70D0101 01050003 4B003048 024100B8 F4A17A70
FAB5C2E3 39186513 486779C7 61EF0AC1 3B6CFF83 810E6D28 B3E4C034 CD803CFF 5158C270 28FEBCDE
CB6EF2D4 83BDD9B3 EAF915DB 78266E96 500CD702 03010001 A3443042 300B0603 551D0F04 04030205
20302806 03551D11 0421301F 82177374 616E6E6F 75732D69 73646E2E 63697363 6F2E636F 6D8704AB
4475BD30 09060355 1D130402 3000300D 06092A86 4886F70D 01010405 00034100 BF65B931 0F960195
ABDD41D5 622743D9 C12B5499 B3A8EB30 5005E6CC 7FDF7C5B 51D13EB8 D46187E5 A1E7F711 AEB7B33B
AA4C6728 7A4BA692 00A44A05 C5CF973F quit certificate ca 3051DF7169BEE31B821DFE4B3A338E5F
30820182 3082012C A0030201 02021030 51DF7169 BEE31B82 1DFE4B3A 338E5F30 0D06092A 864886F7
0D010104 05003042 31163014 06035504 0A130D43 6973636F 20537973 74656D73 3110300E 06035504
0B130744 65767465 73743116 30140603 55040313 0D434953 434F4341 2D554C54 5241301E 170D3937
31323032 30313036 32385A17 0D393831 32303230 31303632 385A3042 31163014 06035504 0A130D43
6973636F 20537973 74656D73 3110300E 06035504 0B130744 65767465 73743116 30140603 55040313
0D434953 434F4341 2D554C54 5241305C 300D0609 2A864886 F70D0101 01050003 4B003048 024100C1
B69D7BF6 34E4EE28 A84E0DC6 FCA4DEA8 04D89E50 C5EBE862 39D51890 D0D4B732 678BDBF2 80801430
E5E56E7C C126E2DD DBE9695A DF8E5BA7 E67BAE87 29375302 03010001 300D0609 2A864886 F70D0101
04050003 410035AA 82B5A406 32489413 A7FF9A9A E349E5B4 74615E05 058BA3CE 7C5F00B4 019552A5
E892D2A3 86763A1F 2852297F C68EECE1 F41E9A7B 2F38D02A B1D2F817 3F7B quit certificate
52A46D5D10B18A6F51E6BC735A36508C 308201E0 3082018A A0030201 02021052 A46D5D10 B18A6F51 E6BC735A
36508C30 0D06092A 864886F7 0D010104 05003042 31163014 06035504 0A130D43 6973636F 20537973
74656D73 3110300E 06035504 0B130744 65767465 73743116 30140603 55040313 0D434953 434F4341
2D554C54 5241301E 170D3938 30343038 30303030 30305A17 0D393930 34303832 33353935 395A305A
31263024 06092A86 4886F70D 01090216 17737461 6E6E6F75 732D6973 646E2E63 6973636F 2E636F6D
311E301C 060A2B06 0104012A 020B0201 130E3137 312E3638 2E313137 2E313839 3110300E 06035504
05130735 36373939 3139305C 300D0609 2A864886 F70D0101 01050003 4B003048 024100D7 71AD5672
B487A019 5ECD1954 6F919A3A 6270102E 5A9FF4DC 7A608480 FB27A181 715335F4 399D3E57 7F72B323
BF0620AB 60C371CF 4389BA4F C60EE6EA 21E06302 03010001 A3443042 300B0603 551D0F04 04030207
80302806 03551D11 0421301F 82177374 616E6E6F 75732D69 73646E2E 63697363 6F2E636F 6D8704AB
4475BD30 09060355 1D130402 3000300D 06092A86 4886F70D 01010405 00034100 8AD45375 54803CF3
013829A8 8DB225A8 25342160 94546F3C 4094BBA3 F2F5A378 97E2F06F DCFFC509 A07B930A FBE6C3CA
E1FC7FD9 1E69B872 C402E62A A8814C09 quit ! interface Ethernet0 ip address 20.20.20.20
255.255.255.0 ! interface BRI0 description bri to rtp ip address 12.12.12.12 255.255.255.0 no ip
proxy-arp encapsulation ppp no ip mroute-cache bandwidth 128 load-interval 30 dialer idle-
timeout 99999 dialer hold-queue 40 dialer-group 1 isdn spid1 919472417100 4724171 isdn spid2
919472417201 4724172 ppp authentication chap crypto map test ! ip classless ip route 0.0.0.0

0.0.0.0 12.12.12.13 access-list 133 permit ip 20.20.20.0 0.0.0.255 40.40.40.0 0.0.0.255 dialer-list 1 protocol ip permit ! line con 0 exec-timeout 0 0 line vty 0 4 password ww login ! end lab-isdn# ------------------ RSA-sig -------------------------- lab-isdn#**show debug**
Cryptographic Subsystem: Crypto ISAKMP debugging is on Crypto Engine debugging is on Crypto IPSEC debugging is on lab-isdn# lab-isdn# *Mar 21 20:16:50.871: ISAKMP (4): processing SA payload. message ID = 0 *Mar 21 20:16:50.871: ISAKMP (4): Checking ISAKMP transform 1 against priority 65535 policy *Mar 21 20:16:50.875: ISAKMP: encryption DES-CBC *Mar 21 20:16:50.875: ISAKMP: hash SHA *Mar 21 20:16:50.875: ISAKMP: default group 1 *Mar 21 20:16:50.875: ISAKMP: auth RSA sig *Mar 21 20:16:50.879: ISAKMP (4): atts are acceptable. Next payload is 0 *Mar 21 20:16:50.879: Crypto engine 0: generate alg param *Mar 21 20:16:54.070: CRYPTO_ENGINE: Dh phase 1 status: 0 *Mar 21 20:16:54.090: ISAKMP (4): SA is doing RSA signature authentication *Mar 21 20:16:57.343: ISAKMP (4): processing KE payload. message ID = 0 *Mar 21 20:16:57.347: Crypto engine 0: generate alg param *Mar 21 20:17:01.168: ISAKMP (4): processing NONCE payload. message ID = 0 *Mar 21 20:17:01.176: Crypto engine 0: create ISAKMP SKEYID for conn id 4 *Mar 21 20:17:01.188: ISAKMP (4): SKEYID state generated *Mar 21 20:17:07.331: ISAKMP (4): processing ID payload. message ID = 0 *Mar 21 20:17:07.331: ISAKMP (4): processing CERT payload. message ID = 0 *Mar 21 20:17:07.497: ISAKMP (4): cert approved with warning *Mar 21 20:17:07.600: ISAKMP (4): processing SIG payload. message ID = 0 *Mar 21 20:17:07.608: Crypto engine 0: RSA decrypt with public key *Mar 21 20:17:07.759: generate hmac context for conn id 4 *Mar 21 20:17:07.767: ISAKMP (4): SA has been authenticated *Mar 21 20:17:07.775: generate hmac context for conn id 4 *Mar 21 20:17:07.783: Crypto engine 0: RSA encrypt with private key *Mar 21 20:17:08.672: CRYPTO_ENGINE: key process suspended and continued *Mar 21 20:17:08.878: CRYPTO_ENGINE: key process suspended and continued *Mar 21 20:17:09.088: CRYPTO_ENGINE: key process suspended and continued *Mar 21 20:17:09.291: CRYPTO_ENGINE: key process suspended and continued *Mar 21 20:17:09.493: CRYPTO_ENGINE: key process suspended and continued *Mar 21 20:17:09.795: CRYPTO_ENGINE: key process suspended and continued *Mar 21 20:17:10.973: generate hmac context for conn id 4 *Mar 21 20:17:10.981: ISAKMP (4): processing SA payload. message ID = -538880964 *Mar 21 20:17:10.981: ISAKMP (4): Checking IPSec proposal 1 *Mar 21 20:17:10.981: ISAKMP: transform 1, AH_SHA_HMAC *Mar 21 20:17:10.985: ISAKMP: attributes in transform: *Mar 21 20:17:10.985: ISAKMP: encaps is 1 *Mar 21 20:17:10.985: ISAKMP: SA life type in seconds *Mar 21 20:17:10.985: ISAKMP: SA life duration (basic) of 3600 *Mar 21 20:17:10.989: ISAKMP: SA life type in kilobytes *Mar 21 20:17:10.989: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50 0x0 *Mar 21 20:17:10.993: ISAKMP (4): atts are acceptable. *Mar 21 20:17:10.993: ISAKMP (4): Checking IPSec proposal 1 *Mar 21 20:17:10.993: ISAKMP: transform 1, ESP_DES *Mar 21 20:17:10.997: ISAKMP: attributes in transform: *Mar 21 20:17:10.997: ISAKMP: encaps is 1 *Mar 21 20:17:10.997: ISAKMP: SA life type in seconds *Mar 21 20:17:10.997: ISAKMP: SA life duration (basic) of 3600 *Mar 21 20:17:11.001: ISAKMP: SA life type in kilobytes *Mar 21 20:17:11.001: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50 0x0 *Mar 21 20:17:11.001: ISAKMP: HMAC algorithm is SHA *Mar 21 20:17:11.005: ISAKMP (4): atts are acceptable. *Mar 21 20:17:11.005: IPSEC(validate_proposal_request): proposal part #1, (key eng. msg.) dest= 12.12.12.12, SRC= 12.12.12.13, dest_proxy= 20.20.20.0/0.0.0.0/0/0, src_proxy= 40.40.40.0/0.0.0.16/0/0, protocol= AH, transform= ah-sha-hmac , lifedur= 0s and 0kb, spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4 *Mar 21 20:17:11.013: IPSEC(validate_proposal_request): proposal part #2, (key eng. msg.) dest= 12.12.12.12, SRC= 12.12.12.13, dest_proxy= 20.20.20.0/0.0.0.0/0/0, src_proxy= 40.40.40.0/0.0.0.16/0/0, protocol= ESP, transform= esp-des esp-sha-hmac , lifedur= 0s and 0kb, spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4 *Mar 21 20:17:11.021: ISAKMP (4): processing NONCE payload. message ID = -538880964 *Mar 21 20:17:11.021: ISAKMP (4): processing ID payload. message ID = -538880964 *Mar 21 20:17:11.021: ISAKMP (4): processing ID payload. message ID = -538880964 *Mar 21 20:17:11.025: IPSEC(key_engine): got a queue event... *Mar 21 20:17:11.029: IPSEC(spi_response): getting spi 112207019 for SA from 12.12.12.13 to 12.12.12.12 for prot 2 *Mar 21 20:17:11.033: IPSEC(spi_response): getting spi 425268832 for SA from 12.12.12.13 to 12.12.12.12 for prot 3 *Mar 21 20:17:11.279: generate hmac context for conn id 4 *Mar 21 20:17:11.612: generate hmac context for conn id 4 *Mar 21 20:17:11.644: ISAKMP (4): Creating IPSec SAs *Mar 21 20:17:11.644: inbound SA from 12.12.12.13 to 12.12.12.12 (proxy 40.40.40.0 to 20.20.20.0) *Mar 21 20:17:11.648: has spi 112207019 and conn_id 5 and flags 4 *Mar 21 20:17:11.648: lifetime of 3600 seconds *Mar 21 20:17:11.648: lifetime of 4608000 kilobytes *Mar 21 20:17:11.652: outbound SA from 12.12.12.12 to 12.12.12.13 (proxy 20.20.20.0 to 40.40.40.0) *Mar 21 20:17:11.652: has spi 83231845 and conn_id 6 and flags 4 *Mar 21 20:17:11.656: lifetime of 3600 seconds *Mar 21 20:17:11.656: lifetime of 4608000 kilobytes *Mar 21 20:17:11.656: ISAKMP (4): Creating IPSec SAs *Mar 21 20:17:11.656: inbound SA from 12.12.12.13 to 12.12.12.12 (proxy 40.40.40.0 to 20.20.20.0) *Mar 21 20:17:11.660: has spi 425268832 and conn_id 7 and flags 4 *Mar 21 20:17:11.660: lifetime of 3600 seconds *Mar 21 20:17:11.664: lifetime of 4608000 kilobytes *Mar 21 20:17:11.664: outbound SA from 12.12.12.12 to 12.12.12.13 (proxy 20.20.20.0 to 40.40.40.0) *Mar 21 20:17:11.668: has spi 556010247 and conn_id 8 and flags 4 *Mar 21

```
20:17:11.668: lifetime of 3600 seconds *Mar 21 20:17:11.668: lifetime of 4608000 kilobytes *Mar
21 20:17:11.676: IPSEC(key_engine): got a queue event... *Mar 21 20:17:11.676:
IPSEC(initialize_sas): , (key eng. msg.) dest= 12.12.12.12, SRC= 12.12.12.13, dest_proxy=
20.20.20.0/255.255.255.0/0/0, src_proxy= 40.40.40.0/255.255.255.0/0/0, protocol= AH, transform=
ah-sha-hmac , lifedur= 3600s and 4608000kb, spi= 0x6B024AB(112207019), conn_id= 5, keysize= 0,
flags= 0x4 *Mar 21 20:17:11.680: IPSEC(initialize_sas): , (key eng. msg.) SRC= 12.12.12.12,
dest= 12.12.12.13, src_proxy= 20.20.20.0/255.255.255.0/0/0, dest_proxy=
40.40.40.0/255.255.255.0/0/0, protocol= AH, transform= ah-sha-hmac , lifedur= 3600s and
4608000kb, spi= 0x4F60465(83231845), conn_id= 6, keysize= 0, flags= 0x4 *Mar 21 20:17:11.687:
IPSEC(initialize_sas): , (key eng. msg.) dest= 12.12.12.12, SRC= 12.12.12.13, dest_proxy=
20.20.20.0/255.255.255.0/0/0, src_proxy= 40.40.40.0/255.255.255.0/0/0, protocol= ESP, transform=
esp-des esp-sha-hmac , lifedur= 3600s and 4608000kb, spi= 0x19591660(425268832), conn_id= 7,
keysize= 0, flags= 0x4 *Mar 21 20:17:11.691: IPSEC(initialize_sas): , (key eng. msg.) SRC=
12.12.12.12, dest= 12.12.12.13, src_proxy= 20.20.20.0/255.255.255.0/0/0, dest_proxy=
40.40.40.0/255.255.255.0/0/0, protocol= ESP, transform= esp-des esp-sha-hmac , lifedur= 3600s
and 4608000kb, spi= 0x21240B07(556010247), conn_id= 8, keysize= 0, flags= 0x4 *Mar 21
20:17:11.699: IPSEC(create_sa): sa created, (sa) sa_dest= 12.12.12.12, sa_prot= 51, sa_spi=
0x6B024AB(112207019), sa_trans= ah-sha-hmac , sa_conn_id= 5 *Mar 21 20:17:11.703:
IPSEC(create_sa): sa created, (sa) sa_dest= 12.12.12.13, sa_prot= 51, sa_spi=
0x4F60465(83231845), sa_trans= ah-sha-hmac , sa_conn_id= 6 *Mar 21 20:17:11.707:
IPSEC(create_sa): sa created, (sa) sa_dest= 12.12.12.12, sa_prot= 50, sa_spi=
0x19591660(425268832), sa_trans= esp-des esp-sha-hmac , sa_conn_id= 7 *Mar 21 20:17:11.707:
IPSEC(create_sa): sa created, (sa) sa_dest= 12.12.12.13, sa_prot= 50, sa_spi=
0x21240B07(556010247), sa_trans= esp-des esp-sha-hmac , sa_conn_id= 8 *Mar 21 20:18:06.767:
ISADB: reaper checking SA, conn_id = 4 lab-isdn#
```

Dépannage pour IPSec et ISAKMP

Il est généralement le meilleur de commencer chaque session de dépannage en recueillant des informations utilisant les commandes suivantes. Un astérisque (*) indique une commande particulièrement utile. Veuillez voir également le dépannage de sécurité IP - Comprenant et utilisant des commandes de débogage pour information les informations complémentaires.

Certaines commandes **show** sont prises en charge par l'Output Interpreter Tool (clients enregistrés uniquement), qui vous permet de voir une analyse de la sortie de la commande show.

Remarque: Avant d'exécuter les commandes **debug**, référez-vous à la section **Informations importantes sur les commandes Debug**.

Commandes	
transport de PKI de debug crypto	* debug crypto ipsec
* debug crypto isakmp	clé de debug crypto
sess de debug crypto	debug crypto engine
connexions de show crypto engine actives	relâcher-paquet de connexions de show crypto engine
configuration de show crypto engine	* show crypto ca certificat
* show crypto key mypubkey rsa	* show crypto key pubkey-chain rsa
show crypto isakmp policy	show crypto isakmp sa
show crypto ipsec sa	affichez la crypto clé de session d'ipsec

affichez la crypto transformer-proposition d'ipsec	affichez l'interface bri 0 de crypto map
affichez le test de balise de crypto map	effacez le crypto id de <connection de connexion de SA>
* clear crypto isakmp	* clear crypto sa
compteurs de clear crypto sa	carte de clear crypto sa
pair de clear crypto sa	spi de clear crypto sa
compteurs de clear crypto sa	

La sortie témoin de certaines de ces commandes est affichée ci-dessous.

```
wan2511#show crypto engine connections active ID Interface IP-Address State Algorithm Encrypt
Decrypt 9 Serial0 20.20.20.21 set HMAC_SHA 0 240 10 Serial0 20.20.20.21 set HMAC_SHA 240 0
wan2511#show crypto engine connections dropped-packet Interface IP-Address Drop Count
wan2511#show crypto engine configuration slot: 0 engine name: unknown engine type: software
serial number: 01496536 platform: rp crypto engine crypto lib version: 10.0.0 Encryption Process
Info: input queue top: 140 input queue bot: 140 input queue count: 0 wan2511#show crypto key
mypubkey rsa % Key pair was generated at: 00:09:04 UTC Mar 1 1993 Key name: wan2511.cisco.com
Usage: General Purpose Key Key Data: 305C300D 06092A86 4886F70D 01010105 00034B00 30480241
00E9007B E5CD7DC8 6E1C0423 92044254 92C972AD 0CCE9796 86797EAA B6C4EFF0 0F0A5378 6AFAE43B
3A2BD92F 98039DAC 08741E82 5D9053C4 D9CFABC1 AB54E0E2 BB020301 0001 wan2511#show crypto key
pubkey-chain rsa wan2511# wan2511#show crypto isakmp policy Protection suite of priority 1
encryption algorithm: DES - Data Encryption Standard (56 bit keys). hash algorithm: Secure Hash
Standard authentication method: Pre-Shared Key Diffie-Hellman group: #2 (1024 bit) lifetime: 240
seconds, no volume limit Default protection suite encryption algorithm: DES - Data Encryption
Standard (56 bit keys). hash algorithm: Secure Hash Standard authentication method: Rivest-
Shamir-Adleman Signature Diffie-Hellman group: #1 (768 bit) lifetime: 86400 seconds, no volume
limit wan2511#show crypto isakmp sa dst src state conn-id slot 20.20.20.21 20.20.20.20 QM_IDLE 7
0 wan2511# wan2511#show crypto ipsec sa interface: Serial0 Crypto map tag: test, local addr.
20.20.20.21 local ident (addr/mask/prot/port): (50.50.50.0/255.255.255.0/0/0) remote ident
(addr/mask/prot/port): (60.60.60.0/255.255.255.0/0/0) current_peer: 20.20.20.20 PERMIT,
flags={origin_is_acl,ident_is_ipsec,} #pkts encaps: 320, #pkts encrypt: 320, #pkts digest 320
#pkts decaps: 320, #pkts decrypt: 320, #pkts verify 320 #send errors 0, #recv errors 0 local
crypto endpt.: 20.20.20.21, remote crypto endpt.: 20.20.20.20 path mtu 1500, media mtu 1500
current outbound spi: 6625CD inbound esp sas: spi: 0x1925112F(421859631) transform: esp-des esp-
sha-hmac , in use settings ={Tunnel, } slot: 0, conn id: 11, crypto map: test sa timing:
remaining key lifetime (k/sec): (4607971/3354) IV size: 8 bytes replay detection support: Y
inbound ah sas: spi: 0x12050DD2(302321106) transform: ah-sha-hmac , in use settings ={Tunnel, }
slot: 0, conn id: 9, crypto map: test sa timing: remaining key lifetime (k/sec): (4607958/3354)
replay detection support: Y outbound esp sas: spi: 0x3262313(52830995) transform: esp-des esp-
sha-hmac , in use settings ={Tunnel, } slot: 0, conn id: 12, crypto map: test sa timing:
remaining key lifetime (k/sec): (4607971/3354) IV size: 8 bytes replay detection support: Y
outbound ah sas: spi: 0x6625CD(6694349) transform: ah-sha-hmac , in use settings ={Tunnel, }
slot: 0, conn id: 10, crypto map: test sa timing: remaining key lifetime (k/sec): (4607958/3354)
replay detection support: Y wan2511#show crypto ipsec session-key Session key lifetime: 4608000
kilobytes/3600 seconds wan2511#show crypto ipsec transform-proposal Transform proposal auth2: {
ah-sha-hmac } supported settings = { Tunnel, }, default settings = { Tunnel, }, will negotiate =
{ Tunnel, }, { esp-des esp-sha-hmac } supported settings = { Tunnel, }, default settings = {
Tunnel, }, will negotiate = { Tunnel, }, wan2511#show crypto map interface serial 0 Crypto Map
"test" 10 ipsec-isakmp Peer = 20.20.20.20 Extended IP access list 133 access-list 133 permit ip
source: addr = 50.50.50.0/0.0.0.255 dest: addr = 60.60.60.0/0.0.0.255 Current peer: 20.20.20.20
Session key lifetime: 4608000 kilobytes/3600 seconds PFS (Y/N): N Transform proposals={ auth2, }
wan2511#show crypto map tag test Crypto Map "test" 10 ipsec-isakmp Peer = 20.20.20.20 Extended
IP access list 133 access-list 133 permit ip source: addr = 50.50.50.0/0.0.0.255 dest: addr =
60.60.60.0/0.0.0.255 Current peer: 20.20.20.20 Session key lifetime: 4608000 kilobytes/3600
```

seconds PFS (Y/N): N Transform proposals={ auth2, } wan2511# ---------- lab-isdn1#**show crypto engine connections active** ID Interface IP-Address State Algorithm Encrypt Decrypt 5 BRI0 12.12.12.13 set HMAC_SHA 0 89 6 BRI0 12.12.12.13 set HMAC_SHA 89 0 lab-isdn1#**show crypto engine connections dropped-packet** Interface IP-Address Drop Count BRI0 12.12.12.13 4 lab-isdn1#**show crypto engine configuration** slot: 0 engine name: unknown engine type: software serial number: 05679987 platform: rp crypto engine crypto lib version: 10.0.0 Encryption Process Info: input queue top: 243 input queue bot: 243 input queue count: 0 lab-isdn1#**show crypto ca cert** Certificate Subject Name Name: lab-isdn1.cisco.com Serial Number: 05679987 Status: Available Certificate Serial Number: 3E1ED472BDA2CE0163FB6B0B004E5EEE Key Usage: Encryption CA Certificate Status: Available Certificate Serial Number: 3051DF7169BEE31B821DFE4B3A338E5F Key Usage: Not Set Certificate Subject Name Name: lab-isdn1.cisco.com Serial Number: 05679987 Status: Available Certificate Serial Number: 503968D890F7D409475B7280162754D2 Key Usage: Signature lab-isdn1#**show crypto key mypubkey rsa** % Key pair was generated at: 03:10:23 UTC Mar 21 1993 Key name: lab-isdn1.cisco.com Usage: Signature Key Key Data: 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00BECE2D 8CB32E6B 090ADE0D 46AF8D4A 1F378500 3435D0C7 293BF915 180C9E4C F81A6A43 AEE4F046 87B8E285 9D33D5CE 042E5DDE A63DA54A 312AD425 5A756014 CB020301 0001 % Key pair was generated at: 03:11:17 UTC Mar 21 1993 Key name: lab-isdn1.cisco.com Usage: Encryption Key Key Data: 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00D2D125 FFBBFC6E 5693CB43 855473C1 65BC7CCA F645C35B ED554BAA 0B119AFA 6F0853F5 745E0B84 922E39B5 FA84C4DD 05C19AA6 25818439 5C6CBC7F A4614F61 77020301 0001 lab-isdn1#**show crypto key pubkey-chain rsa** Key name: Cisco SystemsDevtestCISCOCA-ULTRA Key serial number: C7040262 Key usage: signatures only Key source: certificate Key data: 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00C1B69D 7BF634E4 EE28A84E 0DC6FCA4 DEA804D8 9E50C5EB E86239D5 1890D0D4 B732678B DBF28080 1430E5E5 6E7CC126 E2DDDBE9 695ADF8E 5BA7E67B AE872937 53020301 0001 Key name: lab-isdn.cisco.com Key address: 171.68.117.189 Key serial number: 05679919 Key usage: general purpose Key source: certificate Key data: 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00D771AD 5672B487 A0195ECD 19546F91 9A3A6270 102E5A9F F4DC7A60 8480FB27 A1817153 35F4399D 3E577F72 B323BF06 20AB60C3 71CF4389 BA4FC60E E6EA21E0 63020301 0001 lab-isdn1#**show crypto isakmp policy** Default protection suite encryption algorithm: DES - Data Encryption Standard (56 bit keys). hash algorithm: Secure Hash Standard authentication method: Rivest-Shamir-Adleman Signature Diffie-Hellman group: #1 (768 bit) lifetime: 86400 seconds, no volume limit lab-isdn1#**show crypto isakmp sa** dst src state conn-id slot 12.12.12.12 12.12.12.13 QM_IDLE 4 0 lab-isdn1#**show crypto ipsec sa** interface: BRI0 Crypto map tag: test, local addr. 12.12.12.13 local ident (addr/mask/prot/port): (40.40.40.0/255.255.255.0/0/0) remote ident (addr/mask/prot/port): (20.20.20.0/255.255.255.0/0/0) current_peer: 12.12.12.12 PERMIT, flags={origin_is_acl,ident_is_ipsec,} #pkts encaps: 89, #pkts encrypt: 89, #pkts digest 89 #pkts decaps: 89, #pkts decrypt: 89, #pkts verify 89 #send errors 11, #recv errors 0 local crypto endpt.: 12.12.12.13, remote crypto endpt.: 12.12.12.12 path mtu 1500, media mtu 1500 current outbound spi: 6B024AB inbound esp sas: spi: 0x21240B07(556010247) transform: esp-des esp-sha-hmac , in use settings ={Tunnel, } slot: 0, conn id: 7, crypto map: test sa timing: remaining key lifetime (k/sec): (4607989/3062) IV size: 8 bytes replay detection support: Y inbound ah sas: spi: 0x4F60465(83231845) transform: ah-sha-hmac , in use settings ={Tunnel, } slot: 0, conn id: 5, crypto map: test sa timing: remaining key lifetime (k/sec): (4607984/3062) replay detection support: Y outbound esp sas: spi: 0x19591660(425268832) transform: esp-des esp-sha-hmac , in use settings ={Tunnel, } slot: 0, conn id: 8, crypto map: test sa timing: remaining key lifetime (k/sec): (4607989/3062) IV size: 8 bytes replay detection support: Y outbound ah sas: spi: 0x6B024AB(112207019) transform: ah-sha-hmac , in use settings ={Tunnel, } slot: 0, conn id: 6, crypto map: test sa timing: remaining key lifetime (k/sec): (4607984/3062) replay detection support: Y lab-isdn1#**show crypto ipsec session-key** Session key lifetime: 4608000 kilobytes/3600 seconds lab-isdn1#**show crypto ipsec transform-proposal** Transform proposal mypolicy: { ah-sha-hmac } supported settings = { Tunnel, }, default settings = { Tunnel, }, will negotiate = { Tunnel, }, { esp-des esp-sha-hmac } supported settings = { Tunnel, }, default settings = { Tunnel, }, will negotiate = { Tunnel, }, lab-isdn1#**show crypto map interface bri 0** Crypto Map "test" 10 ipsec-isakmp Peer = 12.12.12.12 Extended IP access list 144 access-list 144 permit ip source: addr = 40.40.40.0/0.0.0.255 dest: addr = 20.20.20.0/0.0.0.255 Current peer: 12.12.12.12 Session key lifetime: 4608000 kilobytes/3600 seconds PFS (Y/N): N Transform proposals={ mypolicy, } lab-isdn1#**show crypto map tag test** Crypto Map "test" 10 ipsec-isakmp Peer = 12.12.12.12 Extended IP access list 144 access-list 144 permit ip source: addr = 40.40.40.0/0.0.0.255 dest: addr = 20.20.20.0/0.0.0.255 Current peer: 12.12.12.12 Session key lifetime: 4608000 kilobytes/3600 seconds PFS (Y/N): N Transform proposals={ mypolicy, } lab-isdn1# ---------- lab-isdn1#**clear crypto isakmp** lab-isdn1# *Mar 21 20:58:34.503: ISADB: reaper checking SA, conn_id = 4 DELETE IT! *Mar 21 20:58:34.507: generate hmac context for conn id 4 *Mar 21 20:58:34.519: CRYPTO(epa_release_crypto_conn_entry): released conn 4 lab-isdn1# lab-isdn1#**clear crypto sa** lab-isdn1# *Mar 21 20:58:42.495: IPSEC(delete_sa):

```
deleting SA, (sa) sa_dest= 12.12.12.13, sa_prot= 51, sa_spi= 0x4F60465(83231845), sa_trans= ah-
sha-hmac , sa_conn_id= 5 *Mar 21 20:58:42.499: CRYPTO(epa_release_crypto_conn_entry): released
conn 5 *Mar 21 20:58:42.499: IPSEC(delete_sa): deleting SA, (sa) sa_dest= 12.12.12.12, sa_prot=
51, sa_spi= 0x6B024AB(112207019), sa_trans= ah-sha-hmac , sa_conn_id= 6 *Mar 21 20:58:42.503:
CRYPTO(epa_release_crypto_conn_entry): released conn 6 *Mar 21 20:58:42.503: IPSEC(delete_sa):
deleting SA, (sa) sa_dest= 12.12.12.13, sa_prot= 50, sa_spi= 0x21240B07(556010247), sa_trans=
esp-des esp-sha-hmac , sa_conn_id= 7 *Mar 21 20:58:42.507:
CRYPTO(epa_release_crypto_conn_entry): released conn 7 *Mar 21 20:58:42.507: IPSEC(delete_sa):
deleting SA, (sa) sa_dest= 12.12.12.12, sa_prot= 50, sa_spi= 0x19591660(425268832), sa_trans=
esp-des esp-sha-hmac , sa_conn_id= 8 *Mar 21 20:58:42.511:
CRYPTO(epa_release_crypto_conn_entry): released conn 8 lab-isdn1#
```

Informations connexes

- Configuration et dépannage du chiffrement de couche réseau Cisco Contexte – 1re partie
- PAP DES 46-2 au National Institute of Standards and Technology (NIST)
- PAP 186 de DSS au National Institute of Standards and Technology (NIST)
- Les forums aux questions des laboratoires RSA au sujet du chiffrement d'aujourd'hui
- Normes de sécurité IETF
- Configurer le protocole de sécurité IKE
- Sécurité des réseaux de configuration d'IPSec
- Page d'assistance IPsec
- Support technique - Cisco Systems