Liberté • Égalité • Fraternité
RÉPUBLIQUE FRANÇAISE

MINISTÈRE DE LA DÉFENSE

DGA

Bruz, le 11/06/2007

RT/07/101422/CELAR/ CELAR/SSI/SSY/EA/51700469/NP

Version : 2.0

Ce document comporte 46 pages

DELEGATION GENERALE POUR L'ARMEMENT

DIRECTION DES EXPERTISES TECHNIQUES

Centre d'Electronique de l'Armement

Affaire / prestations : Analyse de produits civils

Rapport technique

Recommandations pour la sécurisation des routeurs CISCO

Noms et fonctions		Date	Visa
Rédigé par : **Nom : David Boucart**	**Département: SSI/SSY**		
Vérifié par : **Nom : Alain Besson**	**Département: SSI/SSY**		
Vérifié par : **Nom : Sébastien Gay**	**Département: SSI/SSY**		
Approuvé par : **Nom : Guy Monnerais**	**Chef du Département SSI/SSY**		

CELAR – Boîte Postale 7 – 35998 Rennes Armées
CELAR – BP 5 7419 – 35174 Bruz Cedex
Téléphone : + 33 (0) 2 99 42 90 11 - Télécopie : + 33 (0) 2 99 42 91 01

FICHE D'IDENTIFICATION ET DE DIFFUSION DES DOCUMENTS

ORGANISME ÉMETTEUR	Niveaux de protection des documents	
CENTRE D'ÉLECTRONIQUE DE L'ARMEMENT	DÉFENSE	SPÉCIFIQUE
35170 - BRUZ	NP	NC
	Date de déclassification : ../../.... ☐ Automatique ☐ Sur ordre	

Document non protégé, veuillez cocher une des deux cases suivantes : ☐ à diffusion limitée ☐ sans restriction de diffusion

Grille de diffusion du document et du signalement

Diffusion Destinataires	par l'émetteur exclusivement	par le CEDOCAR avec l'accord de l'émetteur	par le CEDOCAR sans accord de l'émetteur	diffusion du signalement par le CEDOCAR
DGA			X	
Défense + SGDN			X	
Organismes agréés			X	

Titre du document : Recommandations pour la sécurisation des routeurs CISCO

Auteur(s) : DAVID BOUCART

Numéro d'origine du document	Numéro du contrat	Service de l'État qui suit l'exécution du contrat	Date de publication

Titre déclassifié : (éventuellement)

Résumé d'auteur : Le document a pour objet de recenser les différents mécanismes offerts aux administrateurs pour améliorer le niveau de sécurité des routeurs Cisco et de proposer des recommandations de paramétrage.

Notions d'indexage: Sécurité Cisco

HISTORIQUE

Version	Date	Nature
1.0	11/07/2000	• Création du document
1.1	24/10/2000	• Prise en compte des remarques internes
1.2	13/11/2000	• Refonte des annexes
1.3	07/02/2002	• Correction du contrôle d'accès sur vty, aux et console • Précision sur le cache de routage • Listes d'accès réflexives • Transmission des broadcasts • Désactivation de SNMP
2.0	11/06/2007	• Restructuration du document • Authentification • Journalisation • Routage • Connexions sécurisées

DESTINATAIRES

Nom	Société	Nombre ou numéros des exemplaires

SOMMAIRE

1 Objet

Le présent document a pour objet de fournir des recommandations aux administrateurs et architectes réseau concernant la sécurité des routeurs CISCO.

Il décrit quelles sont les opérations à réaliser sur le routeur afin d'augmenter de façon sensible la sécurité du réseau et du routeur lui-même.

2 Avant-propos

2.1 Pré-requis

Ce guide s'adresse à des administrateurs réseaux expérimentés dans la configuration de routeurs CISCO.

Comme tout système d'exploitation, l'IOS possède des bugs et des failles. Il faut donc toujours veiller à utiliser une version récente du système. Ce guide a été conçu pour les versions 12.0 et suivantes (la plupart des commandes décrites peuvent cependant s'appliquer à des versions plus anciennes).

La saisie de commandes implicites ne laisse pas de traces dans les fichiers de configuration. Dans ce cas, il est peut être utile d'utiliser les commandes de diagnostics pour vérifier la prise en compte d'une commande.

2.2 Conventions utilisées dans le document

Chaque mécanisme fait l'objet :

- d'une recommandation,

R0. Recommandation

- d'une explication en texte libre, sur un ou plusieurs chapitres,
- d'une méthode ou commande.

```
M0.
     Router(config)# access-list num-acl {permit|deny}
                    {protocole}
                    {source masque-src} {destination masque-dst}
                    [options-spécifiques] [log]
```

`Router`	nom du routeur
`Router>`	mode utilisateur

`Router#`	mode privilégié
`Router(config)#`	indique le mode de configuration
`Router(config-if)#`	indique le mode de configuration sur une interface
`gras`	mot-clé de l'IOS
`italique`	valeur à entrer par l'utilisateur
`{commande}`	groupe logique
`cmd1 | cmd2`	cmd1 ou cmd2
`[opt]`	optionnel

2.3 Limitations

D'autres possibilités de sécurisation existent et ne sont pas traitées dans ce document, en particulier :

- Chiffrement IPSEC
- Routage IPX, Appletalk, etc.
- Filtrage IPX, Appletalk…

3 Protection physique du routeur

R1. Protéger les accès physiques au routeur

Il est possible via le port console de réinitialiser les mots de passe configurés sur le routeur. Cette procédure permet de récupérer un accès au routeur en cas d'oubli des mots de passe.

Il est par conséquent conseillé d'assurer la sécurité physique du routeur et de sa console :

- au minimum, par la fermeture des locaux techniques,
- éventuellement, par la surveillance de ces locaux (vidéo-surveillance, détection d'intrusion…)

R2. Garantir des conditions de fonctionnement optimales

D'autre part, il est important de garantir des conditions de fonctionnement optimales du routeur afin de s'assurer de sa disponibilité. Il est conseillé :

- de climatiser les locaux,
- de surveiller le taux d'humidité,
- de disposer d'alimentations électriques secourues.
- de disposer d'un routeur identique en secours

- de disposer d'administrateurs formés.

4 Protection logique du routeur

R3. Sauvegarder les configurations

Afin de permettre une restauration d'une configuration en cas de problème (effacement de la mémoire flash, changement du routeur), il est recommandé de sauvegarder les configurations du routeur, soit en utilisant un serveur TFTP, soit par la capture du résultat des commandes de visualisation. Le but étant d'avoir la configuration dans un fichier informatique ou sur papier.

M1.
```
Router#copy running-config tftp
Router#copy startup-config tftp
```

R4. Contrôler régulièrement l'intégrité de la configuration du routeur

Cette sauvegarde permet également de comparer régulièrement la configuration active avec une configuration de référence afin de détecter le cas échéant une modification accidentelle.

5 Configuration du routeur

5.1 Désactivation des interfaces inutilisées

R5. Désactiver les interfaces inutilisées

Par défaut, la plupart des versions d'IOS désactivent les interfaces lors de la génération du fichier de configuration. Dans le cas contraire, il est possible de désactiver manuellement chaque interface inutilisée par la commande :

M2.
```
Router(config)#interface nom_de_l'interface
Router(config-if)#shutdown
Router(config-if)#exit
```

ATTENTION : cette recommandation interdit l'utilisation d'une interface par simple raccordement sur cette dernière. Toutefois, elle implique un accès physique au routeur, et par conséquent est liée à la recommandation R1.

5.2 Services TCP/IP

Les routeurs CISCO peuvent offrir un certain nombre de services TCP/IP qui ne sont pas indispensables. Certains services sont configurés implicitement. Il est par conséquent recommandé de les désactiver.

5.2.1 Services TCP/IP simples

R6. Désactiver les services TCP/IP simples

Les services TCP/IP simples sont des services internes, appelés *small servers*, tels *echo* (ports TCP et UDP numéro 7), *discard* (ports TCP et UDP numéro 9), *chargen* (ports TCP et UDP numéro 19), etc.. Selon la version de l'IOS, ils peuvent être activés par défaut.

Il est possible pour des utilisateurs mal intentionnés d'utiliser ces services afin de provoquer un déni de service.

Il est recommandé de désactiver ces services avec les commandes :

```
M3.
     Router(config)#no service tcp-small-servers
     Router(config)#no service udp-small-servers
```

Un balayage des ports TCP ouverts sur l'adresse du routeur permet de s'assurer de l'absence de ces services.

5.2.2 Service finger

R7. Désactiver le service finger

Le service finger permet de lister les utilisateurs connectés sur le routeur. Ce service peut donc fournir des informations sur les comptes utilisateurs ou les habitudes de travail. Il est recommandé de le désactiver.

```
M4.
     Router(config)#no service finger
```

5.2.3 Service config

R8. Désactiver le service config

Les routeurs Cisco peuvent charger leur fichier de configuration sur le réseau à l'aide du protocole TFTP. Ils émettent pour cela des requêtes en diffusion sur chaque interface.

Ce mécanisme peut être contourné pour télécharger une fausse configuration sur le routeur (il n'y a pas d'authentification en TFTP). Il est donc recommandé de désactiver le service config à l'aide de la commande suivante:

M5.

```
Router(config)#no service config
```

Remarque : la désactivation du service config n'interdit pas l'utilisation du protocole TFTP pour le transfert des fichiers de configuration.

5.2.4 Cisco Discovery Protocol

R9. Désactiver le protocole CDP

Le protocole CDP (Cisco Discovery Protocol) est un protocole propriétaire Cisco qui permet aux routeurs de s'échanger des informations. Il n'est pas indispensable pour le fonctionnement du routeur.

L'écoute du trafic CDP sur un réseau permet d'obtenir les noms des routeurs, de leurs interfaces, les versions des matériels et des IOS.

M6.

- Pour désactiver globalement le protocole pour toutes les interfaces :

```
Router(config)#no cdp run
```

- Pour désactiver le protocole sur une interface :

```
Router(config)#interface nom_de_l'interface
Router(config-if)#no cdp enable
```

5.2.5 Serveur HTTP interne

R10. Désactiver le serveur HTTP

Le serveur HTTP permet la gestion du routeur à partir d'une interface HTML. Il est très rudimentaire et ne présente que peu d'intérêts.

Ce service s'avère, dans certaines versions, particulièrement dangereux (voir Avis du Cert-A N° 2001-071"Vulnérabilité du serveur HTTP IOS"). Il est donc fortement recommandé de le désactiver.

M7.

```
Router(config)#no ip http server
```

Si néanmoins, l'administrateur tient absolument à utiliser la gestion WEB, il faut limiter les accès à l'aide d'une liste de contrôle d'accès standard (ACL), aux seuls postes d'administration (cf. chapitre 8. pour la définition des listes de contrôle d'accès) :

M8.

- Définir l'autorisation d'accès pour le poste possédant l'adresse *adresse-ip*

```
Router(config)#access-list acl-http permit ip host adresse-ip
```

- Activer la liste de contrôle d'accès pour le serveur HTTP :

```
Router(config)#ip http access-class acl-http
```

5.2.6 Services BOOTP et DHCP

R11. Désactiver les serveurs BOOTP et DHCP

BOOTP est un protocole utilisé par certains matériels pour charger leur système d'exploitation au travers d'un réseau. Les routeurs Cisco sont capable de fonctionner comme des serveurs BOOTP, principalement pour d'autres matériels Cisco. Il est conseillé de désactiver ce service.

DHCP est l'évolution du protocole BOOTP. Il permet l'allocation dynamique des adresses IP et des paramètres de configuration IP. Les routeurs Cisco peuvent fonctionner en tant que serveur DHCP pour les réseaux auxquels ils sont connectés. Il est conseillé de désactiver ce service si le protocole DHCP n'est pas mis en œuvre, ou si un autre serveur DHCP remplit ce rôle.

M9.

```
Router(config)#no ip bootp server
Router(config)#no ip dhcp server
```

5.2.7 Protocole SNMP

R12. Désactiver l'agent SNMPv1 et v2c

SNMP est un protocole de supervision de réseau qui utilise un mécanisme de sécurité assez faible (nom de communauté circulant en clair). Dans le cas où la supervision à l'aide du protocole SNMP ne serait pas mise en œuvre, il faut désactiver l'agent SNMP présent sur le routeur.

ATTENTION : pour désactiver correctement SNMP, il faut retirer de la configuration toutes les lignes concernées. La seule commande **`no snmp-server`** n'efface pas les noms de communautés (même s'ils n'apparaissent plus dans le fichier de configuration !).

M10.

Pour désactiver complètement l'agent :

- retirer les noms de communautés

```
Router(config)# no snmp-server community communaute [RO|RW]
```

- désactiver les traps

```
Router(config)# no snmp-server enable traps
```

- désactiver l'agent

```
Router(config)# no snmp-server
```

La recommandation suivante remplace les recommandations R12 lorsque le routeur et la station de supervision supportent SNMPv3.

R12bis. Si SNMP est nécessaire, utiliser de préférence la version SNMPv3 avec authentification MD5 ou SHA

La version 3 du protocole SNMP offre un mécanisme d'authentification forte, complété éventuellement d'un mécanisme de chiffrement DES. Les versions d'IOS récentes permettent d'activer l'authentification forte.

M11.

- Définir le nom de l'agent :

```
Router(config)#snmp-server engineId nom_agent
```

- Définir un groupe utilisant un modèle de sécurité avec authentification seule:

```
Router(config)#snmp-server group nom_groupe v3 auth
```

- Créer un compte utilisateur avec un mot de passe (de plus de 8 caractères) et une authentification md5:

```
Router (config)#snmp-server user nom_utilisateur group
nom_groupe v3 auth md5 mot_de_passe
```

ATTENTION : la définition des groupes et des utilisateurs pour SNMPv3 ne supprime pas le mode d'authentification de SNMPv1 et v2c. Il faut donc pas oublier de supprimer les noms de communauté SNMPv1 et v2c.

La recommandation suivante ne s'applique que la supervision SNMP est indispensable et que la version SNMP v3 n'est pas supportée.

R12ter. A défaut, sécuriser les protocoles SNMPv1 ou v2c :

- **changer les noms de communautés en valeurs non triviales**
- **limiter les accès en lecture seule**
- **restreindre les accès par une ACL standard aux seuls postes de supervision SNMP**

Les noms de communautés utilisés par défaut sont connus, la précaution minimale consiste donc à les changer. Si l'écoute du trafic réseau n'est pas possible, ces noms de communautés ne seront pas connus.

Le protocole SNMP autorise la consultation et la modification de certains paramètres. La limitation en lecture seule autorisera la supervision sans permettre pour autant la modification de la configuration de l'équipement.

Enfin, l'utilisation d'une liste de contrôle d'accès permet de restreindre l'accès à l'agent SNMP aux seules stations de supervision autorisées.

M12.

- Définir l'autorisation d'accès pour le poste possédant l'adresse *adresse-ip* :

```
Router(config)#access-list acl-snmp permit ip host adresse-ip
```

- Changer le nom de communauté, limiter l'accès en lecture et activer la liste de contrôle d'accès :

```
Router(config)# snmp-server community communaute RO acl-snmp
```

5.3 Protocoles particuliers

5.3.1 Broadcasts dirigés

R13. Interdire la propagation des broadcasts dirigés

Les broadcasts dirigés sont des datagrammes envoyés vers toutes les stations d'un réseau, même distant. Cette technique est utilisée dans les attaques de type "Smurf" (attaque en déni de service). Il est possible de bloquer la propagation de ce type de datagramme au niveau des interfaces du routeur.

ATTENTION : cette commande ne bloque la propagation des broadcasts dirigés qu'au niveau de l'interface reliée au réseau concerné par le broadcast. Elle n'agit donc que sur le dernier routeur.

M13.

- Appliquer les commandes suivantes sur chaque interface :

```
Router(config)#interface nom_de_l'interface
Router(config-if)#no ip directed-broadcast
```

5.3.2 Routage par la source

R14. Interdire le routage par la source

Le routage par la source (Source-routing) est une option du protocole IP qui permet de spécifier le chemin que doit prendre le datagramme pour accéder à sa destination, indépendamment des tables de routages des routeurs traversés.

L'utilisation de cette option permet de contourner la politique de routage définie par l'administrateur du réseau. Il est donc recommandé d'interdire le routage par la source :

M14.

```
Router(config)#no ip source-route
```

5.3.3 Messages ICMP

R15. Désactiver l'émission des messages ICMP d'information

Le protocole ICMP utilise plusieurs messages d'information. Par exemple, le message "redirect" (redirection ICMP) permet d'informer un hôte du réseau d'une meilleure route pour atteindre une destination. Le message "Host-unreachable" informe que le destinataire n'est pas joignable, ce

qui peut indiquer la présence d'un filtrage. Enfin, le message ICMP "Mask reply" fournit le masque de sous-réseau de l'équipement contacté.

Ces messages fournissent par conséquent des informations sur l'architecture du réseau. Il est recommandé de désactiver leur émission par le routeur.

M15.

- Appliquer les commandes suivantes sur chaque interface :

```
Router(config)#interface nom_de_l'interface
Router(config-if)#no ip redirects
Router(config-if)#no ip unreachables
Router(config-if)#no ip mask-reply
```

5.3.4 Relais ARP

R16. Désactiver la fonction Relais ARP

Un routeur Cisco peut se comporter comme un relais ARP (Proxy-ARP) et répondre à des requêtes ARP concernant des hôtes situés sur un autre segment. Ce mécanisme était surtout utile pour des clients ne disposant pas de masque de sous-réseau.

Ce mécanisme n'est plus utilisé actuellement aussi est-il préférable de le désactiver sur chacune des interfaces.

M16.

- Appliquer les commandes suivantes sur chaque interface :

```
Router(config)#interface nom_de_l'interface
Router(config-if)#no ip proxy-arp
```

5.3.5 Cache de routage

R17. Sur les versions d'IOS antérieures à la 12.x, désactiver le mécanisme de cache de routage

Le mécanisme de cache de routage permet d'améliorer les performances grâce à la commutation rapide des paquets.

Sur les anciennes versions d'IOS (8.x et 9.x), le mécanisme de cache était vulnérable. Dans ce cas, il est préférable de le désactiver.

M17.

- Appliquer les commandes suivantes sur chaque interface :

```
Router(config)#interface nom_de_l'interface
Router(config-if)#no ip route-cache
Router(config-if)#no ip mroute-cache
```

Sur les versions d'IOS 12 et ultérieurs, le mécanisme de cache est remplacé par le CEF (Cisco Express Forwarding). Ce dernier est nécessaire à la mise en œuvre de l'Unicast Reverse Path Forwarding (8.4)

5.3.6 Sous-réseau « zéro »

R18. Interdire l'utilisation du sous-réseau « zéro »

L'utilisation du sous-réseau « zéro », le premier d'un adressage par sous-réseau, est fortement déconseillée dans les RFC. Par exemple, le réseau avec une adresse en 14.2.0.0/24 dispose d'une adresse de sous-réseau à 0 au niveau du troisième octet. Par défaut, les routeurs Cisco rejettent les paquets de cette sorte, mais il est recommandé d'interdire explicitement ce mécanisme.

M18.

```
Router(config)#no ip subnet-zero
```

5.3.7 Protocole de maintenance

R19. Désactiver le protocole de maintenance

Le protocole de maintenance permet d'administrer le routeur à distance (configuration, chargement du logiciel système au démarrage, effacement de la mémoire flash...). Il est préférable de le désactiver.

M19.

```
Router(config)#no mop enabled
```

6 Contrôle d'accès

6.1 Bannière

R20. Afficher une message d'avertissement

Il est parfois utile d'afficher un message lors de la connexion sur le routeur pour informer des dispositions légales.

M20.

```
Router(config)# banner <délimiteur> Message <délimiteur>
```

<délimiteur> peut être n'importe quel caractère (qui n'apparaît pas dans le message).

ATTENTION : il faut éviter de donner des informations sur le routeur ou l'architecture du réseau dans la bannière !

6.2 Gestion des mots de passe

Il est impératif de sécuriser les accès au routeur en mode non privilégié par le port console, le port auxiliaire et par telnet, puis l'accès au mode privilégié (enable). Par défaut, aucun mot de passe n'est défini. Il est recommandé de choisir des mots de passe différents pour chaque accès.

R21. Chiffrer les mots de passe stockés dans le fichier de configuration

Les mots de passe sont stockés dans les fichiers de configuration de trois façons :

- en clair
- chiffré en type 7 (propriétaire Cisco)
- chiffré en type 5 (MD5)

Par défaut, les mots de passe utilisés avec les commandes enable password, username et password (line) sont stockés en clair.

Il est possible de demander un chiffrement de type 7 de ces mots de passe.

M21.

```
Router(config)#service password-encryption
```

ATTENTION : cet algorithme est réversible. Par conséquent, il est très facile de retrouver les mots de passe clairs à partir de leur cryptogramme à l'aide d'outils disponibles sur Internet. .

Le mot de passe pour l'accès privilégié est crypté en type 5 (hash MD5). Cet algorithme est irréversible. La seule solution pour retrouver le mot de passe en clair consiste à tester toutes les combinaisons de caractères. Il est recommandé de définir un mot de passe pour ce type d'accès :

```
M22.
     Router(config)#enable secret pwdenable
```

Remarque : cette commande remplace et est prioritaire sur l'ancienne commande **enable password**.

R22. Utiliser des mots de passe forts

La gestion des mots de passe nécessite quelques précautions :

- Ne pas utiliser des mots de passe triviaux tel que le nom du routeur...
- Préférer des mots de passe complexes, de plus de huit caractères et combinant plusieurs types de caractères.
- Utiliser la commande **service password-encryption**
- Ne pas laisser des traces des fichiers de configuration accessibles à tous
- Ne pas utiliser le même mot de passe pour l'accès en telnet ou console et l'accès privilégié.
- Ne pas utiliser la commande **enable password** et préférer la commande **enable secret**
- Limiter la commande **username** pour des accès non privilégiés.

ATTENTION : certains mots de passe ne sont pas cryptés avec la commande service password-encryption. C'est le cas des noms de communautés SNMP et des clés de chiffrement RADIUS et TACACS. Il faut éviter aussi d'utiliser l'un de ces mots de passe pour l'accès privilégié.

Enfin, il ne faut pas oublier qu'il ne s'agit ici que du stockage des mots de passe dans les fichiers de configuration.

ATTENTION : Selon les protocoles utilisés, ces mots de passe peuvent toujours circuler en clair sur le réseau.

6.3 Port console

R23. Sécuriser l'accès console

Le port console permet la configuration et la supervision du routeur. Il est également utilisé lors de la procédure de récupération des mots de passe. Il faut donc s'assurer que ce port ne sera pas physiquement accessible à tout le monde. Le routeur doit donc se situer dans un local à accès restreint.

Par défaut, il n'existe pas de mot de passe pour l'accès par le port console. Il est impératif d'en définir un. Il est recommandé de préciser la durée au bout de laquelle la session se ferme automatiquement si aucune activité n'est décelée.

M23.

- sélectionner le port console

```
Router(config)#line console 0
```

- activer l'ouverture de session

```
Router(config-line)#login
```

- définir le mot de passe

```
Router(config-line)#password pwdconsole
```

- définir la durée maximale d'inactivité avant la fermeture de la session

```
Router(config-line)#exec-timeout minutes secondes
```

Par défaut, il est possible pour un utilisateur de réaliser une connexion à partir du réseau vers une ligne série. Ce mécanisme est baptisé Reverse-Telnet. Il est recommandé de bloquer cette utilisation sur le port console à l'aide de la commande :

M24.

```
Router(config)#line console 0
Router(config-line)#transport input none
```

6.4 Port auxiliaire

R24. Bloquer le port auxiliaire

Le port auxiliaire est utilisé pour la gestion du routeur à travers une ligne téléphonique. En général, cette solution n'est pas utilisée. Il est donc recommandé de désactiver ce port.

M25.

```
Router(config)#line aux 0
Router(config-line)#login
Router(config-line)#no password
Router(config-line)#no exec
Router(config-line)#exec-timeout 0 1
```

R24bis. Sécuriser le port auxiliaire

Dans le cas où le port auxiliaire est utilisé pour la télégestion du routeur, il est recommandé de demander une ouverture de session avec mot de passe et une fermeture automatique de la session sur inactivité.

M26.

- sélectionner le port auxilaire

```
Router(config)#line aux 0
```

- activer l'ouverture de session

```
Router(config-line)#login
```

- définir le mot de passe

```
Router(config-line)#password pwdaux
```

- définir la durée maximale d'inactivité avant la fermeture de la session

```
Router(config-line)#exec-timeout minutes secondes
```

6.5 Telnet

R25. Ne pas utiliser le protocole telnet et privilégier SSH (cf R26.)

Le protocole telnet permet la configuration et la supervision du routeur à travers le réseau. C'est un protocole non sécurisé puisque les mots de passe et les commandes circulent en clair sur le réseau. Il faut donc éviter de l'utiliser. Au contraire, le protocole SSH garantit la confidentialité des échanges.

M27.

```
Router(config)#line vty 0 4
Router(config-line)#transport input ssh
```

R25bis. Interdire les accès par le réseau

Dans certaines situations, la gestion du routeur peut être réalisée localement. Si le protocole SSH n'est pas disponible, il est préférable d'utiliser le port console et de bloquer tous les accès par le réseau.

M28.

```
Router(config)#line vty 0 4
Router(config-line)#transport input none
```

ATTENTION : une recommandation courante consiste à utiliser la commande **no password** pour bloquer l'accès. Cette option n'est valable que si elle est associée à la commande **login**. Utilisée seule, elle permet l'accès sans mot de passe.

R25ter. Si le protocole telnet est indispensable, sécuriser son utilisation

La syntaxe de définition d'un mot de passe et de la durée de fermeture de session en cas d'inactivité utilise les mêmes commandes que le port console :

M29.

- sélectionner les terminaux virtuels

```
Router(config)#line vty 0 4
```

- activer l'ouverture de session

```
Router(config-line)#login
```

- définir le mot de passe

```
Router(config-line)#password pwdtelnet
```

- définir la durée maximale d'inactivité avant la fermeture de la session

```
Router(config-line)#exec-timeout minutes secondes
```

Il est également possible de limiter l'utilisation du protocole telnet à des adresses IP particulières à l'aide d'une ACL standard (i.e. : pour un accès limité à partir des postes d'administration réseau). Pour plus d'informations sur l'utilisation des listes de contrôle d'accès, voir le chapitre 9.

M30.

- Définir l'autorisation d'accès pour le poste possédant l'adresse *adresse-ip* :

```
Router(config)#access-list acl-telnet permit ip host adresse-ip
```

- Activer la liste de contrôle d'accès sur les terminaux virtuels:

```
Router(config-line)#access-class acl-telnet in
```

R25quattro. Gérer les routeurs et autres équipements réseaux sur un réseau dédié.

Une autre solution consiste à gérer les routeurs et autres équipements réseau sur un réseau dédié.

ATTENTION : la plupart des IOS disposent de cinq terminaux virtuels numérotés de 0 à 4, mais des versions "enterprise" peuvent disposer de 15 terminaux et même plus. Il est possible dans ce cas de supprimer les terminaux virtuels non utilisés.

M31.

- Pour vérifier le nombre de terminaux virtuels :

```
Router#show line vty
```

6.6 SSH

R26. Utiliser exclusivement le protocole SSH pour la gestion du routeur (cf R25.)

SSH est un protocole remplaçant le protocole telnet et garantissant la confidentialité des échanges. SSH est disponible à partir des IOS 12.1 IPSec (DES ou 3DES).

Pour utiliser le protocole SSH, le routeur doit disposer d'un nom (hostname) et d'un nom de domaine.

SSH utilise une paire de clé RSA. Cette paire de clé peut déjà exister si le protocole IPSEC est utilisé, sinon il suffit de la générer. Il est recommandé de choisir une paire de plus de 1024 bits.

M32.

- définir le nom du routeur

```
Router(config)# hostname routeur
```

- définir le nom de domaine

```
Router(config)# ip domain-name nom_de_domaine
```

- générer une paire de clé RSA de plus de 1024 bits.

```
Router(config)# crypto key generate rsa
```

- activer le protocole SSH pour les terminaux virtuels

```
Router(config)# line vty 0 4
Router(config-line)# transport input ssh
```

Les commandes **show ip ssh** et **show ssh** permettent de connaître l'état du serveur SSH et les connexions SSH actives. La commande **crypto key zeroize rsa** permet de supprimer une paire de clé non utilisée.

6.7 Mécanismes d'authentification (AAA)

R27. Mettre en place un mécanisme d'authentification AAA

Les mécanismes AAA (*Authentication, Authorization, Accounting*) assurent l'authentification des connexions, la définition des autorisations et la journalisation des évènements.

Les routeurs Cisco supportent l'authentification locale et les protocoles d'authentification RADIUS et TACACS+. Ces derniers permettent de centraliser les comptes administrateurs, les droits et les journaux (logs) sur des serveurs de sécurité. Les différentes méthodes d'authentification peuvent coexister avec l'utilisation du mot de passe privilégié.

Le choix entre RADIUS et TACACS+ dépend de l'existant. RADIUS est une solution plus simple, ayant fait l'objet de plusieurs RFC. TACACS+ présente plus de fonctionnalités, mais reste un protocole propriétaire Cisco.

6.7.1 Locale

L'authentification locale permet de définir plusieurs comptes. Les comptes sont uniquement locaux.

M33.
```
Router(config)#username nom password mot_de_passe
Router(config)#line vty 0-4
Router(config-line)#login local
Router(config-line)#exit
```

Il est possible de définir des niveaux de privilèges différents pour chaque utilisateur (de 0 à 15, 1 étant le niveau utilisateur, 15 le niveau privilégié). Toutefois, il est recommandé de ne définir que des utilisateurs avec le niveau 1 à l'aide de cette commande.

ATTENTION : il n'est pas recommandé de définir un compte avec le niveau d'accès 15. En effet, les mots de passe de la commande **username** sont chiffrés avec un algorithme réversible et peuvent être récupérés dans le fichier de configuration.

6.7.2 RADIUS

Le protocole RADIUS (*Remote Authentification Dial In User Service*) est un protocole d'authentification, d'autorisation et de journalisation pour les équipements d'un réseau. Il est décrit dans les RFC 2865 et RFC 2866.

RADIUS assure les échanges entre le client (i.e. : le routeur) et le serveur d'authentification, en s'appuyant sur le protocole UDP.

Le routeur et le serveur RADIUS ont connaissance d'un secret partagé. Ce secret permet de calculer des empreintes MD5 des mots de passe. Seules ces empreintes sont envoyées sur le réseau.

ATTENTION : RADIUS n'assure que le chiffrement des mots de passe entre le routeur et le serveur d'authentification. **RADIUS n'assure aucun chiffrement des informations entre l'utilisateur distant et le routeur (utilisation de telnet par exemple).**

La configuration du protocole RADIUS nécessite :

- de définir les paramètres du serveur RADIUS (adresse IP et ports UDP : 1812 et 1813 pour la plupart des serveurs RADIUS) .
- de définir la clé partagée

- de déclarer un nouveau modèle d'authentification

```
M34.
     Router(config)#radius-server host adr-IP_srv-radius auth-port
                    1812 acct-port 1813
     Router(config)#radius-server key clé-partagée
     Router(config)#aaa new-model
     Router(config)#aaa authentication login default radius enable
```

Dans l'exemple précédent, l'authentification est activée pour la commande **login** (telnet et console), d'abord en utilisant le protocole RADIUS, ensuite, si et seulement si le ou les serveurs RADIUS ne répondent pas, en utilisant le mot de passe pour l'accès privilégié (**enable**).

6.7.3 TACACS+

TACACS+ (*Terminal Access Controller Access Control System*) est un protocole d'authentification, développé par Cisco à partir de TACACS. TACACS+ a fait l'objet d'une proposition de RFC (Draft) en 1997 : The TACACS+ Protocol version 1.78.

TACACS+ assure les échanges entre le client (i.e. : le routeur) et le serveur d'authentification, en s'appuyant sur le protocole TCP.

Le routeur et le serveur TACACS+ peuvent utiliser un secret partagé pour chiffrer les messages échangés entre le routeur et le serveur d'authentification.

ATTENTION : TACACS+ n'assure aucun chiffrement des informations entre l'utilisateur distant et le routeur (utilisation de telnet par exemple).

La configuration du protocole TACACS+ nécessite :

- de définir les paramètres du serveur TACACS+ (adresse IP et port TCP) .
- de définir la clé partagée
- de déclarer un nouveau modèle d'authentification

```
M35.
     Router(config)#tacacs-server host adr-IP_srv [port portnumber]
     Router(config)#tacacs-server key clé-partagée
     Router(config)#aaa new-model
     Router(config)#aaa authentication login default tacacs enable
```

Dans l'exemple précédent, l'authentification est activée pour la commande **login** (telnet et console), d'abord en utilisant le protocole TACACS, ensuite, si et seulement si le ou les serveurs TACACS ne répondent pas, en utilisant le mot de passe pour l'accès privilégié (**enable**).

7 Journalisation des évènements

R28. Mettre en place une journalisation des évènements

La journalisation ou "logging" permet d'afficher et de conserver un certain nombre d'informations sur le comportement du routeur. Il existe trois solutions de journalisation :

- A l'écran (console ou telnet)
- En mémoire (buffer)
- Déportée (vers un serveur syslog)

Il est possible de connaître l'état de la journalisation à l'aide de la commande :

```
M36.
     Router #show logging
     Syslog logging: enabled (0 messages dropped, 0 flushes, 0
     overruns)
     Console logging: level debugging, 20 messages logged
     Monitor logging: level debugging, 0 messages logged
     Buffer logging: disabled
     Trap logging: level informational, 24 message lines logged
        Logging to 130.10.0.1, 24 message lines logged
```

7.1 Journalisation vers la console

Par défaut, la journalisation est active et l'affichage est dirigé vers la console.

7.2 Journalisation vers une session Telnet

Il est possible de déporter l'affichage des messages vers une session Telnet à l'aide de la commande :

```
M37.
     Router #terminal monitor
```

Cette commande est utilisable à partir du mode utilisateur. Elle ne concerne que la session en cours.

7.3 Journalisation en mémoire

Les routeurs peuvent mémoriser un certain nombre d'informations en mémoire. Pour cela, il faut définir la taille d'un buffer de stockage en octets à l'aide de la commande :

```
M38.
     Router(config) #logging buffer taille
```

Lorsque le buffer est plein, les nouveaux enregistrements écrasent les plus anciens. Il est visualisable à l'aide de la commande **show logging**

7.4 Journalisation déportée

R28. Mettre en place une journalisation centralisée des évènements

Le buffer local est limité en taille, il est donc intéressant d'envoyer ces informations vers un serveur pour les conserver et les exploiter.

Des systèmes d'enregistrement sont proposés avec les mécanismes d'authentification TACACS+ ou RADIUS et les alarmes SNMP. Toutefois, l'utilisation d'un démon syslog est une solution simple et pratique.

La configuration de la journalisation nécessite de fournir au minimum l'adresse du serveur exécutant le démon syslog.

```
M39.
     Router(config) #logging adresse-ip-syslog
```

ATTENTION : les flux de journalisation Syslog sont émis en clair et sans authentification.

7.5 Niveaux de journalisation

Il est possible de définir le niveau de criticité des évènements journalisés. Le niveau 0 concerne les évènements les plus graves, le niveau 7 les informations de diagnostics. Chaque niveau inclut les évènements des niveaux inférieurs.

Type	Niveau	Description
Emergency	0	Urgent : système hors service
Alerts	1	Alerte : intervention requise
Critical	2	Conditions critiques
Errors	3	Erreurs
Warnings	4	Avertissements
Notifications	5	Conditions d'utilisation normales
Informational	6	Messages d'information
Debugging	7	Diagnostics

Le choix du niveau de journalisation se fait à l'aide des commandes:

- **logging console** *niveau* pour la console

- **logging monitor** *niveau* pour une session telnet
- **logging trap** *niveau* pour un démon syslog

Par défaut, le niveau de journalisation pour la console et une session telnet est positionné à "*debugging*". Le niveau par défaut pour un démon syslog est "*informational*".

7.6 Horodatage

R30. Assurer la synchronisation horaire des routeurs avec une horloge de référence

Les informations enregistrées dans les journaux doivent être datées avec précision pour être correctement exploitées et corrélées avec celles d'autres équipements.

Il est possible de régler manuellement la date et l'heure de chaque routeur, mais la tâche peut devenir trop importante avec le nombre d'équipements.

Le protocole NTP permet de synchroniser automatiquement les équipements d'un réseau avec une horloge de référence, atomique ou GPS.

Ce protocole, ou sa version simplifiée "SNTP", est disponible sur la plupart des IOS récents.

```
M40.
     Router(config)#interface nom_de_l'interface
     Router(config-if)#no ntp disable
     Router(config-if)#exit
     Router(config)#ntp server adresse_ip_serveur_ntp
```

Le protocole NTP nécessite :

- soit de disposer d'un accès Internet vers les serveurs NTP mondiaux
- soit de disposer d'un serveur de référence interne.

Lorsque cela est possible, il est conseillé d'activer l'authentification des messages NTP. Ceci permet d'éviter des techniques d'attaques destinées à modifier l'horodatage des alarmes dans les journaux.

```
M41.
     Router(config)#ntp authentication-key id_cle md5 valeur
     Router(config)#ntp authenticate
     Router(config)#ntp trusted-key id_cle
     Router(config)#ntp server adresse_ip_serveur_ntp key id_cle
```

8 Routage

8.1 Translation d'adresses

R31. Si possible, masquer le plan d'adressage interne.

La translation d'adresse est un mécanisme qui permet de masquer les adresses internes d'un réseau vis-à-vis des réseaux extérieurs. Les adresses internes utilisables sont définies dans la RFC 1918.

Selon le type de translation, le routeur utilise une ou plusieurs adresses publiques (Internet). Il existe trois variantes du mécanisme de translation d'adresse :

- le NAT statique,
- le NAT dynamique,
- le PAT (*Port Address Translation*) ou SUA (*Single User Address*).

8.1.1 Configuration du NAT statique

La configuration d'une translation d'adresse statique consiste à définir :

- une table de correspondance entre les adresses locales (internes) et les adresses globales (externes)
- les interfaces internes ou privées
- les interfaces externes ou publiques

```
M42.
     Router(config)#ip nat inside source static adresse-IP-locale
                    adresse-IP-globale
     Router(config)#interface nom_de_l'interface_interne
     Router(config-if)#ip nat inside
     Router(config-if)#exit
     Router(config)#interface nom_de_l'interface_externe
     Router(config-if)#ip nat outside
     Router(config-if)#exit
```

Seules les adresses locales présentes dans la table de correspondances sont autorisées à utiliser la translation d'adresse.

8.1.2 Configuration du NAT dynamique

Le NAT dynamique permet d'utiliser une plage d'adresses (un "pool") pour la traduction. Dans ce cas, il n'est pas nécessaire de disposer d'autant d'adresses globales que d'adresses locales ayant besoin de sortir. Les adresses globales sont attribuées dynamiquement le temps de la communication.

La configuration du NAT dynamique nécessite donc :

- de déclarer le pool d'adresses globales,
- de définir la liste des adresses locales autorisées à utiliser la translation à l'aide d'une liste de contrôle d'accès standard,
- d'associer le pool d'adresse avec la liste des adresses autorisées,
- d'identifier les interfaces internes ou privées,
- d'identifier les interfaces externes ou publiques.

```
M43.
     Router(config)#ip nat pool nom_pool première_adresse_IP
                    dernière_adresse_IP netmask masque
     Router(config)#access-list num_acl permit adresse_source
                    masque-source
     Router(config)#ip nat inside source list num-acl pool nom-pool
     Router(config)#interface nom_de_l'interface_interne
     Router(config-if)#ip nat inside
     Router(config-if)#exit
     Router(config)#interface nom_de_l'interface_externe
     Router(config-if)#ip nat outside
     Router(config-if)#exit
```

8.1.3 Configuration du PAT

Le PAT se configure comme le NAT mais n'utilise généralement qu'une seule adresse dans la plage. Celle-ci est dite "surchargée" et se configure en précisant **overload**. L'intérêt du PAT est de pouvoir réaliser simultanément une translation entre une adresse globale et plusieurs adresses locales, en utilisant des numéros de port dynamiques.

```
M44.
     Router(config)#ip nat pool nom_pool unique_adresse_IP
                    unique_adresse_IP netmask masque
     Router(config)#access-list num_acl permit adresse_source
                    masque-source
     Router(config)#ip nat inside source list num-acl pool nom-pool
                    overload
     Router(config)#interface nom_de_l'interface_interne
     Router(config-if)#ip nat inside
     Router(config-if)#exit
     Router(config)#interface nom_de_l'interface_externe
     Router(config-if)#ip nat outside
     Router(config-if)#exit
```

8.1.4 Exemple de configuration

Extrait de fichier de configuration

```
ip nat pool plage200 10.0.200.16 10.0.200.16 netmask 255.0.0.0
access-list 1 permit 192.168.200.0 0.0.0.255
ip nat inside source list 1 pool plage200 overload

interface ethernet0
       ip nat inside
interface bri0
       ip nat outside
```

8.2 Filtrage des protocoles de routage dynamiques

R32. Filtrer les protocoles de routage

Le filtrage des protocoles de routage RIP, OSPF, etc. est possible à l'aide de listes de contrôle d'accès en entrée et en sortie des interfaces concernées (pour plus d'informations sur l'utilisation des listes de contrôle d'accès, voir le chapitre 9.). Il est recommandé de contrôler ainsi la cohérence des messages d'information de ces protocoles. A titre d'exemple, les cas de RIP et de OSPF sont décrits ci-dessous.

8.2.1 RIPv1

Les messages RIPv1 sont encapsulés dans UDP et envoyés en diffusion par le routeur. Ils utilisent les ports source et destination RIP (520).

La liste suivante n'autorise ainsi que le trafic RIP sortant effectivement émis par le routeur.

M45.

```
Router(config)#access-list num_acl_sortie permit udp host
adresse_routeur eq rip host 255.255.255.255 eq rip
Router(config)#access-list num_acl_sortie deny udp any eq rip
any eq rip
```

En entrée, le routeur ne doit accepter que des messages RIP reçus en broadcast.

M46.

```
Router(config)#access-list num_acl_entree permit udp any eq
rip host 255.255.255.255 eq rip
Router(config)#access-list num_acl_entree deny udp any eq rip
any eq rip
```

Enfin, lorsque les routeurs voisins sont parfaitement connus et que l'architecture est maîtrisée, il est possible de restreindre l'origine des messages RIP aux adresses des routeurs voisins.

```
M47.
     Router(config)#access-list num_acl_entree permit udp host
     adresse_routeur_voisin eq rip host 255.255.255.255 eq rip
     Router(config)#access-list num_acl_entree deny udp any eq rip
     any eq rip
```

8.2.2 RIPv2

Les messages RIPv2 ne sont pas envoyés en diffusion mais en multicast (224.0.0.9). Ils utilisent les mêmes ports source et destination que RIPv1 (520).

La liste suivante n'autorise ainsi que le trafic RIPv2 sortant effectivement émis par le routeur.

```
M48.
     Router(config)#access-list num_acl_sortie permit udp host
     adresse_routeur eq rip host 224.0.0.9 eq rip
     Router(config)#access-list num_acl_sortie deny udp any eq rip
     any eq rip
```

En entrée, le routeur ne doit accepter que des messages RIPv2 reçus sur l'adresse multicast 224.0.0.9.

```
M49.
     Router(config)#access-list num_acl_entree permit udp any eq
     rip host 224.0.0.9 eq rip
     Router(config)#access-list num_acl_entree deny udp any eq rip
     any eq rip
```

Enfin, lorsque les routeurs voisins sont parfaitement connus et que l'architecture est maîtrisée, il est possible de restreindre l'origine des messages RIPv2 aux adresses des routeurs voisins.

```
M50.
     Router(config)#access-list num_acl_entree permit udp host
     adresse_routeur_voisin eq rip host 224.0.0.9 eq rip
     Router(config)#access-list num_acl_entree deny udp any eq rip
     any eq rip
```

8.2.3 OSPF

Le protocole OSPF est directement encapsulé dans IP. Les messages sont soit diffusés à l'aide d'une adresse IP multicast (224.0.0.5), soit échangés entre routeurs voisins.

La liste suivante n'autorise le trafic OSPF sortant que s'il est envoyé par le routeur lui-même.

```
M51.
     Router(config)#access-list num_acl_sortie permit ospf host
                    adresse_routeur host 224.0.0.5
     Router(config)#access-list num_acl_sortie permit ospf host
                    adresse_routeur any
     Router(config)#access-list num_acl_sortie deny ospf any any
```

Inversement, il est possible de n'autoriser le trafic OSPF entrant que s'il est destiné au routeur lui-même ou à l'adresse de multicast du protocole (224.0.0.5).

```
M52.
     Router(config)#access-list num_acl_entrée permit ospf any host
                    224.0.0.5
     Router(config)#access-list num_acl_entrée permit ospf any host
                    adresse_routeur
     Router(config)#access-list num_acl_entrée deny ospf any any
```

Enfin, lorsque les routeurs voisins sont parfaitement connus et que l'architecture est maîtrisée, il est recommandé de restreindre le trafic OSPF à celui qui est effectivement échangé avec ces voisins.

```
M53.
     Router(config)#access-list num_acl_sortie permit ospf host
                    adresse_routeur host 224.0.0.5
     Router(config)#access-list num_acl_sortie permit ospf host
                    adresse_routeur host adresse_voisin

     Router(config)#access-list num_acl_entrée permit ospf host
                    adresse_voisin host 224.0.0.5
     Router(config)#access-list num_acl_entrée permit ospf host
                    adresse_voisin host adresse_routeur
```

8.3 Protocoles de routage dynamiques authentifiés

R33. Assurer l'authentification des informations de routage

L'authentification des routeurs voisins permet de certifier les informations de routage échangées à l'aide des protocoles dynamiques.

L'authentification est possible avec les protocoles OSPF, BGP, RIPv2 et EIGRP. Elle doit être configurée sur l'ensemble des routeurs concernés. Elle est basée soit sur l'échange d'un mot de passe partagé, soit une signature MD5 des messages émis. Cette dernière solution est préférable puisque le mot de passe ne circule pas sur le réseau.

RIPv2 et EIGRP permettent d'utiliser des clés à durée de vie limitée, avec renouvellement automatique. Il est nécessaire dans ce cas d'assurer la synchronisation horaire de tous les routeurs (voir Recommandation R28).

8.3.1 RIPv2

Le protocole RIPv2 accepte une signature MD5. Pour cela, il est nécessaire de:

- s'assurer que la version RIP utilisée est bien la v2.
- définir des clés partagées par les routeurs dans une chaîne de clé
- fixer le mode d'authentication md5 sur les interfaces concernées
- préciser la chaîne de clé utilisée pour l'interface.

```
M54.
     Router(config)#router rip
     Router(config-router)#version 2
     Router(config-router)#exit
     Router(config)#key chain nom_chaine
     Router(config-keychain)#key 1
     Router(config-keychain-key)#key-string cle-partagee
     Router(config-keychain-key)#exit
     Router(config-keychain)#exit
     Router(config)#interface nom_de_l'interface
     Router(config-if)#ip rip authentication mode md5
     Router(config-if)#ip rip authentication key-chain nom_chaine
     Router(config)#end
```

8.3.2 OSPF

Le protocole OSPF utilise également une signature MD5. La clé partagée est définie au niveau de l'interface.

```
M55.
     Router(config)#router ospf process_id
     Router(config-router)#network num_réseau masque area zone
     Router(config-router)#area zone authentication message-digest
     Router(config-router)#exit
     Router(config)#interface nom_de_l'interface
     Router(config-if)#ip ospf message-digest-key id md5 clé
     Router(config)#end
```

8.4 Unicast Reverse Path Forwarding

R34. Activer la vérification des routes inverses

Les versions d'IOS 12.0 et ultérieures supportent les mécanismes de CEF (Cisco Express Forwarding) et de RPF (Reverse Path Forwarding).

Le routeur peut ainsi vérifier que l'adresse source d'un datagramme qui arrive sur une interface est effectivement accessible par cette interface : i.e. le routeur connaît une route vers cette adresse et cette route emprunte cette interface. Dans le cas contraire, le datagramme est rejeté.

Attention, ce mécanisme ne fonctionne que lorsque le routage est symétrique : la même route doit être utilisée dans les deux sens.

Le mécanisme RPF nécessite en outre que le CEF soit préalablement activé.

```
M56.
     Router(config)#ip cef
     Router(config)#interface nom-interface
     Router(config-if)#ip verify unicast rpf
```

Il est possible de connaître le nombre de datagrammes rejetés par la commande **show ip traffic.**

```
Router#show ip traffic
     IP statistics:
     Rcvd:  627 total, 339 local destination
            1 format errors, 0 checksum errors, 4 bad hop count
            0 unknown protocol, 0 not a gateway
            0 security failures, 0 bad options, 0 with options
     Opts:  0 end, 0 nop, 0 basic security, 0 loose source route
            0 timestamp, 0 extended security, 0 record route
            0 stream ID, 0 strict source route, 0 alert, 0 cipso
            0 other
     Frags: 0 reassembled, 0 timeouts, 0 couldn't reassemble
            0 fragmented, 0 couldn't fragment
     Bcast: 321 received, 8 sent
     Mcast: 0 received, 0 sent
     Sent:  30 generated, 201 forwarded
     Drop:  13 encapsulation failed, 0 unresolved, 0 no adjacency
            62 no route, 8 unicast RPF, 0 forced drop
```

9 Filtrage : les listes de contrôle d'accès

R35. Appliquer une politique de filtrage à l'aide de listes de contrôle d'accès.

9.1 Définir une politique de sécurité

Avant de commencer la sécurisation du réseau, il est nécessaire de déterminer une politique de sécurité répondant de façon simple à la question suivante :

Qu'est-ce qui est permis et qu'est-ce qui ne l'est pas ?

Une conception communément admise de la politique de sécurité est d'identifier les flux autorisés et de rejeter tout le reste : « Tout ce qui n'est pas explicitement autorisé doit être interdit ».

Il faut donc identifier précisément l'ensemble des flux réseau autorisés traversant le routeur :

- Les services utilisés par les utilisateurs internes sur le réseau externe (messagerie, dns, …)
- Les services fournis par le réseau interne aux utilisateurs externes (web, …)
- Les services fournis par le routeur lui-même ou nécessaires à son administration ou sa gestion (telnet, protocoles de routage, SNMP, TFTP,…)

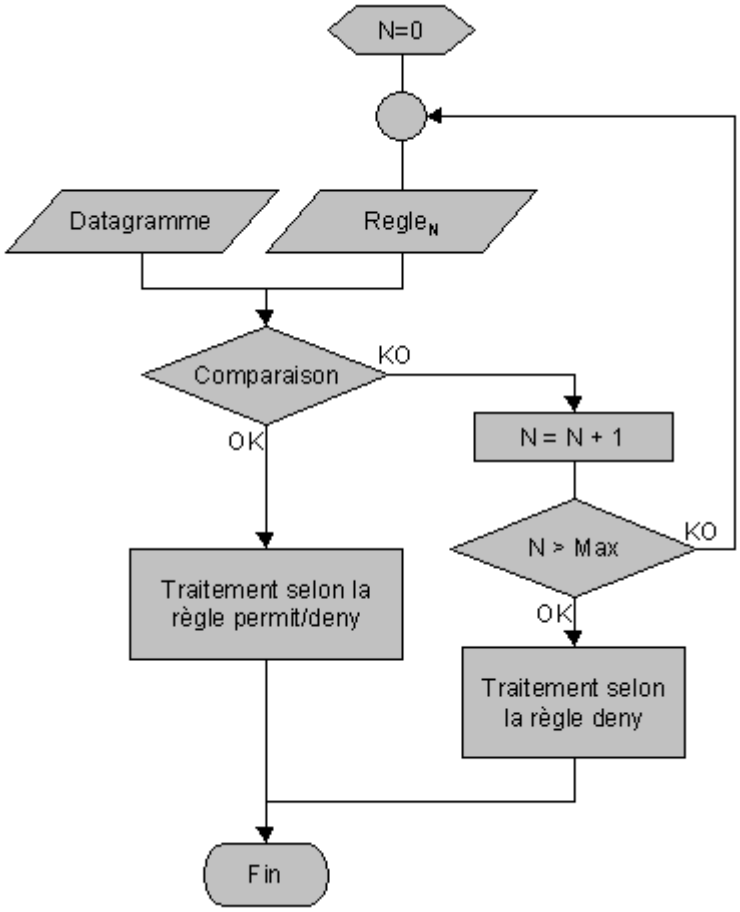
On veillera à implémenter dans la politique de sécurité l'anti-spoofing :

- en interdisant sur la patte extérieure du routeur tout trafic entrant dont l'adresse source est comprise dans le plan d'adressage interne,
- en interdisant sur la patte intérieure du routeur tout trafic sortant dont l'adresse source n'est pas dans le plan d'adressage interne,
- en interdisant sur les deux pattes du routeur tout trafic entrant dont l'adresse source est incohérente (adresse de bouclage, adresse 0.0.0.0…).

9.2 Définition d'une ACL (Access Control List)

Les listes de contrôle d'accès permettent de définir les flux de trafic autorisés ou rejetés par le routeur, définis dans la politique de sécurité.

Les instructions de la liste sont traitées de façon séquentielle : dès qu'une instruction fait l'objet d'une correspondance, la recherche prend fin et l'action définie par l'instruction est exécutée.



Logique implémentée par une ACL :

- Le paquet est comparé aux paramètres de la première instruction
- Si une correspondance est trouvée, l'action définie (autoriser, rejeter) est exécutée
- Sinon les deux premières étapes sont répétées avec l'instruction suivante.
- Si aucune correspondance n'est trouvée, le paquet est rejeté par défaut

ATTENTION : certaines anciennes versions d'IOS peuvent laisser passer un paquet sans correspondance aussi, on veillera à implémenter en dernière instruction de l'ACL un refus de tout paquet ne correspondant à aucune des règles précédentes :

```
Router(config)#access-list num-acl deny ip any any

Router(config)#access-list standard nom-acl
Router(config-std-nacl)#deny any any
```

9.2.1 Listes d'accès IP standards

Les listes d'accès standards recherchent des correspondances en examinant uniquement le champ « adresse IP source » dans l'en-tête IP d'un paquet. N'importe quels bits sur les 32 de l'adresse peuvent être comparés.

Un masque générique est utilisé pour définir le sous-ensemble de bits devant correspondre dans les 32 bits de l'adresse IP. Contrairement à l'utilisation classique des masques de sous-réseau, les bits du masque dont la valeur est égale à 0 indiquent les positions binaires à comparer. Ceux dont la valeur est à 1 sont dits génériques : ces bits ne sont pas comparés (cf. « exemple de masques génériques » en annexe).

- Syntaxe :

```
Router(config)#access-list num-acl {permit|deny} {source [masque]}
```

num-acl : identifie la liste à laquelle l'entrée appartient ; nombre entre 1 et 99

permit|deny : indique si l'entrée autorise ou bloque le trafic en provenance de l'adresse spécifiée

source : adresse IP de la source

masque : indique quels bits de l'adresse doivent être comparés. 0.0.0.0 par défaut si omis (adresse d'un hôte).

NB : Les nouvelles lignes sont toujours ajoutées à la fin. Il n'est pas possible d'ajouter ou d'enlever des lignes de façon sélective. Il est donc recommandé d'effacer toute définition existante d'une ACL avant de la reconstruire :

```
Router(config)#no access-list num-acl
```

De même, l'ajout d'une nouvelle instruction dans une liste numérotée requiert la suppression de la liste entière, puis l'insertion de toutes les instructions.

9.2.2 Listes d'accès IP étendues

Une liste d'accès étendue permet de comparer un plus grand nombre de champs dans un paquet IP. Une instruction est considérée comme correspondante si une correspondance est trouvée pour chacun de ses paramètres.

- Syntaxe :

```
Router(config)#access-list num-acl {permit|deny} {protocole}
               {source masque-src} {destination masque-dst}
               [options-spécifiques] [log]
```

num-acl : identifie la liste à laquelle l'entrée appartient ; nombre entre 100 et 199.

source : adresse IP de la source

masque : indique quels bits de l'adresse doivent être comparés. 0.0.0.0 par défaut si omis (adresse d'un hôte).

protocole : ip, tcp, udp, icmp, igmp, gre, igrp, eigrp, ospf, nos, … ou un nombre entre 0 et 255.

log : messages d'historique envoyés à la console quand l'instruction est vérifiée. Attention : consomme du CPU.

Possibilité d'envoyer les événements vers un serveur syslog :

```
Router(config)#logging trap debugging
Router(config)#logging adresse-serveur-syslog
```

- Mots clés :

any : adresse 0.0.0.0 et masque générique 255.255.255.255, correspond à n'importe quelle valeur.

host suivi d'une @IP : s'applique uniquement à cette adresse précise (= adresse d'un hôte), est utilisé à la place d'une adresse précise suivie du masque 0.0.0.0

- Options spécifiques :

Filtrage du protocole ICMP

```
Router(config)#access-list num-acl {permit|deny} icmp {source
               masque-src} {destination masque-dst} [icmp-type
               [icmp-code] | icmp-message]
```

Permet de filtrer les paquets ICMP sur les champs type et code (0 à 255) ou à l'aide d'un message représentant une combinaison des deux. Exemple : echo, echo-reply. Utiliser l'aide contextuelle avec « **?** » pour connaître la liste des messages reconnus.

Filtrage du protocole TCP

```
Router(config)#access-list num-acl {permit|deny} tcp {source
     masque-src} [[lt|gt|eq|neq] port-src] {destination masque-dst}
     [[lt|gt|eq|neq] port-dst] [established]
```

port : 0 à 65535 ou un nom représentatif. Utiliser l'aide contextuelle pour obtenir la liste des abréviations disponibles.

established : teste si le bit ACK ou RST est positionné. Indique que le paquet fait partie d'une connexion existante. Permet de filtrer le sens d'établissement de la connexion.

Filtrage du protocole UDP

```
Router(config)#access-list num-acl {permit|deny} udp {source
               masque-src} [[lt|gt|eq|neq] port-src]{destination
               masque-dst} [[lt|gt|eq|neq] port-dst]
```

9.2.3 Listes d'accès IP nommées

Les listes d'accès nommées implémentent le même logique que les listes d'accès numérotées et possèdent les avantages suivants :

- Nom représentatif
- Pas de limitation de nombre
- Les instructions des listes d'accès nommées peuvent être supprimées séparément, ce qui n'est pas le cas des listes numérotées.

```
Router(config)#access-list {standard|extended} nom-acl
Router(config-std-nacl)#{deny|permit} source masque-src
Router(config-ext-nacl)#{deny|permit} protocol source masque-src
               [port] destination masque-dst [port] [options-
               spécifiques]
```

Les listes de contrôle d'accès nommées sont limitées aux protocoles TCP/IP.

9.2.4 Listes d'accès IP réflexives

Les listes d'accès IP nommées sont utilisées, entre autre, dans le cadre des listes d'accès réflexives. Ces dernières sont des listes d'accès dynamiques qui permettent de gérer les flux retours en association avec une liste d'accès étendue nommée.

Les flux autorisés sur la première liste étendue mettent à jour la liste réflexive lorsqu'ils sont autorisés (adresses et ports source et destination). Cette solution est plus restrictive que l'utilisation du mot clé established ou les restrictions sur les ports supérieurs à 1024 sur les listes d'accès étendues numérotées. Elle permet de restreindre les flux de retour aux seuls flux en adéquation avec ceux qui sont effectivement déjà passés dans l'autre sens sur la première liste.

```
Router(config)#ip access-list extended nom-acl-initiale
Router(config-ext-nacl)#permit protocol source masque-src [port]
                destination maque-dst [port] reflect identifiant-
                flux

Router(config)#ip access-list extended nom-acl-reflexive
Router(config-ext-nacl)#evaluate identifiant-flux
```

9.3 Appliquer une ACL sur une interface

9.3.1 Commande

Une fois définies, il faut appliquer les ACL aux paquets entrants ou sortants d'une interface particulière :

```
Router(config)#ip access-group num-acl|nom-acl {in|out}
```

NB : **out** est l'interprétation par défaut si rien n'est précisé.

ATTENTION : Lorsqu'on applique une ACL sur une interface, les nouvelles lignes créées ensuite dans l'ACL correspondante prennent effet immédiatement. Il vaut donc mieux créer une ACL complètement avant de l'appliquer.

9.3.2 Positionnement :

Dans une configuration complexe, il existe souvent plusieurs choix de routeurs et d'interfaces pour positionner les ACL, une des questions primordiale est donc : Où placer les ACL ?

- En entrée plutôt qu'en sortie : les paquets indésirables n'entrent pas dans le processus de routage.
- ACL standard : le plus près possible de la destination
- ACL étendue : Il existe moins de contraintes sur le positionnement des ACL étendues que pour les ACL standards. En général, on essaie de :
    - Minimiser la distance parcourue par le trafic qui doit être rejeté.
    - Prévoir l'évolution du réseau
    - Tenir compte de la charge CPU des routeurs implémentant les ACL

9.3.3 Optimisation des ACL

Le nombre d'ACL à parcourir pour traiter un paquet peut impacter directement sur les performances (Cf. 9.2). De plus la traduction d'une politique de sécurité en ACL en point à point peut rapidement dépasser la capacité mémoire du routeur et devenir difficilement gérable.

Il est par conséquent souhaitable dans le cadre de déploiements de réseaux offrant divers services, de définir le plan d'adressage des postes en liaison avec la politique de sécurité. Ainsi les ACL pourront largement être positionnées à l'aide de masques.

De même, lorsque les ACL sont nombreuses, pour minimiser le traitement de contrôle de celles-ci sur chaque paquet, il peut être judicieux d'ordonner les ACL dans l'ordre décroissant du nombre de paquets transitant sur l'interface.

9.3.4 Turbo ACL

Les versions d' IOS 12.1(6) et ultérieures supportent les listes de contrôle d'accès compilées, appelée « Turbo ACLs ». Ce mécanisme permet d'améliorer les performances du filtrage.

```
Router(config)#access-list compiled.
```

9.3.5 Vérification

Lister toutes les ACL

```
Router(config)#show access-list
```

Visualiser une ACL particulière

```
Router(config)# show ip access-list [acl-num]
```

Visualiser une ACL compilée (Turbo-ACL)

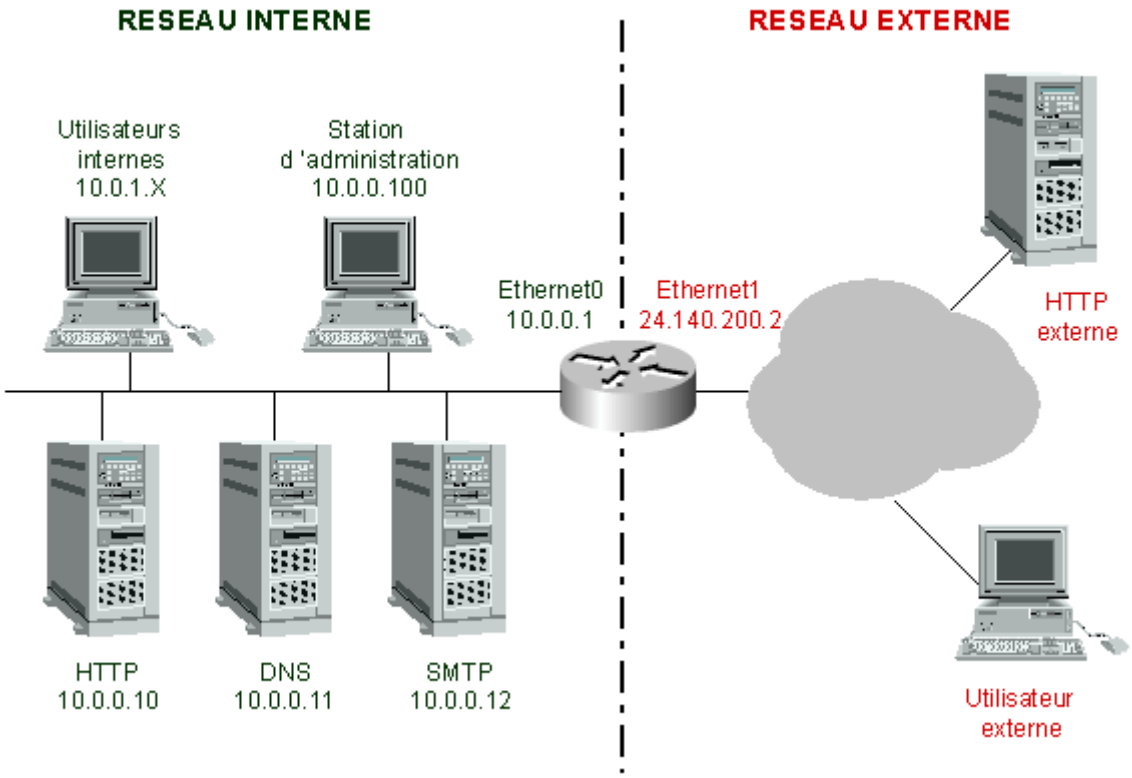
```
Router(config)# show access-list compiled
```

Pour les ACL étendues, le nombre d'utilisations est indiqué, pour remettre les compteurs à zéro :

```
Router(config)# clear access-list counters [acl-num]
```

9.4 Exemple

9.4.1 Architecture réseau



9.4.2 Politique de sécurité

- Interdire le trafic entrant si l'adresse source fait partie du réseau interne (anti-spoofing).
- Autoriser le trafic TCP entrant faisant partie d'une session établie.
- Autoriser les connexions HTTP vers le serveur Web interne
- Autoriser les connexions SMTP vers le serveur Email interne
- Autoriser les connexions DNS vers le serveur DNS interne
- Autoriser les mises à jour par le protocole de routage OSPF
- Autoriser les pings sur l'interface externe du routeur
- Autoriser les pings entrants sur les serveurs
- Autoriser les pings vers l'extérieur depuis la station d'administration
- Autoriser les accès telnet sur le routeur depuis la station d'administration
- Interdire tous les autres trafics entrants

- Autoriser tous les trafics sortants (NB : la politique pourrait être beaucoup plus restrictive).
- Désactiver les services inutiles sur le routeur

9.4.3 Fichier de configuration

```
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
service password-encryption
!
hostname Router
!
enable secret 5 $1$YtuE$lo/NsFhfAnz75VwjR.mdY/
!
memory-size iomem 10
ip subnet-zero
no ip source-route
no ip finger
no ip domain-lookup
!
!
interface FastEthernet0/0
 ip address 10.0.0.1 255.0.0.0
 no ip unreachables
 no ip directed-broadcast
 no cdp enable
 no ntp disable

! en supposant qu'il y a un serveur NTP en 10.0.0.111
 ntp server 10.0.0.111!

interface FastEthernet0/1
 ip address 24.140.200.2 255.255.255.0
 ip access-group 110 in
 no ip redirects
 no ip unreachables
 no ip mask-reply
 no ip directed-broadcast
 no ip proxy-arp
 no ip route-cache
 no ip mroute-cache
no cdp enable

!
router ospf 10
 network 10.0.0.0 0.255.255.255 area 1
 network 24.140.200.0 0.0.0.255 area 1
!
ip classless
no ip http server
!
access-list 10 permit 10.0.0.100
```

```
access-list 110 deny   ip 10.0.0.0 0.255.255.255 any
access-list 110 permit tcp any 10.0.0.0 0.255.255.255 gt 1023
established
access-list 110 permit tcp any host 10.0.0.10 eq www
access-list 110 permit tcp any host 10.0.0.11 eq domain
access-list 110 permit udp any host 10.0.0.11 eq domain
access-list 110 permit tcp any host 10.0.0.12 eq smtp
access-list 110 permit udp host 10.0.0.111 host 10.0.0.1 eq ntp
access-list 110 permit ospf any host 224.0.0.5
access-list 110 permit ospf any host 224.0.0.6
access-list 110 permit ospf any host 24.140.200.2
access-list 110 permit icmp any host 24.140.200.2 echo
access-list 110 permit icmp any host 10.0.0.10 echo
access-list 110 permit icmp any host 10.0.0.11 echo
access-list 110 permit icmp any host 10.0.0.12 echo
access-list 110 permit icmp any host 10.0.0.100 echo-reply
no cdp run
!
line con 0
 exec-timeout 25 0
 password 7 14141B180F0B
 login
 transport input none
line aux 0
 exec-timeout 0 0
line vty 0 4
 input ssh
 access-class 10 in
 exec-timeout 20 0
 password 7 14141B180F0B
 login
!
end
```

10 Outils d'audit : Router Auditing Tool

RAT (http://www.cisecurity.org/bench_cisco.html) est un script Perl qui analyse de façon automatique la configuration d'un routeur Cisco, pour identifier les écarts par rapport aux recommandations (NSA/CIS) et aux services attendus.

RAT utilise un jeu de question/réponse afin de cerner la politique de sécurité (utilisation de SSH, administration SNMP…).

```
ncat_config
```

A partir des réponses, il analyse la configuration du routeur à partir d'un fichier de configuration

```
rat <fichier-conf>
```

Le résultat de l'analyse est présenté sous forme d'un fichier HTML où les éléments à corriger apparaissent en rouge. Enfin, un lien hypertexte est positionné pour chaque test afin d'afficher la recommandation correspondante.