Guide de conception et d'application du pare-feu de stratégie basé sur la zone

Contenu

Introduction

La version 12.4(6)T du logiciel Cisco IOS® a mis en place la politique de pare-feu selon des zones (ZFW), un nouveau modèle de configuration pour l'ensemble de fonctionnalités du pare-feu Cisco IOS. Ce nouveau modèle de configuration propose des politiques intuitives pour les routeurs à

interfaces multiples, une plus grande granularité de l'application de la politique de pare-feu et une politique de déni par défaut qui empêche le trafic entre les zones de sécurité du pare-feu jusqu'à ce qu'une politique explicite soit appliquée pour permettre un trafic souhaitable.

Quasiment toutes les fonctionnalités du pare-feu classique de Cisco IOS mises en application avant la version 12.4(6)T du logiciel Cisco IOS sont prises en charge par cette nouvelle interface d'inspection de la politique selon des zones :

- Inspection dynamique des paquets
- Pare-feu Cisco IOS sensible à VRF
- Filtrage des URL
- Réduction du déni de service (DoS)

La version 12.4(9)T du logiciel Cisco IOS a ajouté la prise en charge ZFW pour la session/connexion par classe et des limites de débit, aussi bien que l'inspection de l'application et le contrôle :

- HTTP
- Post Office Protocol (POP3), Internet Mail Access Protocol (IMAP), Simple Mail Transfer Protocol/Enhanced Simple Mail Transfer Protocol (SMTP/ESMTP)
- Sun Remote Procedure Call (RPC)
- Applications de messagerie instantanée (IM) :Microsoft MessengerYahoo! MessengerAOL Instant Messenger
- Partage de fichiers en pair à pair (P2P) :BittorrentKaZaAGnutellaeDonkey

La version 12.4(11)T du logiciel Cisco IOS a ajouté des statistiques pour un ajustement plus facile de la protection DoS.

Certaines fonctionnalités et capacités du pare-feu classique de Cisco IOS ne sont pas encore prises en charge dans un ZFW dans la version 12.4(15)T du logiciel Cisco IOS :

- Proxy d'authentification
- Basculement dynamique du pare-feu
- MIB unifié de pare-feu
- Inspection dynamique d'IPv6
- Assistance aux pannes TCP

ZFW améliore d'une manière générale les performances de Cisco IOS pour la plupart des activités d'inspection du pare-feu.

Ni ZFW ni le pare-feu classique de Cisco IOS n'incluent la prise en charge de l'inspection dynamique du trafic de multidiffusion.

Conditions préalables

Conditions requises

Aucune spécification déterminée n'est requise pour ce document.

Composants utilisés

Ce document n'est pas limité à des versions de matériel et de logiciel spécifiques.

Conventions

Pour plus d'informations sur les conventions utilisées dans ce document, reportez-vous à Conventions relatives aux conseils techniques Cisco.

Aperçu de la politique selon des zones

L'inpection dynamique du pare-feu classique de Cisco IOS (auparavant connu sous le nom de contrôle d'accès selon le contexte ou CBAC) a utilisé le modèle de configuration sur la base d'une interface, dans lequel une politique d'inspection dynamique a été appliquée à une interface. Tout le trafic qui passe par cette interface a reçu la même politique d'inspection. Ce modèle de configuration a limité la granularité des politiques de pare-feu et a entraîné la confusion de l'application appropriée des politiques de pare-feu, en particulier dans les scénarios où les politiques de pare-feu doivent être appliquées entre plusieurs interfaces.

Le pare-feu de la politique selon les zones (également connu sous le nom de pare-feu de zone politique ou ZFW) change la configuration du pare-feu de l'ancien modèle basé sur l'interface pour un modèle plus souple et plus facilement compréhensible basé sur des zones. Des interfaces sont affectées aux zones et la politique d'inspection est appliquée au trafic qui se déplace entre les zones. Les politiques interzonales offrent une flexibilité et une granularité considérables, afin que différentes politiques d'inspection puissent être appliquées aux multiples groupes hôtes connectés à la même interface du routeur .

Des politiques de pare-feu sont configurées avec la politique linguistique de Cisco® (CPL), qui utilise une structure hiérarchisée pour définir l'inspection pour des protocoles de réseau et les groupes d'hôtes auxquels l'inspection sera appliquée.

Modèle de la politique de configuration selon des zones

ZFW change complètement la façon dont vous configurez une inspection de pare-feu Cisco IOS, comparativement au pare-feu classique de Cisco IOS.

Le premier changement majeur de la configuration du pare-feu consiste en la mise en place d'une configuration sur la base de zones. Le pare-feu Cisco IOS est la première fonctionnalité de défense contre des menaces du logiciel Cisco IOS pour mettre en application un modèle de configuration par zone. D'autres fonctionnalités pourraient venir à adopter le modèle zonal au cours du temps. Le modèle de configuration sur la base d'interface (ou CBAC) de l'inspection dynamique du pare-feu classique de Cisco IOS qui utilise l'ensemble de commande **ip inspect** est conservé pendant un certain temps. Cependant, peu de nouvelles fonctionnalités, voire aucune, sont configurables avec l'interface de ligne de commande (CLI) classique. ZFW n'utilise pas l'inspection dynamique ou les commandes CBAC. Les deux modèles de configuration peuvent être utilisés simultanément sur des routeurs, mais pas combinés sur des interfaces. Une interface ne peut pas être configurée comme membre d'une zone de sécurité ni être configurée pour l'**ip inspect** simultanément.

Les zones établissent les frontières de sécurité de votre réseau. Une zone définit une borne où le trafic est soumis aux restrictions politiques à mesure qu'elle se dirige vers une autre region de votre réseau. La politique par défaut de ZFW entre les zones est tout refuser. Si aucune politique n'est explicitement configurée, tout le trafic qui se déplace entre les zones est bloqué. C'est un départ significatif depuis le modèle de l'inspection dynamique où le trafic a été implicitement autorisé jusqu'à être explicitement bloqué avec une liste de contrôle d'accès (ACL).

Le deuxième principal changement consiste en la mise en place d'une nouvelle configuration de la politique linguistique connue sous le nom de CPL. Les utilisateurs familiarisés avec la CLI (MQC) de la qualité de service (QoS) modulaire du logiciel Cisco IOS pourront constater que le format est semblable à l'utilisation des cartes de classe de QoS pour spécifier quel trafic sera affecté par l'action appliquée dans une carte de politique.

Règles pour l'application de la politique de pare-feu selon des zones

L'adhésion d'interfaces de réseau de routeurs dans des zones est sujette à plusieurs règles qui régissent le comportement de l'interface, de même que le trafic qui se déplace entre les interfaces de zone membre :

- Une zone doit être configurée avant que des interfaces puissent être affectées à la zone.
- Une interface peut être affectée à seulement une zone de sécurité.
- Tout trafic en direction et en provenance d'une interface donnée est implicitement bloqué quand l'interface est affectée à une zone, excepté pour le trafic en direction et en provenance d'autres interfaces dans la même zone et le trafic vers toute interface sur le routeur.
- Le trafic est implicitement autorisé à s'écouler par défaut parmi les interfaces qui sont membres de la même zone.
- Afin d'autoriser le trafic en direction et en provenance d'une interface de zone membre, une politique autorisant ou inspectant le trafic doit être configurée entre cette zone et n'importe quelle autre zone.
- La zone individuelle est la seule exception à la politique par défaut tout refuser. Tout trafic vers n'importe quelle interface de routeur est autorisé jusqu'à ce que le trafic soit explicitement refusé.
- Le trafic ne peut pas s'écouler entre une interface d'un membre de zone et toute interface qui n'est pas un membre de zone. Les actions de passage, inspection et abandon peuvent seulement être appliquées entre deux zones.
- Les interfaces qui n'ont pas été affectées à une fonction de zone en tant que ports de routeur classiques et pourraient encore utiliser une configuration inspection/CBAC dynamique classique.
- S'il est nécessaire qu'une interface sur la boîte ne fasse pas partie de la politique de répartition en zones/pare-feu. Il pourrait encore être nécessaire de placer cette interface dans une zone et de configurer une politique « tout passer » (genre de politique factice) entre cette zone et n'importe quelle autre zone vers lesquelles le flux de trafic est souhaité.
- Il ressort de ce qui précède que, si le trafic doit s'écouler parmi toutes les interfaces dans un routeur, toutes les interfaces doivent faire partie du modèle de répartition en zones (chaque interface doit être membre d'une zone ou d'une autre).
- La seule exception au déni précédent d'approche par défaut est le trafic en direction et en provenance du routeur, qui sera permis par défaut. Une politique explicite peut être configurée pour restreindre ce trafic.

Conception de la politique de sécurité du réseau selon des zones

Une zone de sécurité devrait être configurée pour chaque region de sécurité relative dans le réseau, de sorte que toutes les interfaces qui sont affectées à la même zone soient protégées

avec un niveau de sécurité semblable. Par exemple, considérez un routeur d'accès avec trois interfaces :

- Un interface connectée à l'Internet public
- Une interface connectée à un LAN privé qui ne doit pas être accessible depuis l'Internet public
- Une interface connectée à une zone démilitarisée de service Internet (DMZ), où un serveur Web, système de noms de domaine (DNS) et serveur de messagerie électronique doivent être accessibles à l'Internet public

Chaque interface de ce réseau sera affectée à sa propre zone, bien que vous puissiez vouloir permettre divers accès de l'Internet public aux hôtes spécifiques dans le DMZ et diverses politiques d'utilisation d'application pour des hôtes dans le LAN protégé. (Voir la figure 1.)

Figure 1 : Topologie de zone de sécurité de base

Dans cet exemple, chaque zone contient seulement une interface. Si une interface supplémentaire est ajoutée à la zone privée, les hôtes connectés à la nouvelle interface dans la zone peuvent passer du trafic vers tous les hôtes sur l'interface existante dans la même zone. En outre, le trafic d'hôtes vers des hôtes d'autres zones est pareillement affecté par des politiques existantes.

Généralement, le réseau d'exemple aura trois politiques principales :

- Connectivité de zone privée à Internet
- Connectivité de zone privée aux hôtes DMZ
- Connectivité de zone Internet aux hôtes DMZ

Puisque le DMZ est exposé à l'Internet public, les hôtes DMZ pourraient être soumis à l'activité non souhaitée de personnes malveillantes qui pourraient réussir à compromettre un ou plusieurs hôtes DMZ. Si aucune politique d'accès n'est fournie pour que des hôtes DMZ atteignent soit des hôtes de zone privés soit des hôtes de zone Internet, alors les personnes qui ont compromis les hôtes DMZ ne peuvent pas utiliser les hôtes DMZ pour conduire toute nouvelle attaque contre des hôtes privés ou Internet. ZFW impose une position de sécurité prohibitive par défaut. Par conséquent, à moins que les hôtes DMZ aient spécifiquement accès à d'autres réseaux, d'autres réseaux sont sauvegardés contre toutes les connexions des hôtes DMZ. De même, aucun accès n'est donné aux hôtes Internet pour accéder aux hôtes de zone privée, ainsi les hôtes de zone privée sont sûrs d'avoir un accès non désiré par des hôtes Internet.

Utilisation de VPN IPSec avec la politique de pare-feu selon des zones

Les améliorations récentes de VPN IPSec simplifient la configuration de la politique de pare-feu pour la connectivité de VPN. IPSec Virtual Tunnel Interface (VTI) et GRE+IPSec permettent la restriction de site à site VPN et les connexions du client à une zone de sécurité spécifique par le placement des interfaces du tunnel dans une zone de sécurité déterminée. Des connexions peuvent être isolées dans un DMZ VPN si la connectivité doit être limitée par une politique spécifique. Ou, si la connectivité de VPN est implicitement reconnue comme sûre, la connectivité de VPN peut être placée dans la même zone de sécurité que le réseau interne reconnu comme sûr.

Si un IPSec non-VTI est appliqué, la politique de pare-feu de connectivité VPN requiert un examen minutieux pour assurer la sécurité. La politique zonale doit spécifiquement permettre l'accès d'une adresse IP pour les hôtes des sites distants ou les clients VPN si les hôtes sûrs sont

dans une zone différente que la connexion encryptée VPN du client au routeur. Si la politique d'accès n'est pas correctement configurée, les hôtes qui devraient être protégés peuvent finir exposés aux hôtes non désirés et potentiellement hostiles. Reportez-vous à l' Utilisation de VPN avec la politique de pare-feu selon les zones pour une plus ample discussion sur le concept et la configuration.

Configuration de la politique linguistique de Cisco (CPL)

Cette procédure peut être utilisée pour configurer un ZFW. L'ordre des étapes n'est pas important, mais quelques procédures doivent être effectuées dans l'ordre. Par exemple, vous devez configurer une carte-classe avant d'affecter une carte-classe à une carte-politique. De même, vous ne pouvez pas affecter une carte-politique à une zone-paire jusqu'à ce que vous ayez configuré la politique. Si vous essayez de configurer une section qui se fonde sur une autre partie de la configuration que vous n'avez pas configurée, le routeur répond avec un message d'erreur.

1. Définissez les zones.
2. Définissez des zone-paires.
3. Définissez les cartes-classes qui décrivent le trafic qui doit avoir une politique appliquée tandis qu'il traverse une zone-paire.
4. Définissez les cartes-politiques pour appliquer une action à votre trafic de carte-classe.
5. Appliquez les cartes-politiques aux zones-paires.
6. Affectez les interfaces aux zones.

Configuration des class-maps de la politique de pare-feu selon des zones

Les cartes-classes définissent le trafic que le pare-feu sélectionne pour l'application de la politique. Les cartes-classes de la couche 4 trient le trafic sur la base des critères énumérés ici. Ces critères sont spécifiés à l'aide de la commande **match** dans une carte-classe :

- Groupe d'accès - Un ACL standard, étendu ou nommé peut filtrer le trafic sur la base de l'adresse IP de source et de destination et du port de source et de destination.
- Protocole - Les protocoles de la couche 4 (TCP, UDP et ICMP) et services d'applications tels que HTTP, SMTP, DNS. Tout service réputé ou défini par l'utilisateur connu du mappage d'application du port peut être spécifié.
- Carte-classe - Une carte-classe subordonnée qui fournit des critères de correspondance supplémentaires peut être emboîtée dans une autre carte-classe.
- Not (non) - Le critère *not (non)* spécifie que tout traffic qui ne correspond pas à un service spécifique (protocole), groupe d'accès ou carte-classe subordonnée sera sélectionné pour la carte-classe.

Combinaison des critères de « correspondance » : « Correspondance-N'importe quel » face à « Correspondance-Tout »

Les cartes-classes peuvent appliquer les opérateurs correspondance-n'importe quel ou correspondance-tout pour déterminer comment appliquer les critères de correspondance. Si match-any est spécifié, le trafic doit satisfaire seulement un des critères de correspondance dans la carte-classe. Si match-all est spécifié, le trafic doit correspondre à tous les critères de la carte-classe afin d'appartenir à cette classe particulière.

Des critères de correspondance doivent être appliqués dans l'ordre du plus spécifique au moins spécifique, si le trafic répond à des critères multiples. Par exemple, considérez cette carte-classe :

```
class-map type inspect match-any my-test-cmap
 match protocol http
 match protocol tcp
```

Le trafic HTTP doit rencontrer le protocole de correspondance http d'abord pour s'assurer que le trafic est traité par les capacités spécifiques du service de l'inspection HTTP. Si les lignes de correpondance sont inversées, le trafic trouve la déclaration tcp du protocole correspondant avant de le comparer au protocole de correspondance http, le trafic est classifié simplement comme trafic TCP et inspecté selon les capacités du composant de l'inspection TCP du pare-feu. Il s'agit d'un problème pour certains services tels que FTP, TFTP et divers services multimédias et de signalisation de voix tels que H.323, SIP, Skinny, RTSP et d'autres. Ces services requièrent des capacités d'inspection supplémentaires pour identifier les activités plus complexes de ces services.

Appliquer un ACL comme critère de correspondance

Les cartes-classes peuvent appliquer un ACL en tant qu'un des critères de correspondance pour l'application de la politique. Si le seul critère de correspondance d'une carte-classe est un ACL et que la carte-classe est associée à une carte-politique appliquant l'action d'inspection, le routeur applique l'inspection TCP ou UDP de base pour tout le trafic autorisé par ACL, excepté l'inspection sensible à l'application fournie par ZFW. Ceci inclut (mais ne se limité à) FTP, SIP, Skinny (SCCP), H.323, Sun RPC et TFTP. Si l'inspection spécifique à l'application est disponible et que l'ACL permet le canal primaire ou de contrôle, tout canal secondaire ou média associé au primaire/contrôle est autorisé, indépendamment de l'autorisation de trafic par ACL.

Si une carte-classe applique seulement ACL 101 comme critère de correspondance, un ACL 101 apparaît tel que :

```
access-list 101 permit ip any any
```

Tout trafic est autorisé en direction de la politique de service appliquée à une zone-paire déterminée et le trafic de retour correspondant est permis dans la direction opposée. Par conséquent, ACL doit appliquer la restriction pour limiter le trafic aux types spécifiques désirés. Remarquez que la liste PAM inclut des services d'applications tels que HTTP, NetBIOS, H.323 et DNS. Cependant, malgré la connaissance de PAM de l'utilisation de l'application spécifique d'un port déterminé, le pare-feu applique seulement la capacité spécifique à l'application suffisante pour adapter les conditions bien connues du trafic d'application. Ainsi, le trafic d'application simple tel que telnet, SSH et d'autres applications à canal unique sont inspectés comme TCP, et leurs statistiques sont combinées dans la sortie de commande **show**. Si la visibilité spécifique à l'application dans l'activité réseau est souhaitée, vous devez configurer l'inspection pour des services par nom d'application (configurez le protocole http correspondant, le protocole correspondant telnet, etc.).

Comparez les statistiques disponibles dans la commande **show policy-map type inspect zone-pair** de cette configuration avec la politique de pare-feu plus explicite montrée plus loin dans la page. Cette configuration est utilisée pour inspecter le trafic d'un téléphone IP Cisco, aussi bien que plusieurs postes de travail qui utilisent une grande variété de trafic, comprenant http, ftp, netbios, ssh et dns :

```
class-map type inspect match-all all-private
 match access-group 101
!
policy-map type inspect priv-pub-pmap
```

```
 class type inspect all-private
  inspect
 class class-default
!
zone security private
zone security public
zone-pair security priv-pub source private destination public
 service-policy type inspect priv-pub-pmap
!
interface FastEthernet4
 ip address 172.16.108.44 255.255.255.0
 zone-member security public
!
interface Vlan1
 ip address 192.168.108.1 255.255.255.0
 zone-member security private
!
access-list 101 permit ip 192.168.108.0 0.0.0.255 any
```

Alors que cette configuration est facile à définir et facilite tout trafic qui provient de la zone privée (tant que le trafic observe les ports de destination standards, reconnus par PAM), elle offre une visibilité limitée dans l'activité de service et ne donne pas l'occasion d'appliquer la bande passante de ZFW et les limites de session pour des types déterminés de trafic. Cette sortie de commande **show policy-map type inspect zone-pair priv-pub** est le résultat de la configuration simple précédente qui utilise seulement une autorisation de tout ACL ip [sous-réseau] entre zone-paires. Comme vous pouvez le voir, la plupart du trafic du poste de travail est compté dans les statistiques TCP ou UDP de base :

```
stg-871-L#show policy-map type insp zone-pair priv-pub Zone-pair: priv-pub Service-policy
inspect : priv-pub-pmap Class-map: all-private (match-all) Match: access-group 101 Inspect
Packet inspection statistics [process switch:fast switch] tcp packets: [413:51589] udp packets:
[74:28] icmp packets: [0:8] ftp packets: [23:0] tftp packets: [3:0] tftp-data packets: [6:28]
skinny packets: [238:0] Session creations since subsystem startup or last reset 39 Current
session counts (estab/half-open/terminating) [3:0:0] Maxever session counts (estab/half-
open/terminating) [3:4:1] Last session created 00:00:20 Last statistic reset never Last session
creation rate 2 Maxever session creation rate 7 Last half-open session total 0 Class-map: class-
default (match-any) Match: any Drop (default action) 0 packets, 0 bytes
```

En revanche, une configuration semblable qui ajoute les classes spécifiques à l'application fournit des statistiques et un contrôle d'application plus granulaires et assure toujours la même largeur des services qui a été montrée dans le premier exemple en définissant la carte-classe de dernière occasion appariant seulement l'ACL comme l'ACL de dernière occasion dans la carte-politique :

```
class-map type inspect match-all all-private
 match access-group 101
class-map type inspect match-all private-ftp
 match protocol ftp
 match access-group 101
class-map type inspect match-any netbios
 match protocol msrpc
 match protocol netbios-dgm
 match protocol netbios-ns
 match protocol netbios-ssn
class-map type inspect match-all private-netbios
 match class-map netbios
 match access-group 101
class-map type inspect match-all private-ssh
 match protocol ssh
 match access-group 101
class-map type inspect match-all private-http
 match protocol http
 match access-group 101
!
```

```
policy-map type inspect priv-pub-pmap
 class type inspect private-http
  inspect
 class type inspect private-ftp
  inspect
 class type inspect private-ssh
  inspect
 class type inspect private-netbios
  inspect
 class type inspect all-private
  inspect
 class class-default!
zone security private
zone security public
zone-pair security priv-pub source private destination public
 service-policy type inspect priv-pub-pmap
!
interface FastEthernet4
 ip address 172.16.108.44 255.255.255.0
 zone-member security public
!
interface Vlan1
 ip address 192.168.108.1 255.255.255.0
 zone-member security private
!
access-list 101 permit ip 192.168.108.0 0.0.0.255 any
```

La configuration plus spécifique fournit cette sortie granulaire substantielle pour la commande **show policy-map type inspect zone-pair priv-pub** :

```
stg-871-L#sh policy-map type insp zone-pair priv-pub Zone-pair: priv-pub Service-policy inspect
: priv-pub-pmap Class-map: private-http (match-all) Match: protocol http Match: access-group 101
Inspect Packet inspection statistics [process switch:fast switch] tcp packets: [0:2193] Session
creations since subsystem startup or last reset 731 Current session counts (estab/half-
open/terminating) [0:0:0] Maxever session counts (estab/half-open/terminating) [0:3:0] Last
session created 00:29:25 Last statistic reset never Last session creation rate 0 Maxever session
creation rate 4 Last half-open session total 0 Class-map: private-ftp (match-all) Match:
protocol ftp Inspect Packet inspection statistics [process switch:fast switch] tcp packets:
[86:167400] ftp packets: [43:0] Session creations since subsystem startup or last reset 7
Current session counts (estab/half-open/terminating) [0:0:0] Maxever session counts (estab/half-
open/terminating) [2:1:1] Last session created 00:42:49 Last statistic reset never Last session
creation rate 0 Maxever session creation rate 4 Last half-open session total 0 Class-map:
private-ssh (match-all) Match: protocol ssh Inspect Packet inspection statistics [process
switch:fast switch] tcp packets: [0:62] Session creations since subsystem startup or last reset
4 Current session counts (estab/half-open/terminating) [0:0:0] Maxever session counts
(estab/half-open/terminating) [1:1:1] Last session created 00:34:18 Last statistic reset never
Last session creation rate 0 Maxever session creation rate 2 Last half-open session total 0
Class-map: private-netbios (match-all) Match: access-group 101 Match: class-map match-any
netbios Match: protocol msrpc 0 packets, 0 bytes 30 second rate 0 bps Match: protocol netbios-
dgm 0 packets, 0 bytes 30 second rate 0 bps Match: protocol netbios-ns 0 packets, 0 bytes 30
second rate 0 bps Match: protocol netbios-ssn 2 packets, 56 bytes 30 second rate 0 bps Inspect
Packet inspection statistics [process switch:fast switch] tcp packets: [0:236] Session creations
since subsystem startup or last reset 2 Current session counts (estab/half-open/terminating)
[0:0:0] Maxever session counts (estab/half-open/terminating) [1:1:1] Last session created
00:31:32 Last statistic reset never Last session creation rate 0 Maxever session creation rate 1
Last half-open session total 0 Class-map: all-private (match-all) Match: access-group 101
Inspect Packet inspection statistics [process switch:fast switch] tcp packets: [51725:158156]
udp packets: [8800:70] tftp packets: [8:0] tftp-data packets: [15:70] skinny packets: [33791:0]
Session creations since subsystem startup or last reset 2759 Current session counts (estab/half-
open/terminating) [2:0:0] Maxever session counts (estab/half-open/terminating) [2:6:1] Last
session created 00:22:21 Last statistic reset never Last session creation rate 0 Maxever session
creation rate 12 Last half-open session total 0 Class-map: class-default (match-any) Match: any
Drop (default action) 4 packets, 112 bytes
```

Un autre avantage d'utiliser une configuration carte-classe et carte-politique plus granulaire,

comme indiqué précédemment, réside en la possibilité d'appliquer des limites à la session et des valeurs de débit spécifiques à la classe ainsi que d'ajuster spécifiquement des paramètres d'inspection en appliquant une carte-paramètre pour ajuster le comportement d'inspection de chaque classe.

Configuration des policy-maps de la politique de pare-feu selon des zones

La carte-politique s'applique à des actions de politique de pare-feu à une ou plusieurs cartes-classes pour définir la politique de service qui sera appliquée à une zone-paire de sécurité. Quand une carte-politique de type inspecter est créée, une classe par défaut nommée classe / classe par défaut est appliquée à l'extrémité de la classe. L'action de la politique par défaut de la classe / classe par défaut est l'abandon, mais peut être changée pour passage. L'option de journal peut être ajoutée avec l'action d'abandon. L'inspection ne peut pas être appliquée à la classe / classe par défaut.

Actions de la politique de pare-feu selon des zones

ZFW fournit trois actions pour le trafic qui traverse d'une zone à l'autre :

- **Abandon** - C'est l'action par défaut pour tout le trafic, telle qu'appliquée par la « classe / classe par défaut » qui termine chaque carte-politique de type inspecter. D'autres cartes-classes dans une carte-politique peuvent également être configurées pour abandonner le trafic non désiré. Le trafic qui est traité par l'action d'abandon est « silencieusement » abandonné (c.-à-d., aucun avis d'abandon n'est envoyé à l'hôte final concerné) par le ZFW, par opposition au comportement d'un ACL d'envoi d'un ICMP « hôte inaccessible » à l'hôte qui a envoyé le trafic refusé. Actuellement, il n'y a pas l'option de changer le comportement d'« abandon silencieux ». L'option journal peut être ajoutée avec l'abandon pour la notification de syslog que le trafic a été abandonné par le pare-feu.
- **Passage** - Cette action laisse le routeur expédier le trafic d'une zone à l'autre. L'action de passage n'assure pas le suivi de l'état des connexions ou sessions dans le trafic. Le passage permet seulement le trafic dans une direction. Une politique correspondante doit être appliquée pour permettre au trafic retourné de passer dans la direction opposée. L'action de passage est utile pour des protocoles tels que IPSec ESP, IPSec AH, ISAKMP et d'autres protocoles sécurisés de façon inhérente dotés d'un comportement prévisible. Cependant, la plupart du trafic d'application est mieux traité dans le ZFW avec l'action inspecter.
- **Inspecter** - L'action inspecter offre le contrôle de trafic selon l'état. Par exemple, si le trafic de la zone privée à la zone Internet dans l'exemple de réseau précédent est inspecté, le routeur met à jour la connexion ou les informations de session pour le trafic TCP et le protocole de datagramme utilisateur (UDP). Par conséquent, le routeur permet le trafic de retour envoyé des serveurs de zone Internet en réponse aux requêtes de connexion de zone privée. En outre, inspecter peut fournir l'inspection et le contrôle d'application pour certains protocoles de service qui pourraient porter le trafic d'application vulnérable ou sensible. La vérification rétrospective peut être appliquée avec une carte-paramètre pour enregistrer le début de la connexion/session, arrêt, durée, volume de données transférées et adresses d'origine et de destination.

Les actions sont associées à des cartes-classes dans des cartes-politiques :

```
conf t
 policy-map type inspect z1-z2-pmap
```

```
  class type inspect service-cmap
   inspect|drop|allow [service-parameter-map]
```

Options d'offre de cartes-paramètres pour modifier les paramètres de connexion pour la politique d'inspection d'une carte-classe déterminée.

Configurer les parameter-maps de la politique de pare-feu selon des zones

Les cartes-paramètres spécifient le comportement d'inspection pour ZFW, pour des paramètres tels que la protection DoS, les temporisateurs de session de la connexion/UDP TCP et les paramètres de journalisation de la vérification rétrospective. Les cartes-paramètres sont également appliquées avec la classe de couche 7 et les cartes-politiques pour définir le comportement spécifique à l'application, des conditions telles que les objets HTTP, conditions d'authentification POP3 et IMAP et toute autre information spécifique à l'application.

Les cartes-paramètres d'inspection pour ZFW sont configurées en tant que **inspecter type**, semblable à l'autre classe ZFW et objets de la politique :

```
stg-871-L(config)#parameter-map type inspect z1-z2-pmap stg-871-L(config-profile)#? parameter-map commands: alert Turn on/off alert audit-trail Turn on/off audit trail dns-timeout Specify timeout for DNS exit Exit from parameter-map icmp Config timeout values for icmp max-incomplete Specify maximum number of incomplete connections before clamping no Negate or set default values of a command one-minute Specify one-minute-sample watermarks for clamping sessions Maximum number of inspect sessions tcp Config timeout values for tcp connections udp Config timeout values for udp flows
```

Les types spécifiques de cartes-paramètres spécifient des paramètres appliqués par les politiques d'inspection de la couche 7. Les cartes-paramètres de type Regex définissent une expression régulière pour l'usage avec l'inspection de l'application HTTP qui filtre le trafic utilisant une expression régulière :

```
parameter-map type regex [parameter-map-name]
```

Les paramètres-cartes de type Protocole-Info définissent des noms d'hôte à utiliser avec l'application de la messagerie instantanée :

```
parameter-map type protocol-info [parameter-map-name]
```

Des détails de la configuration complète pour l'inspection de l'application HTTP et IM sont fournis dans les respectives sections d'inspection d'application de ce document.

L'ajustement de la protection DoS est abordée dans une section ultérieure de ce document.

La configuration de l'inspection de l'application est abordée dans une section ultérieure de ce document.

Application de la journalisation pour les politiques de pare-feu selon des zones

ZFW offre des options de journalisation pour le trafic qui est abandonné ou inspecté par défaut ou des actions de politique de pare-feu configurées. La journalisation de la vérification rétrospective est disponible pour le trafic que le ZFW inspecte. La vérification rétrospective est appliquée en définissant la vérification rétrospective dans une carte-paramètre avec l'action inspecter dans une carte-politique :

```
conf t
 policy-map type inspect z1-z2-pmap
  class type inspect service-cmap
   inspect|drop|allow [parameter-map-name (optional)]
```

La journalisation de l'abandon est disponible pour le trafic que ZFW abandonne. La journalisation de l'abandon est configurée en ajoutant le journal à l'action d'abandon dans une carte-politique :

```
conf t
 policy-map type inspect z1-z2-pmap
  class type inspect service-cmap
   inspect|drop|allow [service-parameter-map]
```

Modification des class-maps et des policy-maps de la politique de pare-feu selon des zones

ZFW n'incorpore actuellement pas un éditeur permettant de modifier les diverses structures ZFW telles que des cartes-politiques, des cartes-classes et des cartes-paramètres. Afin de réordonner des déclarations de correspondance dans une carte-classe ou application d'action pour diverses cartes-classes contenues dans une carte-politique, vous devez suivre ces étapes :

1. Copier la structure existante dans un éditeur de texte tel que Microsoft Windows Notepad ou un éditeur comme les plates-formes Linux/Unix.
2. Supprimer la structure existante de la configuration du routeur.
3. Modifier la structure dans votre éditeur de texte.
4. Copier la structure de nouveau dans la CLI du routeur.

Exemples de configuration

Cet exemple de configuration utilise un routeur à services intégrés Cisco 1811. Une configuration de base avec une connectivité IP, configuration VLAN et un pontage transparent entre deux segments privés de réseau Ethernet est disponible dans l'annexe A. Le routeur est divisé en cinq zones :

- L'Internet public est connecté à FastEthernet 0 (zone d'Internet)
- Deux serveurs Internet sont connectés à FastEthernet 1 (zone DMZ)
- Le commutateur Ethernet est configuré avec deux VLAN :Des postes de travail sont connectés à VLAN1 (zone client).Des serveurs sont connectés à VLAN2 (zone de serveur).Les zones du client et du serveur sont sur le même sous-réseau. Un pare-feu transparent sera appliqué entre les zones, afin que les politiques interzonales sur ces deux interfaces affectent seulement le trafic entre le client et les zones du serveur.
- Les interfaces VLAN1 et VLAN2 communiquent avec d'autres réseaux par le pont d'interface virtuelle (BVI1). Cette interface est affectée à la zone privée. (Voir la figure 2.)**Figure 2 : Détail de topologie de zone**

Ces politiques sont appliquées, en utilisant les zones de réseau définies plus tôt :

- Les hôtes dans la zone Internet peuvent atteindre les services DNS, SMTP et SSH sur un serveur dans le DMZ. L'autre serveur offrira des services SMTP, HTTP et HTTPS. La politique de pare-feu limitera l'accès aux services spécifiques disponibles sur chaque hôte.
- Les serveurs DMZ ne peuvent se connecter aux serveurs dans aucune autre zone.
- Les serveurs dans la zone client peuvent se connecter aux hôtes dans la zone de serveur sur tous les services TCP, UDP et ICMP.
- Les serveurs dans la zone de serveur ne peuvent pas se connecter à des hôtes dans la zone client, excepté un serveur d'application basé sur Unix pouvant ouvrir des sessions de client X Windows pour des serveurs X Windows sur les PC de bureau dans la zone client sur les ports

6900 à 6910.

- Tous les hôtes dans la zone privée (combinaison des clients et des serveurs) peuvent accéder à des hôtes dans le DMZ sur des services SSH, FTP, POP, IMAP, ESMTP et HTTP et, dans la zone Internet, sur des services HTTP, HTTPS et DNS ainsi qu'ICMP. En outre, l'inspection de l'application sera appliquée à des connexions HTTP de la zone privée à la zone Internet afin de s'assurer que la messagerie instantanée et les applications P2P prises en charge ne sont pas acheminées sur le port 80. (Voir la figure 3.)**Figure 3 : Autorisations de service de zone-paire à appliquer dans l'exemple de configuration**

Ces politiques de pare-feu sont configurées par ordre de complexité :

1. Inspection des clients-serveurs TCP/UDP/ICMP
2. Inspection privée-DMZ SSH/FTP/POP/IMAP/ESMTP/HTTP
3. Internet - Inspection DMZ SMTP/HTTP/DNS limitée par l'adresse de hôte
4. Inspection des serveurs-clients X Windows avec un service déterminé de mappage d'application de port (PAM)
5. Internet-privé HTTP/HTTPS/DNS/ICMP avec inspection de l'application HTTP

Puisque vous appliquerez des parties de la configuration à différents segments du réseau à différentes périodes, il est important de se rappeler qu'un segment de réseau perdra la connectivité avec d'autres segments en étant placé dans une zone. Par exemple, quand la zone privée est configurée, les serveurs dans la zone privée perdront la connectivité avec les zones du DMZ et d'Internet jusqu'à ce que leurs politiques respectives soient définies.

Pare-feu de routage d'inspection dynamique

Configurez la politique de l'Internet privé

La figure 4 montre la configuration de la politique privée d'l'Internet.

Figure 4 : Service d'inspection depuis une zone privée vers une zone Internet

La politique de l'Internet privé s'applique à l'inspection de la couche 4 pour HTTP, HTTPS, DNS et l'inspection de la couche 4 pour ICMP de la zone privée à la zone Internet. Ceci permet des connexions de la zone privée à la zone Internet et permet le trafic de retour. L'inspection de la couche 7 présente les avantages d'un contrôle d'application plus strict, d'une plus grande sécurité et de la prise en charge d'applications nécessitant une mise au point. Cependant, l'inspection de la couche 7, comme mentionné, requiert une meilleure compréhension de l'activité de réseau, dans la mesure où les protocoles de la couche 7 qui ne sont pas configurés pour l'inspection ne seront pas autorisés entre les zones.

1. Définissez les cartes-classes qui décrivent le trafic que vous voulez autoriser entre les zones, selon des politiques décrites plus tôt :`conf t`
```
class-map type inspect match-any internet-traffic-class
 match protocol http
 match protocol https
 match protocol dns
 match protocol icmp
```
2. Configurez une carte-politique pour inspecter le trafic sur les cartes-classes que vous venez de définir :`conf t`
```
policy-map type inspect private-internet-policy
 class type inspect internet-traffic-class
  inspect
```
3. Configurez les zones privées et d'Internet et affectez les interfaces du routeur à leurs zones

respectives :
```
conf t
zone security private
zone security internet
int bvi1
zone-member security private
int fastethernet 0
zone-member security internet
```

4. Configurez la zone-paire et appliquez la carte-politique appropriée.**Remarque:** Vous devez seulement configurer la zone paire privée d'Internet à ce stade afin d'inspecter les connexions originaires de la zone privée voyageant vers la zone Internet :
```
conf t
 zone-pair security private-internet source private destination internet
  service-policy type inspect private-internet-policy
```
Ceci achève la configuration de la politique d'inspection de la couche 7 sur la zone-paire privée d'Internet pour permettre aux connexions HTTP, HTTPS, DNS et ICMP de la zone de clients à la zone de serveurs et d'appliquer l'inspection de l'application au trafic HTTP pour s'assurer que le passage du trafic non désiré n'est pas permis sur le port de service HTTP, TCP 80.

Configurez la politique privée de DMZ

La figure 5 montre la configuration de la politique privée de DMZ.

Figure 5 : Inspection de service d'une zone privée vers une zone DMZ

La politique privée de DMZ ajoute de la complexité parce qu'elle requiert une meilleure compréhension du trafic du réseau entre les zones. Cette politique s'applique à l'inspection de la couche 7 de la zone privée au DMZ. Ceci permet des connexions de la zone privée au DMZ et permet le trafic de retour. L'inspection de la couche 7 présente les avantages d'un contrôle d'application plus strict, d'une plus grande sécurité et de la prise en charge d'applications nécessistant une mise au point. Cependant, l'inspection de la couche 7, comme mentionné, requiert une meilleure compréhension de l'activité de réseau, dans la mesure où les protocoles de la couche 7 qui ne sont pas configurés pour l'inspection ne seront pas autorisés entre les zones.

1. Définissez les cartes-classes qui décrivent le trafic que vous voulez autoriser entre les zones, selon des politiques décrites plus tôt :
```
conf t
 class-map type inspect match-any L7-inspect-class
  match protocol ssh
  match protocol ftp
  match protocol pop
  match protocol imap
  match protocol esmtp
  match protocol http
```
2. Configurez des cartes-politiques pour inspecter le trafic sur les cartes-classes que vous venez de définir :
```
conf t
 policy-map type inspect private-dmz-policy
  class type inspect L7-inspect-class
   inspect
```
3. Configurez les zones privées et DMZ et affectez les interfaces du routeur à leurs zones respectives :
```
conf t
zone security private
zone security dmz
int bvi1
zone-member security private
int fastethernet 1
zone-member security dmz
```
4. Configurez la zone-paire et appliquez la carte-politique appropriée.**Remarque:** Vous devez

seulement configurer la zone paire privée de DMZ à ce stade afin d'inspecter les connexions originaires de la zone privée voyageant vers la zone DMZ :`conf t`
```
zone-pair security private-dmz source private destination dmz
 service-policy type inspect private-dmz-policy
```Ceci achève la configuration de la politique d'inspection de la couche 7 sur le DMZ privé pour permettre toutes les connexions TCP, UDP et ICMP de la zone de clients à la zone de serveurs. La politique n'applique pas la mise au point pour les canaux subordonnés, mais fournit un exemple de politique simple pour faciliter la plupart des connexions d'application.

### Configurez la politique Internet DMZ

La figure 6 montre la configuration de la politique Internet DMZ.

### Figure 6 : Inspection de service de la zone Internet à la zone DMZ

Cette politique s'applique à l'inspection de la couche 7 de la zone Internet au DMZ. Ceci permet des connexions de la zone Internet au DMZ et permet le trafic de retour des hôtes DMZ aux hôtes Internet qui ont lancé la connexion. La politique Internet DMZ combine l'inspection de la couche 7 avec des groupes d'adresses définis par les ACL pour restreindre l'accès aux services spécifiques sur les hôtes spécifiques, groupes d'hôtes ou sous-réseaux. Ceci est réalisé en emboîtant une carte-classe spécifiant des services dans une autre carte-classe mettant en référence un ACL pour spécifier des adresses IP.

1. Définissez les cartes-classes et ACL qui décrivent le trafic que vous voulez autoriser entre les zones, selon les politiques décrites plus tôt.Des cartes-classes multiples pour des services doivent être utilisées, car des politiques d'accès différentes seront appliquées pour l'accès à deux serveurs différents. Des connexions DNS et HTTP à 172.16.2.2 sont concédées aux hôtes Internet et des connexions SMTP sont concédées à 172.16.2.3. Notez la différence entre les cartes-classes. Les cartes-classes spécifiant des services utilisent le mot clé **match-any** pour permettre l'un des services énumérés. Les cartes-classes associant les ACL aux cartes-classes de service utilisent le mot-clé **match-all** pour exiger que les deux conditions dans la carte-classe soient remplies pour permettre le trafic :`conf t`
```
access-list 110 permit ip any host 172.16.2.2
access-list 111 permit ip any host 172.16.2.3
class-map type inspect match-any dns-http-class
 match protocol dns
 match protocol http
class-map type inspect match-any smtp-class
 match protocol smtp
class-map type inspect match-all dns-http-acl-class
 match access-group 110
 match class-map dns-http-class
class-map type inspect match-all smtp-acl-class
 match access-group 111
 match class-map smtp-class
```
2. Configurez des cartes-politiques pour inspecter le trafic sur les cartes-classes que vous venez de définir :`conf t`
```
policy-map type inspect internet-dmz-policy
 class type inspect dns-http-acl-class
  inspect
 class type inspect smtp-acl-class
  inspect
```
3. Configurez les zones Internet et DMZ et affectez les interfaces du routeur à leurs zones respectives. Sautez la configuration DMZ si vous l'avez établie dans la section précédente

```
:conf t
zone security internet
zone security dmz
int fastethernet 0
 zone-member security internet
int fastethernet 1
 zone-member security dmz
```

4. Configurez la zone-paire et appliquez la carte-politique appropriée.**Remarque:** Vous avez seulement besoin de configurer la zone paire Internet de DMZ à ce stade afin d'inspecter les connexions originaires de la zone Internet voyageant vers la zone DMZ :`conf t`
   `zone-pair security internet-dmz source internet destination dmz`
   `service-policy type inspect internet-dmz-policy`Ceci achève la configuration de la politique d'inspection spécifique à l'adresse de la couche 7 sur la zone-paire DMZ d'Internet.

Pare-feu transparent d'inspection dynamique

Configurez la politique serveurs-clients

La figure 7 montre la politique de configuration du serveur-client.

Figure 7 : Inspection de service d'une zone de serveurs vers une zone clients

La politique de serveurs-clients applique l'inspection utilisant un service défini par l'utilisateur. L'inspection de la couche 7 est appliquée de la zone de serveurs à la zone de clients. Ceci permet les connexions de X Windows à une plage de ports spécifique de la zone de serveurs vers la zone de clients et permet le trafic de retour. X Windows n'est pas un protocole pris en charge à l'origine dans PAM, ainsi un service configuré de l'utilisateur dans PAM doit être défini afin que ZFW puisse reconnaître et inspecter le trafic approprié.

Deux interfaces du routeur ou plus sont configurées dans le groupe de pontage IEEE pour fournir l'Integrated Routing and Bridging (IRB) assurant le pontage entre les interfaces dans le groupe de pontage et le routage vers d'autres sous-réseaux par l'intermédiaire de l'interface virtuelle du pont (BVI). La politique de pare-feu transparent offrira d'appliquer l'inspection de pare-feu pour le trafic « traversant le pont », mais pas pour le trafic qui quitte le groupe de pontage par l'intermédiaire du BVI. La politique d'inspection s'applique seulement au trafic qui traverse le groupe de pontage. Par conséquent, dans ce scénario, l'inspection sera seulement appliquée au trafic qui se déplace entre zones de clients et les zones de serveurs, qui sont emboîtées à l'intérieur de la zone privée. La politique appliquée entre la zone privée et les zones publiques et DMZ, entre seulement en jeu quand le trafic sort du groupe de pontage par l'intermédiaire du BVI. Quand le trafic sort par l'intermédiaire du BVI soit des clients soit des zones de serveurs, la politique de pare-feu transparent n'est pas invoquée.

1. Configurez PAM avec une entrée définie par l'utilisateur pour X Windows.Les clients X Windows (où des applications sont accueillies) ouvrent des connexions pour l'information d'affichage aux clients (où l'utilisateur opère) dans une portée commençant au port 6900.Chaque connexion supplémentaire utilise les ports successifs, ainsi si un client affiche 10 sessions différentes sur un hôte, le serveur utilise des ports 6900-6909. Par conséquent, si vous inspectez la plage de ports de 6900 à 6909, les connexions ouvertes aux ports au delà de 6909 échoueront :`conf t`
   `ip port-map user-Xwindows port tcp from 6900 to 6910`
2. Examinez les documents PAM pour des questions PAM supplémentaires ou examinez la documentation de l'inspection du protocole granulaire pour des informations sur les détails

de l'interopérabilité entre PAM et l'inspection dynamique du pare-feu Cisco IOS.

3. Définissez les cartes-classes qui décrivent le trafic que vous voulez autoriser entre les zones, selon des politiques décrites plus tôt :`conf t`
```
 class-map type inspect match-any Xwindows-class
  match protocol user-Xwindows
```
4. Configurez des cartes-politiques pour inspecter le trafic sur les cartes-classes que vous venez de définir :`conf t`
```
 policy-map type inspect servers-clients-policy
  class type inspect Xwindows-class
   inspect
```
5. Configurez les zones de client et de serveur et affectez les interfaces du routeur à leurs zones respectives.Si vous avez configuré ces zones et affecté les interfaces dans la section de la politique de configuration Clients-Serveurs, vous pouvez passer à la définition de la zone-paire. La configuration du pontage IRB est donnée à des fins d'exhaustivité :`conf t`
```
bridge irb
bridge 1 protocol ieee
bridge 1 route ip
zone security clients
zone security servers
 int vlan 1
 bridge-group 1
 zone-member security clients
int vlan 2
 bridge-group 1
 zone-member security servers
```
6. Configurez la zone-paire et appliquez la carte-politique appropriée.**Remarque:** Vous devez seulement configurer les zones paires de serveurs-clients à ce stade afin d'inspecter des connexions originaires de la zone de serveurs voyageant vers la zone de clients :`conf t`
```
 zone-pair security servers-clients source servers destination clients
  service-policy type inspect servers-clients-policy
```
Ceci achève la configuration de la politique d'inspection définie par l'utilisateur dans la zone-paire de serveurs-clients pour permettre des connexions X Windows de la zone de serveur à la zone client.

Configurez la politique Serveurs-Clients

La figure 8 montre la configuration de la politique serveur-client.

Figure 8 : Inspection de service d'une zone de clients vers une zone de serveurs

La politique serveurs-client est moins complexe que les autres. L'inspection de la couche 4 est appliquée de la zone de clients à la zone de serveurs. Ceci permet des connexions de la zone de clients à la zone de serveurs et permet le trafic de retour. L'inspection de la couche 4 présente l'avantage de la simplicité de la configuration du pare-feu, du fait que seulement quelques règles sont nécessaires pour permettre la plupart du trafic. Cependant, l'inspection de la couche 4 comporte également deux inconvénients importants :

- Les applications telles que FTP ou des services de flux des médias négocient fréquemment un canal subordonné supplémentaire du serveur au client. Cette fonctionnalité est habituellement facilitée dans un service qui surveille le dialogue de canal de contrôle et permet le canal subordonné. Cette fonction n'est pas disponible dans l'inspection de la couche 4.
- L'inspection de la couche 4 permet quasiment tout le trafic de la couche d'application. Si l'utilisation du réseau doit être contrôlée afin que seulement quelques applications soient

permises par le pare-feu, un ACL doit être configuré sur le trafic en partance pour limiter les services permis par le pare-feu.

Les deux interfaces du routeur sont configurées dans un groupe de pontage IEEE, ainsi cette politique de pare-feu appliquera l'inspection du pare-feu transparent. Cette politique est appliquée sur deux interfaces dans un groupe de pontage IEEE IP. La politique d'inspection s'applique seulement au trafic qui traverse le groupe de pontage. Ceci explique pourquoi les zones de clients et de serveurs sont emboîtées à l'intérieur de la zone privée.

1. Définissez les cartes-classes qui décrivent le trafic que vous voulez autoriser entre les zones, selon des politiques décrites plus tôt :`conf t`
```
class-map type inspect match-any L4-inspect-class
match protocol tcp
match protocol udp
match protocol icmp
```
2. Configurez des cartes-politiques pour inspecter le trafic sur les cartes-classes que vous venez de définir :`conf t`
```
policy-map type inspect clients-servers-policy
class type inspect L4-inspect-class
 inspect
```
3. Configurez les zones de clients et de serveurs et affectez les interfaces du routeur à leurs zones respectives :`conf t`
```
zone security clients
zone security servers
int vlan 1
zone-member security clients
int vlan 2
zone-member security servers
```
4. Configurez la zone-paire et appliquez la carte-politique appropriée.**Remarque:** Vous devez seulement configurer les zones-paires de clients-serveurs à ce stade, afin d'inspecter des connexions originaires de la zone de clients voyageant vers la zone de serveurs :`conf t`
```
zone-pair security clients-servers source clients destination servers
 service-policy type inspect clients-servers-policy
```
Ceci achève la configuration de l'inspection de la couche 4 pour que la zone-paire de clients-serveurs permette toutes les connexions TCP, UDP, et ICMP de la zone de clients à la zone de serveurs. La politique n'applique pas la mise au point pour les canaux subordonnés, mais fournit un exemple de politique simple pour faciliter la plupart des connexions d'application.

Régulation du débit pour la politique de pare-feu selon des zones

Les réseaux de données bénéficient fréquemment de la capacité à limiter le taux de transmission de types spécifiques de trafic sur le réseau et à limiter l'incidence du trafic de basse priorité sur le trafic d'informations professionnelles plus essentielles. Le logiciel Cisco IOS offre cette capacité avec la régulation de trafic, qui limite le taux nominal et l'éclatement du trafic. Le logiciel Cisco IOS a assuré la régulation du trafic depuis la version 12.1(5)T de Cisco IOS.

La version 12.4(9)T du logiciel Cisco IOS augmente ZFW avec une limite de débit en ajoutant la capacité de contrôler le trafic correspondant aux définitions d'une carte-classe spécifique pendant qu'il traverse le pare-feu d'une zone de sécurité à l'autre. Ceci confère la commodité d'offrir un point de configuration pour décrire le trafic spécifique, appliquer la politique de pare-feu et contrôler la consommation de bande passante de ce trafic. Le contrôle ZFW diffère du contrôle basé sur l'interface parce qu'il amène les actions à transmettre seulement pour la conformité à la politique et l'abandon pour violation de politique. Le contrôle ZFW ne peut pas marquer le trafic pour DSCP.

Le contrôle ZFW peut seulement spécifier l'utilisation de la bande passante en octets/seconde, paquet/seconde et le contrôle du pourcentage de la bande passante n'est pas assuré. Le contrôle ZFW peut être appliqué avec ou sans le contrôle basé sur l'interface. Par conséquent, si des capacités de contrôle supplémentaires sont nécessaires, ces fonctionnalités peuvent être appliquées par le contrôle basé sur l'interface. Si le contrôle basé sur l'interface est utilisé en même temps que le contrôle du pare-feu, assurez-vous que les politiques ne sont pas en conflit.

Configuration du contrôle ZFW

Le contrôle ZFW limite le trafic dans une carte-politique à une valeur de débit définie par l'utilisateur entre 8.000 et 2.000.000.000 bits par seconde, avec une valeur d'éclatement configurable de l'ordre de 1.000 à 512.000.000 octets.

Le contrôle ZFW est configuré par une ligne supplémentaire de configuration dans la carte-politique, qui est appliquée après l'action de politique :

```
policy-map type inspect private-allowed-policy
 class type inspect http-class
  inspect
  police rate [bps rate value <8000-2000000000>] burst [value in bytes <1000-512000000>]
```

Contrôle de session

Le contrôle ZFW a également mis en place le contrôle de session pour limiter le décompte de sessions pour le trafic dans une carte-politique correspondant à une carte-classe. Ceci s'ajoute à la capacité existante pour appliquer la politique de protection DoS par carte-classe. En effet, ceci permet le contrôle granulaire sur le nombre de sessions correspondant à toute carte-classe qui traverse une zone-paire. Si la même carte-classe est utilisée sur de multiples cartes-politiques ou zones-paires, différentes limites de session peuvent être appliquées sur les diverses applications de carte-classe.

Le contrôle de session est appliqué en configurant une carte-paramètre qui contient le volume de session souhaité, ajoutant alors la carte-paramètre à l'action d'inspection appliquée à une carte-classe sous une carte-politique :

```
parameter-map type inspect my-parameters
 sessions maximum [1-2147483647]

policy-map type inspect private-allowed-policy
 class type inspect http-class
  inspect my-parameters
```

Les cartes-paramètres peuvent seulement être appliquées à l'action d'inspection et ne sont pas disponibles pour des actions de passage ou d'abandon.

Le contrôle session ZFW et les activités de contrôle sont visibles avec cette commande

```
show policy-map type inspect zone-pair
```

Inspection d'application

L'inspection d'application met en place une fonction supplémentaire de ZFW. Les politiques d'inspection d'application sont appliquées à la couche 7 du modèle OSI, où les applications de l'utilisateur envoient et reçoivent des messages qui permettent aux applications d'offrir des capacités utiles. Quelques applications pourraient offrir des fonctions non souhaitées ou

vulnérables, ainsi les messages associés à ces fonctions doivent être filtrés pour limiter des activités aux services d'application.

ZFW du logiciel Cisco IOS fournit l'inspection d'application et le contrôle de ces services d'applications :

- HTTP
- SMTP
- POP3
- IMAP
- Sun RPC
- Trafic d'application P2P
- IM applications

L'inspection et le contrôle d'application (AIC) varient en termes de capacité par service. L'inspection HTTP offre le filtrage granulaire sur plusieurs types d'activités d'application, offrant des capacités pour limiter la taille de transfert, les longueurs d'adresse Web et l'activité du navigateur pour imposer la conformité aux normes d'application-comportement et pour limiter les types de contenu qui sont transférés sur le service. AIC pour SMTP peut limiter la longueur du contenu et imposer la conformité au protocole. L'inspection POP3 et IMAP peut aider à assurer que les utilisateurs emploient des mécanismes d'authentification sécurisés pour empêcher la compromission des identifiants de l'utilisateur.

L'inspection d'application est configurée en tant qu'ensemble supplémentaire des cartes-classes et des politiques-cartes spécifiques à l'application, qui sont alors appliquées aux cartes-classes et aux cartes-politiques en définissant la politique de service d'application dans la carte-politique.

Inspection d'application HTTP

L'inspection d'application peut être appliquée au trafic HTTP pour contrôler l'utilisation non souhaitée du port de service HTTP pour d'autres applications telles que IM, partage de fichier P2P et les applications de tunnellisation qui peuvent rediriger des applications autrement soumises au pare-feu à travers le TCP 80.

Configurez une carte-classe d'inspection d'application pour décrire le trafic qui viole le trafic HTTP permis :

```
! configure the actions that are not permitted
class-map type inspect http match-any http-aic-cmap
 match request port-misuse any
 match req-resp protocol-violation
! define actions to be applied to unwanted traffic
policy-map type inspect http http-aic-pmap
 class type insp http http-aic-cmap
  reset
  log
! define class-map for stateful http inspection
class-map type inspect match-any http-cmap
 match protocol http
! define class-map for stateful inspection for other traffic
class-map type inspect match-any other-traffic-cmap
 match protocol smtp
 match protocol dns
 match protocol ftp
! define policy-map, associate class-maps and actions
policy-map type inspect priv-pub-pmap
```

```
 class type inspect http-cmap
  inspect
  service-policy http http-aic-pmap
 class type inspect other-traffic-cmap
  inspect
```

Améliorations d'inspection d'application HTTP

La version 12.4(9)T du logiciel Cisco IOS a apporté des améliorations à l'inspection HTTP des fonctions de ZFW. Le pare-feu Cisco IOS a mis en place l'inspection d'application HTTP introduite dans la version 12.3(14)T du logiciel Cisco IOS. La version 12.4(9)T du logiciel Cisco IOS augmente les capacités existantes en ajoutant :

- Capacité d'autoriser, refuser et contrôler les demandes et les réponses basées sur le nom de l'en-tête et les valeurs d'en-tête. Ceci est utile pour bloquer les demandes et les réponses qui portent des champs d'en-tête vulnérables.
- La capacité à limiter les tailles de différents éléments dans les en-têtes de requête et de réponse HTTP telles que la longueur maximum d'URL, la longueur d'en-tête maximum, le nombre maximal d'en-têtes, la longueur maximum d'en-tête, etc. Ceci est utile pour empêcher des dépassements de mémoire tampon.
- Capacité de bloquer des demandes et des réponses qui portent les en-têtes multiples du même type ; par exemple, une requête avec deux en-têtes de longueur de contenu.
- Capacité à bloquer des demandes et des réponses avec des en-têtes non-ASCII. Ceci est utile pour empêcher les diverses attaques qui emploient des caractères binaires ainsi que non-ASCII pour introduire des vers et tout autre contenu malveillant dans les serveurs Web.
- La capacité à grouper des méthodes HTTP dans des catégories personnalisées par l'utilisateur et la flexibilité de bloquer/autoriser/contrôler chacun des groupes est offerte. Le RFC de HTTP permet un ensemble limité de méthodes HTTP. Certaines des méthodes standards sont considérées peu sûres parce qu'elles peuvent être utilisées pour exploiter des vulnérabilités sur un serveur Web. Plusieurs des méthodes non standards présentent un faible degré de sécurité.
- Méthode pour bloquer les URI spécifiques basée sur une expression régulière configurée par l'utilisateur. Cette fonctionnalité donne à un utilisateur la capacité de bloquer des URI personnalisés et des requêtes.
- Capacité de mystifier des types d'en-tête (en particulier type d'en-tête de serveur) avec les chaînes de caractères personnalisables d'utilisateur. Ceci est utile au cas où un attaquant analyserait des réponses de serveur Web et apprendrait autant d'informations que possible, puis lancerait une attaque exploitant les faiblesses de ce serveur Web particulier.
- Capacité à bloquer ou émettre une alerte sur une connexion HTTP si une ou plusieurs valeurs de paramètre HTTP correspondent à des valeurs saisies par l'utilisateur comme expression régulière. Certains des contextes de valeur possibles HTTP incluent l'en-tête, le corps, nom d'utilisateur, mot de passe, agent utilisateur, la ligne de requête, ligne d'état et les variables CGI décodées.

Les exemples de configuration pour des améliorations d'inspection d'application HTTP supposent un réseau simple :

Le pare-feu regroupe le trafic dans deux classes :

- Trafic HTTP
- Tout autre trafic TCP, UDP et ICMP à canal unique

HTTP est séparé pour permettre l'inspection spécifique sur le trafic Web. Ceci vous permet de configurer le contrôle dans la première section de ce document et l'inspection de l'application HTTP dans la deuxième section. Vous configurerez les cartes-classes et les cartes-politiques spécifiques pour le trafic P2P et IM dans la troisième section de ce document. La connectivité est permise de la zone privée à la zone publique. Aucune connectivité n'est fournie de la zone publique à la zone privée.

Une configuration complète mettant en application la politique initiale est donnée dans l'Annexe C, configuration de base de pare-feu selon une politique par zone pour deux zones.

Configuration des améliorations d'inspection d'application HTTP

L'inspection d'application HTTP (aussi bien que d'autres politiques d'inspection d'application) requiert une configuration plus complexe que la configuration de base de la couche 4 . Vous devez configurer la classification et la politique du trafic de la couche 7 pour identifier le trafic spécifique que vous souhaitez contrôler et appliquer l'action désirée au trafic désirable et indésirable.

L'inspection d'application HTTP (semblable à d'autres types d'inspection d'application) peut seulement être appliquée au trafic HTTP. Donc, vous devez définir des cartes-classes et des cartes-politiques de la couche 7 pour le trafic HTTP spécifique, puis définir une carte-classe de la couche 4 spécifiquement pour HTTP et appliquer la politique de la couche 7 à l'inspection HTTP dans une carte-politique de la couche 4, ainsi :

```
!configure the layer-7 traffic characteristics:
class-map type inspect http match-any http-l7-cmap
 match  req-resp protocol-violation
 match  request body length gt 4096
!
!configure the action to be applied to the traffic
!matching the specific characteristics:
policy-map type inspect http http-l7-pmap
 class type inspect http http-l7-cmap
  reset
  log
!
!define the layer-4 inspection policy
class-map type inspect match-all http-l4-cmap
 match protocol http
!
!associate layer-4 class and layer-7 policy-map
!in the layer-4 policy-map:
policy-map type inspect private-allowed-policy
 class type inspect http-l4-cmap
  inspect
  service-policy http http-l7-pmap
```

Toutes ces caractéristiques d'inspection de trafic d'application HTTP sont définies dans une carte-classe de la couche 7 :

- **Inspection d'en-tête** - Cette commande fournit la capacité d'autoriser/refuser/contrôler des demandes ou des réponses dont l'en-tête correspond à l'expression régulière configurée. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-6-HTTP_HDR_REGEX_MATCHED`*Utilisation des commandes* :`match {request|response|req-resp} header regex <parameter-map-name>` *Exemple de cas*

*d'utilisation*Configurez une politique appfw HTTP pour bloquer une requête ou une réponse dont l'en-tête contient des caractères non-ASCII.`parameter-map type regex non_ascii_regex
   pattern “[^\x00-\x80]”
class-map type inspect http non_ascii_cm
   match req-resp header regex non_ascii_regex
policy-map type inspect http non_ascii_pm
   class type inspect http non_ascii_cm
      reset`

- **Inspection de longueur d'en-tête** - Cette commande contrôle la longueur d'une en-tête de requête ou réponse et applique une action si la longueur dépasse le seuil configuré. L'action est permettre ou réinitialiser. L'ajout de l'action journal entraîne un message syslog :`APPFW-4-HTTP_HEADER_LENGTH`*Utilisation des commandes :*`match {request|response|req-resp} header length gt <bytes>` *Exemple de cas d'utilisation*Configurez une politique appfw http pour bloquer des requêtes et des réponses ayant une longueur d'en-tête plus grande que 4096 octets.`class-map type inspect http hdr_len_cm
   match req-resp header length gt 4096

policy-map type inspect http hdr_len_pm
   class type inspect http hdr_len_cm
      reset`
- **Inspection du nombre d'en-tête** - Cette commande vérifie le numéro de lignes d'en-tête (zones) dans une requête/réponse et applique l'action quand le nombre dépasse le seuil configuré. L'action est permettre ou réinitialiser. L'ajout de l'action journal entraîne un message syslog :`APPFW-6- HTTP_HEADER_COUNT`.*Utilisation des commandes :*`match {request|response|req-resp} header count gt <number>` *Exemple de cas d'utilisation*Configurez une politique appfw http pour bloquer une requête qui a plus de 16 champs d'en-tête.`class-map type inspect http hdr_cnt_cm
   match request header count gt 16

policy-map type inspect http hdr_cnt_pm
   class type inspect http hdr_cnt_cm
      reset`
- **Inspection de champ d'en-tête** - Cette commande confère la capacité d'autoriser/refuser/contrôler des demandes/réponses qui contiennent un champ d'en-tête HTTP et une valeur spécifiques. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-6- HTTP_HDR_FIELD_REGEX_MATCHED`*Utilisation des commandes :*`match {request|response|req-resp} header <header-name>` *Exemple de cas d'utilisation*Configurez une politique d'inspection d'application HTTP pour bloquer le spyware/adware :`parameter-map type regex ref_regex
   pattern “\.delfinproject\.com”
   pattern “\.looksmart\.com”

parameter-map type regex host_regex
   pattern “secure\.keenvalue\.com”
   pattern “\.looksmart\.com”

parameter-map type regex usragnt_regex
   pattern “Peer Points Manager”

class-map type inspect http spy_adwr_cm
   match request header refer regex ref_regex
   match request header host regex host_regex
   match request header user-agent regex usragnt_regex

policy-map type inspect http spy_adwr_pm`

```
   class type inspect http spy_adwr_cm
      reset
```

- **Inspection de longueur de champ d'en-tête** - Cette commande confère la capacité de limiter la longueur d'une ligne de champ d'en-tête. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-6- HTTP_HDR_FIELD_LENGTH`.*Utilisation des commandes :*`match {request|response|req-resp} header <header-name> length gt <bytes>` *Exemple de cas d'utilisation*Configurez une politique appfw http pour bloquer une requête dont la longueur de champ de cookie et d'agent-utilisateur dépasse 256 et 128, respectivement.
```
class-map type inspect http hdrline_len_cm
   match request header cookie length gt 256
   match request header user-agnet length gt 128

policy-map type inspect http hdrline_len_pm
   class type inspect http hdrline_len_cm
      reset
```

- **Inspection de répétition de champ d'en-tête** - Cette commande vérifie si une requête ou réponse a répété des champs d'en-tête. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. Une fois activée, l'action journal entraîne un message syslog :`APPFW-6-HTTP_REPEATED_HDR_FIELDS`.*Utilisation des commandes :*`match {request|response|req-resp} header <header-name>` *Exemple de cas d'utilisation*Configurez une politique appfw http pour bloquer une requête ou une réponse qui a plusieurs longueurs de contenu de lignes d'en-tête. Ceci est l'une des fonctionnalités les plus utiles utilisées pour empêcher la contrebande de session.
```
class-map type inspect http multi_occrns_cm
   match req-resp header content-length count gt 1

policy-map type inspect http multi_occrns_pm
   class type inspect http multi_occrns_cm
      reset
```

- **Inspection de méthode** - Le RFC HTTP permet un ensemble limité de méthodes HTTP. Cependant, même certaines des méthodes standards sont considérées comme peu sûres dans la mesure où certaines méthodes peuvent être utilisées pour exploiter des vulnérabilités sur un serveur Web. Plusieurs des méthodes non standards sont fréquemment utilisées pour une activité malveillante. Ceci nécessite de regrouper les méthodes en diverses catégories et de laisser l'utilisateur choisir l'action pour chaque catégorie. Cette commande confère à l'utilisateur une façon flexible de grouper les méthodes dans diverses catégories telles que des méthodes sûres, méthodes peu sûres, méthodes webdav, méthodes rfc et méthodes étendues. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-6-HTTP_METHOD`.*Utilisation des commandes :*`match request method <method>` *Exemple de cas d'utilisation*Configurez une politique appfw http qui groupe les méthodes HTTP dans trois catégories : sûr, peu sûr et webdav. Ceux-ci sont montrés dans la table. Configurez les actions telles que :toutes les méthodes sûres sont permises sans journaltoutes les méthodes peu sûres sont permises avec le journaltoutes les méthodes webdav sont bloquées avec le journal.`http policy`:
```
class-map type inspect http safe_methods_cm
   match request method get
   match request method head
   match request method option

class-map type inspect http unsafe_methods_cm
```

```
   match request method post
   match request method put
   match request method connect
   match request method trace

class-map type inspect http webdav_methods_cm
   match request method bcopy
   match request method bdelete
   match request method bmove

policy-map type inspect http methods_pm
   class type inspect http safe_methods_cm
      allow
   class type inspect http unsafe_methods_cm
      allow log
      class type inspect http webdav_methods_cm
      reset log
```

- **Inspection URI** - Cette commande confère la capacité d'autoriser/refuser/contrôler des requêtes pour lesquelles les correspondances URI ont configuré une inspection régulière. Ceci donne à l'utilisateur la capacité de bloquer des URL personnalisés et des requêtes. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-6- HTTP_URI_REGEX_MATCHED`*Utilisation des commandes :*`match request uri regex <parameter-map-name>` *Exemple de cas d'utilisation*Configurez une politique appfw http pour bloquer une requête dont l'URI correspond à l'une de ces expressions régulières :. *cmd.exe. *sex. *gambling`parameter-map type regex uri_regex_cm`
```
   pattern “.*cmd.exe”
   pattern “.*sex”
   pattern “.*gambling”

class-map type inspect http uri_check_cm
   match request uri regex uri_regex_cm

policy-map type inspect http uri_check_pm
   class type inspect http uri_check_cm
      reset
```

- **Longueur d'inspection URI** - Cette commande vérifie la longueur de l'URI étant envoyé dans une requête et applique l'action configurée quand la longueur dépasse le seuil configuré. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-6- HTTP_URI_LENGTH`. *Utilisation des commandes :*`match request uri length gt <bytes>` *Exemple de cas d'utilisation*Configurez une politique appfw http pour déclencher une alarme à chaque fois que la longueur URI d'une requête dépasse 3076 octets.`class-map type inspect http uri_len_cm`
```
   match request uri length gt 3076

policy-map type inspect http uri_len_pm
   class type inspect http uri_len_cm
      log
```

- **Inspection d'argument** - Cette commande confère la capacité à autoriser, refuser ou contrôler une requête dont les arguments (paramètres) correspondent à l'inspection régulière configurée. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-6- HTTP_ARG_REGEX_MATCHED`*Utilisation des commandes :*`match request arg regex <parameter-map-name>` *Exemple de cas d'utilisation*Configurez une politique appfw http pour bloquer une requête dont les arguments correspondent à l'une de ces expressions

régulières :. *codered. *attack`parameter-map type regex arg_regex_cm
   pattern “.*codered”
   pattern “.*attack”

class-map type inspect http arg_check_cm
   match request arg regex arg_regex_cm

policy-map type inspect http arg_check_pm
   class type inspect http arg_check_cm
      reset`

- **Longueur d'argument URI** - Cette commande vérifie la longueur des arguments étant envoyés dans une requête et applique l'action configurée quand la longueur dépasse le seuil configuré. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-6- HTTP_ARG_LENGTH`.*Utilisation des commandes* :`match request arg length gt <bytes>` *Exemple de cas d'utilisation*Configurez une politique appfw http pour déclencher une alarme à chaque fois que la longueur d'argument d'une requête dépasse 512 octets.`class-map type inspect http arg_len_cm
   match request arg length gt 512

policy-map type inspect http arg_len_pm
   class type inspect http arg_len_cm
      log`
- **Inspection de corps** - Cette CLI permet à l'utilisateur de spécifier une liste des expressions régulières à apparier au corps de la requête ou de la réponse. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-6-HTTP_BODY_REGEX_MATCHED`*Utilisation des commandes* :`match {request|response|reg-resp} body regex <parameter-map-name>` *Exemple de cas d'utilisation*Configurez un appfw http pour bloquer une réponse dont le corps contient la configuration. `*[*[Aa][Tt][Tt][Aa][Cc][Kk]parameter-map type regex body_regex
   pattern “.*[Aa][Tt][Tt][Aa][Cc][Kk]”

class-map type inspect http body_match_cm
   match response body regex body_regex

policy-map type inspect http body_match_pm
   class type inspect http body_match_cm
      reset`
- **Inspection de longueur de corps (contenu)** - Cette commande vérifie la taille du message étant envoyé à travers la requête ou la réponse. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-4- HTTP_CONTENT_LENGTH`*Utilisation des commandes* :`match {request|response|req-resp} body length lt <bytes> gt <bytes>` *Exemple de cas d'utilisation*Configurez une politique appfw http pour bloquer une session http qui porte un message de plus de 10K octets dans une requête ou une réponse.`class-map type inspect http cont_len_cm
   match req-resp header content-length gt 10240

policy-map type inspect http cont_len_pm
   class type inspect http cont_len_cm
      reset`
- **Inspection de la ligne d'état** - Cette commande permet à l'utilisateur de spécifier la liste des expressions régulières à apparier face à la ligne d'état d'une réponse. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de

la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-6-HTTP_STLINE_REGEX_MATCHED`.*Utilisation des commandes* :`match response status-line regex <class-map-name>` *Exemple de cas d'utilisation*Configurez un appfw http pour programmer une alarme à chaque fois qu'une tentative est faite pour accéder à une page interdite. Une page interdite contient habituellement un code d'état 403 et la ligne d'état est similaire à HTTP/1.0 403 page forbidden\r\n.`parameter-map type regex status_line_regex`
`   pattern “[Hh][Tt][Tt][Pp][/][0-9][.][0-9][ \t]+403”`

```
class-map type inspect http status_line_cm
   match response status-line regex status_line_regex

policy-map type inspect http status_line_pm
   class type inspect http status_line_cm
      log
```

- **Inspection de type-contenu** - Cette commande vérifie si le type de contenu de l'en-tête du message est dans la liste des types de contenus pris en charge. Il vérifie également que le type de contenu de l'en-tête correspond au contenu des données du message ou de la partie de corps d'entité. Si le mot clé **mismatch** est configuré, la commande vérfie le type de contenu du message de réponse vis-à-vis de la valeur du champ acceptée du message de requête. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne le message syslog approprié :`APPFW-4- HTTP_CONT_TYPE_VIOLATION, APPFW-4- HTTP_CONT_TYPE_MISMATCH, APPFW-4- HTTP_CONT_TYPE_UNKNOWN`*Utilisation des commandes* :`match {request|response|req-resp} header content-type [mismatch|unknown|violation]`*Exemple de cas d'utilisation*Configurez une politique appfw http pour bloquer une session HTTP qui porte des demandes et des réponses ayant un type de contenu inconnu.`class-map type inspect http cont_type_cm`
`   match req-resp header content-type unknown`

```
policy-map type inspect http cont_type_pm
   class type inspect http cont_type_cm
      reset
```

- **Inspection d'utilisation incorrecte de port** - Cette commande est utilisée pour empêcher le port http (80) d'être incorrectement utilisé pour d'autres applications telles qu'IM, P2P, tunnellisation, etc. L'action permettre ou réinitialiser peut être appliquée à une requête ou réponse appariant les critères de cartes-classes. L'ajout de l'action journal entraîne le message syslog approprié :`APPFW-4- HTTP_PORT_MISUSE_TYPE_IM APPFW-4-HTTP_PORT_MISUSE_TYPE_P2P APPFW-4-HTTP_PORT_MISUSE_TYPE_TUNNEL`*Utilisation des commandes* :`match request port-misuse {im|p2p|tunneling|any}`*Exemple de cas d'utilisation*Configurez une politique appfw http pour bloquer une session http en train d'être incorrectement utilisée pour une application IM.`class-map type inspect http port_misuse_cm`
`   match request port-misuse im`

```
policy-map type inspect http port_misuse_pm
   class type inspect http port_misuse_cm
      reset
```

- **Inspection http strict** - Cette commande active un contrôle de conformité de protocole strict vis-à-vis de requêtes et réponses HTTP. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-4- HTTP_PROTOCOL_VIOLATION`*Utilisation des commandes* :`match req-resp protocol-violation`*Exemple de cas d'utilisation*Configurez une politique appfw http pour bloquer des requêtes ou des réponses violant RFC 2616 :`class-map`

```
type inspect http proto-viol_cm
   match req-resp protocol-violation

policy-map type inspect http proto-viol_pm
   class type inspect http proto-viol_cm
      reset
```

- **Inspection de codage de transfert** - Cette commande confère la capacité d'autoriser, refuser ou contrôler la requête/réponse dont le type de codage correspond au type configuré. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-6- HTTP_TRANSFER_ENCODING`*Utilisation des commandes* :`match {request|response|req-resp} header transfer-encoding {regex <parameter-map-name> |gzip|deflate|chunked|identity|all}` *Exemple de cas d'utilisation*Configurez une politique appfw http pour bloquer une requête ou réponse qui a un type de codage comprimé.
```
class-map type inspect http trans_encoding_cm
   match req-resp header transfer-encoding type compress

policy-map type inspect http trans_encoding_pm
   class type inspect http trans_encoding_cm
      reset
```
- **Inspection Applet Java** - Cette commande contrôle si une réponse a l'applet Java et applique l'action configurée à la découverte de l'applet. L'action permettre ou réinitialiser peut être appliquée à une requête ou à une réponse correspondant aux critères de la carte-classe. L'ajout de l'action journal entraîne un message syslog :`APPFW-4- HTTP_JAVA_APPLET` *Utilisation des commandes* :`match response body java-applet`*Exemple de cas d'utilisation*Configurez une politique appfw http pour bloquer des applets Java.
```
class-map type inspect http java_applet_cm
   match response body java-applet

policy-map type inspect http java_applet_pm
   class type inspect http java_applet_cm
      reset
```

Assistance ZFW du contrôle de transmission de messages instantanée et d'application pair à pair

Le logiciel Cisco IOS version 12.4(9)T a mis en place l'assistance ZFW pour IM et les applications P2P.

Le logiciel Cisco IOS a mis en place la prise en charge du contrôle de l'application IM dans le logiciel Cisco IOS version 12.4(4)T. La version initiale de ZFW ne prenait pas en charge l'application IM dans l'interface ZFW. Si le contrôle d'application IM était désiré, les utilisateurs n'étaient pas en mesure de migrer vers l'interface de configuration ZFW. Le logiciel Cisco IOS version 12.4(9)T a mis en place la prise en charge de ZFW pour l'inspection IM, prenant en charge Yahoo ! Messenger (YM), MSN Messenger (MSN) et AOL Instant Messenger (AIM).

Le logiciel Cisco IOS version 12.4(9)T est la première version du logiciel Cisco IOS qui offre une prise en charge du pare-feu d'IOS original pour les applications de partage de fichier P2P.

IM et l'inspection P2P offrent tous les deux les politiques de la couche 4 et de la couche 7 pour le trafic d'application. Ceci signifie que ZFW peut fournir l'inspection dynamique de base pour autoriser ou refuser le trafic, aussi bien que le contrôle granulaire de la couche 7 sur des activités spécifiques dans les divers protocoles, de sorte que certaines activités d'application sont permises tandis que d'autres sont refusées.

Inspection d'application P2P et contrôle

SDM 2.2 a mis en place le contrôle de l'application P2P dans la section de sa configuration du pare-feu. SDM a appliqué un Network-Based Application Recognition (NBAR) et la politique QoS pour détecter et contrôler l'activité d'application P2P d'une fréquence de ligne de zéro, bloquant tout le trafic P2P. Ceci a soulevé la question que des utilisateurs de la CLI, attendant que la prise en charge P2P dans la CLI du pare-feu d'IOS, n'étaient pas en mesure de configurer le P2P bloquant dans la CLI à moins qu'ils se soient rendu compte de la configuration NBAR/QoS nécessaire. Le logiciel Cisco IOS version 12.4(9)T a mis en place le contrôle du P2P original dans la CLI de ZFW influant sur NBAR pour détecter l'activité d'application P2P. Cette version du logiciel prend en charge plusieurs protocoles d'application P2P :

- BitTorrent
- eDonkey
- Fast-Track
- Gnutella
- KaZaA / KaZaA2
- WinMX

Les applications P2P sont particulièrement difficiles à détecter, en résultat d'un comportement « saut de port » et d'autres tours pour éviter la détection, aussi bien que des problèmes mis en place par les modifications et les mises à jour fréquentes pour les applications P2P qui modifient les comportements des protocoles. ZFW combine l'inspection dynamique du pare-feu original avec les capacités de reconnaissance de trafic de NBAR pour assurer le contrôle de l'application P2P dans l'interface de configuration CPL de ZFW. NBAR offre deux excellents avantages :

- Reconnaissance d'application sur une base heuristique facultative pour identifier des applications malgré le comportement complexe, difficile à détecter.
- Infrastructure extensible offrant un mécanisme de mise à jour pour rester à la hauteur des mises à jour et des modifications de protocole

Configurer l'inspection P2P

Comme mentionné précédemment, l'inspection P2P et le contrôle offrent une inspection dynamique de la couche 4 et le contrôle d'application de la couche 7.

L'inspection de la couche 4 est configurée de même que d'autres services d'applications, si l'inspection des ports originaux des services d'applications est adéquate :

```
class-map type inspect match-any my-p2p-class
match protocol [bittorrent | edonkey | fasttrack | gnutella | kazaa | kazaa2 | winmx ]
[signature (optional)]
!
policy-map type inspect private-allowed-policy
 class type inspect my-p2p-class
  [drop | inspect | pass]
```

Remarquez la signature supplémentaire du protocole correspondant [nom-service]. En ajoutant l'option de la signature à la fin de la déclaration du protocole de correspondance, les heuristiques NBAR sont appliquées au trafic à la recherche de contrôleurs de niveau dans le trafic qui indiquent l'activité spécifique de l'application P2P. Ceci comprend le saut de port et d'autres changements du comportement de l'application pour éviter la détection de trafic. Ce niveau d'inspection du trafic est établi au prix de l'utilisation accrue du CPU et de la capacité réduite du débit du réseau. Si l'option de signature n'est pas appliquée, l'analyse heuristique basée sur

NBAR ne sera pas appliquée pour détecter le comportement de saut de port et l'utilisation du CPU ne sera pas affectée au même degré.

L'inspection du service original porte l'inconvénient qu'il est incapable de maintenir le contrôle des applications P2P au cas où l'application « sauts » vers une source non standard et le port de destination ou si l'application est mis à jour pour commencer son action sur un numéro de port non reconnu :

Application	Ports originaux (tels que reconnus par 12.4(15)T PAM liste)
bittorrent	TCP 6881-6889
edonkey	TCP 4662
promotion accélérée	TCP 1214
gnutella	UDP 6346-6348 DU TCP 6355,5634 DU TCP 6346-6349
kazaa2	Dépendant de PAM
winmx	TCP 6699

Si vous souhaitez permettre (inspecter) le trafic P2P, vous pourriez devoir fournir une configuration supplémentaire. Quelques applications peuvent utiliser plusieurs réseaux P2P ou mettre en place des comportements spécifiques dont vous pourriez avoir besoin dans votre configuration de pare-feu pour permettre à l'application de fonctionner :

- Les clients BitTorrent communiquent habituellement avec des « traqueurs » (serveurs de répertoire de pair) par l'intermédiaire de l'exécution http ou de certains ports non standards. Ceci est généralement le TCP 6969, mais vous pourriez devoir contrôler le port traqueur spécifique au torrent. Si vous souhaitez permettre BitTorrent, la meilleure méthode pour adapter le port supplémentaire est de configurer le HTTP en tant qu'un des protocoles correspondants et d'ajouter le TCP 6969 au HTTP à l'aide de la commande **ip port-map** : **`ip port-map http port tcp 6969`** Vous devrez définir le HTTP et BitTorrent comme critères de correspondance appliqués dans la carte-classe.
- eDonkey semble lancer les connexions qui sont détectées comme eDonkey et Gnutella.
- L'inspection de KaZaA dépend entièrement de la détection de signature NBAR.

L'inspection de la couche 7 (application) augmente l'inspection de la couche 4 avec la capacité d'identifier et d'appliquer des actions spécifiques au service, telles que de bloquer sélectivement ou de permettre la recherche de fichier, le transfert de fichier et les capacités texte-dialogue en ligne. Capacités spécifiques au service par service.

L'inspection de l'application P2P est semblable à l'inspection de l'application HTTP :

```
!configure the layer-7 traffic characteristics:
class-map type inspect [p2p protocol] match-any p2p-l7-cmap
 match action
!
!configure the action to be applied to the traffic
!matching the specific characteristics:
policy-map type inspect [p2p protocol] p2p-l7-pmap
 class type inspect p2p p2p-l7-cmap
  [ reset | allow ]
  log
!
```

```
!define the layer-4 inspection policy
class-map type inspect match-all p2p-l4-cmap
 match protocol [p2p protocol]
!
!associate layer-4 class and layer-7 policy-map
!in the layer-4 policy-map:
policy-map type inspect private-allowed-policy
 class type inspect p2p-l4-cmap
  [ inspect | drop | pass ]
  service-policy p2p p2p-l7-pmap
```

L'inspection de l'application P2P offre des capacités spécifiques pour un sous-ensemble des applications prises en charge par l'inspection de la couche 4 :

- edonkey
- promotion accélérée
- gnutella
- kazaa2

Chacune de ces applications offre des options variables des critères de correspondance spécifiques à l'application :

edonkey

```
router(config)#class-map type inspect edonkey match-any edonkey-l7-cmap router(config-
cmap)#match ? file-transfer Match file transfer stream flow Flow based QoS parameters search-
file-name Match file name text-chat Match text-chat
```

promotion accélérée

```
router(config)#class-map type inspect fasttrack match-any ftrak-l7-cmap router(config-
cmap)#match ? file-transfer File transfer stream flow Flow based QoS parameters
```

gnutella

```
router(config)#class-map type inspect gnutella match-any gtella-l7-cmap router(config-
cmap)#match ? file-transfer Match file transfer stream flow Flow based QoS parameters
```

kazaa2

```
router(config)#class-map type inspect kazaa2 match-any kazaa2-l7-cmap router(config-cmap)#match
? file-transfer Match file transfer stream flow Flow based QoS parameters
```

De nouvelles définitions ou mises à jour de protocoles P2P existants peuvent être chargées en utilisant la fonctionnalité dynamique de mise à jour pdlm de NBAR. Celle-ci est la commande de configuration pour charger le nouveau PDLM :

ip nbar pdlm *<file-location>*

Le nouveau protocole est disponible dans des commandes de **match protocol**… pour le class type inspect. Si le nouveau protocole P2P a des services (sous-protocoles), la nouvelle couche 7 qui inspecte des types de carte-classe aussi bien que les critères de correspondance de la couche 7, deviennent disponibles.

Inspection d'application IM et contrôle

Le logiciel Cisco IOS version 12.4(4)T a mis en place l'inspection et le contrôle de l'application IM. La prise en charge de l'IM n'a pas été mis en place avec ZFW dans 12.4(6)T, ainsi les utilisateurs ne pouvaient pas appliquer le contrôle IM et le ZFW dans la même politique de pare-feu, car ZFW et des caractéristiques de pare-feu habituelles ne peuvent pas coexister sur une interface donnée.

Le logiciel Cisco IOS version 12.4(9)T prend en charge l'inspection dynamique et le contrôle d'application pour ces services IM :

- AOL Instant Messenger
- MSN Messenger
- Yahoo! Messenger

L'inspection IM varie légèrement de la plupart des services, car l'inspection IM se fonde sur l'accès de contrôle à un groupe spécifique de serveurs pour chaque service donné. Les services IM comptent généralement sur un groupe relativement permanent de serveurs de répertoire, que les clients doivent pouvoir contacter afin d'accéder au service IM. Les applications IM tendent à être très difficiles à contrôler du point de vue du protocole ou du service. La façon la plus efficace de contrôler ces applications est de limiter l'accès aux serveurs IM fixes.

Configurer l'inspection IM

L'inspection IM et le contrôle offrent tous deux l'inspection dynamique de la couche 4 et le contrôle d'application de la couche 7.

L'inspection de la couche 4 est configurée de la même façon que d'autres services d'applications :

```
class-map type inspect match-any my-im-class
match protocol [aol | msnmsgr | ymsgr ]
!
policy-map type inspect private-allowed-policy
 class type inspect my-im-class
  [drop | inspect | pass
```

Les applications IM peuvent entrer en contact avec leurs serveurs sur plusieurs ports pour conserver leur fonctionnalité. Si vous souhaitez permettre un service IM donné en appliquant l'action inspecter, vous pourriez ne pas avoir besoin d'une liste-serveur pour définir l'accès permis aux serveurs du service IM. Cependant, configurer une carte-classe qui spécifie un service IM donné, tel qu'AOL Instant Messenger, et appliquer l'action d'abandon dans la carte-politique associée peut amener le client IM à essayer de localiser un port différent où la connectivité est autorisée pour Internet. Si vous ne voulez pas permettre la connectivité à un service donné ou si vous voulez restreindre la capacité de service IM au texte-dialogue en ligne, vous devez définir une liste de serveur afin que ZFW puisse identifier le trafic associé à l'application IM :

```
!configure the server-list parameter-map:
parameter-map type protocol-info <name>
  server name <name>
  server ip a.b.c.d
  server ip range a.b.c.d a.b.c.d
```

Par exemple, la liste de serveur IM de Yahoo est définie en tant que telle :

```
parameter-map type protocol-info ymsgr-pmap
  server name scs.msg.yahoo.com
  server name scsd.msg.yahoo.com
  server ip 66.77.88.99
  server ip range 103.24.5.67 103.24.5.99
```

Vous devrez appliquer la liste-serveur à la définition du protocole :

```
class-map type inspect match-any ym-l4-cmap
 match protocol ymsgr ymsgr-pmap
```

Vous devez configurer les commandes **ip domain lookup** et **ip name-server ip.ad.re.ss** afin de permettre la résolution de noms.

Les noms de serveur IM sont assez dynamiques. Vous devrez vérifier régulièrement que vos listes de serveur IM configurées sont complètes et correctes.

L'inspection de la couche 7 (application) augmente l'inspection de la couche 4 avec la capacité d'identifier et d'appliquer des actions spécifiques au service, telles que de bloquer sélectivement ou de permettre les fonctionnalités de texte-dialogue en ligne, tout en refusant d'autres fonctionnalités de service.

L'inspection de l'application IM offre actuellement la capacité de différencier entre l'activité de texte-dialogue en ligne et tous les autres services d'application. Afin de restreindre l'activité IM au texte-dialogue en ligne, configurez la politique de la couche 7 :

```
class-map type inspect ymsgr match-any ymsgr-text-cmap
  match service text-chat

class-map type inspect ymsgr match-any ymsgr-default-cmap
  match service any

policy-map type inspect im ymsgr-l7-pmap
  class type inspect im ymsgr-text-cmap
    allow
    [log]
  class type inspect im ymsgr-text-cmap
    reset
    [log]
```

Appliquez la politique de la couche 7 à Yahoo ! Politique de Messenger configurée plus tôt :

```
class-map type inspect match-any my-im-class
match protocol ymsgr
!
policy-map type inspect private-allowed-policy
 class type inspect my-im-class
  inspect
  service-policy im ymsgr-l7-pmap
```

Filtrage des URL

ZFW offre des fonctionnalités de filtrage URL pour limiter l'accès au contenu Web à celui spécifié par une liste blanche ou noire définie sur le routeur ou en transmettant des noms de domaines vers un serveur de filtrage URL pour vérifier l'accès aux domaines spécifiques. Le filtrage URL de ZFW dans le logiciel Cisco IOS versions 12.4(6)T à 12.4(15)T est appliqué comme action supplémentaire de politique, semblable à l'inspection de l'application.

Pour le Filtrage URL basé sur le serveur, vous devez définir une carte-paramètre qui décrit la configuration du serveur **urlfilter** :

```
parameter-map type urlfilter websense-parmap
 server vendor [n2h2 | websense] 10.1.1.1
```

Si des listes statiques blanches ou noires sont préférées, vous pouvez définir une liste des domaines ou des sous-domaines qui sont spécifiquement permis ou refusés, alors que l'action inverse est appliquée au trafic qui ne correspond pas à la liste :

```
parameter-map type urlfilter websense-parmap
 exclusive-domain deny .disallowed.com
 exclusive-domain permit .cisco.com
```

Si une liste noire d'URL est définie en utilisant des options de refus dans les définitions de domaine exclusif, tous les autres domaines seront autorisés. Si toute définition « autoriser » est

déterminée, tous les domaines qui seront autorisés doivent être explicitement spécifiés, de la même façon que la fonction des listes de contrôle d'accès d'IP.

Établir une carte-classe qui correspond au trafic HTTP :

```
class-map type inspect match-any http-cmap
 match protocol http
```

Définissez une carte-politique qui associe votre carte-classe avec les actions **inspecter** et **urlfilter** :

```
policy-map type inspect http-filter-pmap
 class type inspect http-cmap
  inspect
  urlfilter websense-parmap
```

Ceci configure la condition minimale de communiquer avec un serveur de filtrage URL. Plusieurs options sont disponibles pour définir le comportement supplémentaire de filtrage URL.

Certains déploiements de réseau pourraient vouloir appliquer le filtrage URL pour quelques serveurs ou sous-réseaux, tout en sautant le filtrage URL pour d'autres hôtes. Par exemple, sur la figure 9, tous les serveurs dans la zone privée doivent faire contrôler le trafic HTTP par un serveur de filtre URL, excepté pour l'hôte spécifique 192.168.1.101.

Figure 9 : Exemple de topologie de filtrage URL

Ceci peut être réalisé en définissant deux cartes de carte-classe différentes :

- Une carte-classe qui apparie seulement le trafic HTTP pour le plus grand group des hôtes, qui recevront le filtrage URL.
- Une carte-classe pour le plus petit groupe des serveurs, qui ne recevront pas le filtrage URL. La deuxième carte-classe correspondra au trafic HTTP, aussi bien qu'à une liste des serveurs qui seront exemptés de la politique de filtrage URL.

Les deux cartes-classes sont configurées dans une carte-politique, mais recevra seulement l'action **urlfilter** :

```
class-map type inspect match-any http-cmap
 match protocol http
class-map type inspect match-all http-no-urlf-cmap
 match protocol http
 match access-group 101
!
policy-map type inspect http-filter-pmap
 class type inspect http-no-urlf-cmap
  inspect
 class type inspect http-cmap
  inspect
  urlfilter websense-parmap
!
access-list 101 permit ip 192.168.1.101 any
```

Contrôle de l'accès au routeur

La plupart des ingénieurs de sécurité réseau ne sont pas à l'aise d'exposer les interfaces de gestion du routeur (par exemple, SSH, Telnet, HTTP, HTTPS, SNMP, etc.) à l'Internet public, et dans certaines circonstances, le contrôle pourrait être nécessaire pour l'accès de LAN au routeur également. Le logiciel Cisco IOS offre un certain nombre d'options pour limiter l'accès aux diverses interfaces, qui comprennent la fonctionnalité familiale Network Foundation Protection (NFP), divers mécanismes de contrôle d'accès pour les interfaces de gestion et l'auto-zone de ZFW. Vous devriez passer en revue d'autres fonctionnalités, telles que le contrôle d'accès VTY, la

protection plane de gestion et le contrôle d'accès SNMP pour déterminer quelle combinaison des fonctionnalités de contrôle du routeur fonctionnera le mieux pour votre application spécifique.

Généralement, la famille de fonctionnalité NFP est plus adaptée pour le contrôle du trafic destiné au routeur lui-même. Reportez-vous à l'Aperçu de la sécurité du plan de contrôle dans le logiciel Cisco IOS pour des informations décrivant la protection du routeur utilisant des fonctionnalités NFP.

Si vous décidez d'appliquer ZFW au trafic de contrôle en direction et en provenance d'adresses IP sur le routeur lui-même, vous devez comprendre que la politique par défaut du pare-feu et les fonctionnalités diffèrent de celles disponibles pour le trafic de transit. Le trafic de transit est défini comme trafic sur le réseau dont les adresses IP sources et de destination ne correspondent à aucune des adresses IP appliquées à toute interface des routeurs et le trafic n'amènera pas le routeur à envoyer, par exemple, des messages de contrôle de réseau comme l'expiration de TTL d'ICMP ou des messages d'inaccessibilité réseau/hôte.

ZFW applique une politique par défaut de refuser-tout au trafic se déplaçant entre les zones, excepté comme mentionné dans les règles générales, au trafic dans n'importe quelle zone s'écoulant directement vers les adresses des interfaces du routeur qui est implicitement autorisé. Ceci s'assure que la connectivité aux interfaces de gestion du routeur est conservée quand une configuration de pare-feu de zone est appliquée au routeur. Si la même politique refuser tout affectait la connectivité directement sur le routeur, une configuration de politique complète de gestion devrait être appliquée avant que des zones soient configurées sur le routeur. Ceci perturberait vraisemblablement la connectivité de gestion si la politique était incorrectement mise en application ou appliquée dans un ordre incorrect.

Quand une interface est configurée pour être un membre de zone, les serveurs connectés à l'interface sont inclus dans la zone. Cependant, le trafic s'écoulant en direction et en provenance de ces adresses IP des interfaces du routeur n'est pas contrôlé par les politiques de zone (à l'exception des circonstances décrites dans la note après la figure 10). Au lieu de cela, toutes les interfaces d'IP sur le routeur sont automatiquement rattachées à la zone individuelle quand ZFW est configuré. Afin de contrôler le trafic IP se déplaçant vers les interfaces du routeur des diverses zones sur un routeur, les politiques doivent être appliquées pour bloquer ou autoriser/inspecter le trafic entre la zone et la zone individuelle du routeur, et vice versa. (Voir la figure 10.)

Figure 10 : Appliquez la politique entre les zones de réseau et la zone individuelle du routeur

Bien que le routeur offre une politique d'autorisation par défaut entre toutes les zones et la zone individulle, si une politique est configurée de n'importe quelle zone vers la zone individuelle et qu'aucune politique n'est configurée des zones individuelles aux zones connectées de l'interface et configurables par l'utilisateur du routeur, tout le trafic engendré par le routeur rencontre la zone connectée à la politique de zone individuelle à son retour sur le routeur et est bloqué. Ainsi, le trafic engendré par le routeur doit être inspecté pour permettre son retour à la zone individuelle.

Remarque: Le logiciel Cisco IOS utilise toujours l'adresse IP associée aux hôtes de destination « les plus proches » de l'interface pour le trafic tel que syslog, tftp, telnet et d'autres services de plan de contrôle et soumet ce trafic à la politique de pare-feu de la zone individuelle. Cependant, si un service définit une interface spécifique comme source-interface utilisant les commandes qui incluent, mais non limité au **logging source-interface [nombre de type]**, à l'**ip tftp source-interface [nombre de type]**, et à l'**ip telnet source-interface [nombre de type]**, le trafic est soumis à l'auto-zone.

Remarque: Quelques services (en particulier des services voix-sur-IP de routeurs) utilisent les interfaces éphémères ou non configurables qui ne peuvent pas être affectées aux zones de

sécurité. Ces services pourraient ne pas fonctionner correctement si leur trafic ne pouvait pas être associé à une zone de sécurité configurée.

Limitations de la politique de zone individuelle

La politique de zone individuelle a une fonctionnalité limité en comparaison aux politiques disponibles pour les zones paires de trafic en transit :

- Comme cela était le cas pour l'inspection dynamique, le trafic engendré par routeur se limite à l'inspection du protocole complexe TCP, UDP, ICMP pour H.323.
- L'inspection de l'application n'est pas disponible pour les politiques de zone individuelle.
- La limitation de session et de débit ne peut pas être configurée dans les politiques de zone individuelle.

Configuration de la politique de zone individuelle

Dans la plupart des cas, il s'agit de politiques d'accès souhaitables pour les services de gestion des routeurs :

- La connectivité Telnet refuser tout, comme le protocole de libellé de Telnet expose facilement les identifiants et autres informations confidentielles de l'utilisateur.
- Permettez les connexions SSH de n'importe quel utilisateur dans n'importe quelle zone. SSH encrypte les identifiants de l'utilisateur et les données de session, ce qui assure la protection contre les utilisateurs malveillants qui emploient des outils de saisie de paquet pour surveiller l'activité de l'utilisateur et compromettre les identifiants ou les informations confidentielles telles que la configuration du routeur. SSH version 2 fournit une protection plus forte et des vulnérabilités d'adresses spécifiques inhérentes au SSH version 1.
- Permettez la connectivité de HTTP au routeur des zones privées, si la zone privée est digne de confiance. Autrement, si la zone privée héberge le potentiel pour que les utilisateurs malveillants compromettent l'information, le HTTP n'utilise pas le cryptage pour protéger le trafic de gestion, et pourrait révéler des informations confidentielles telles que identifiants de l'utilisateur ou la configuration.
- Permettez la connectivité HTTPS de n'importe quelle zone. Semblable au SSH, HTTPS encrypte les données de session et les identifiants de l'utilisateur.
- Limitez l'accès restreint SNMP à un hôte spécifique ou sous-réseau. SNMP peut être utilisé pour modifier la configuration du routeur et révéler les informations de configuration. SNMP devrait être configuré avec le contrôle d'accès sur les diverses communautés.
- Bloquez les requêtes ICMP de l'Internet public à la zone privée d'adresse (assumant que l'adresse de zone privée est routable). Une ou plusieurs adresses publiques peuvent être exposées pour le trafic ICMP pour dépanner le réseau, s'il y a lieu. Plusieurs attaques d'ICMP peuvent être utilisées pour accabler les ressources du routeur ou pour reconnaître la topologie du réseau et l'architecture.

Un routeur peut appliquer ce type de politique en plus de deux zones-paires pour chaque zone qui doit être contrôlée. Chaque zone-paire pour traffic interne depuis, ou en partance vers, la zone individuelle du routeur doit être appariée avec la politique respective dans la direction opposée, à moins que le trafic ne soit pas provenu de la direction opposée. Une carte-politique peut être appliquée à chacune des zones-paires interne et externe, décrivant tout le trafic ou des cartes-politiques spécifiques par zone-paire peuvent être appliqués. Configuration des zones-paires spécifiques par carte-politique fournit la granularité pour afficher l'activité appariant chaque carte-

politique.

Assumant un exemple de réseau avec la station de gestion SNMP à 172.17.100.11 et un serveur TFTP à 172.17.100.17, cette sortie fournit un exemple de la politique d'accès entière de l'interface de gestion :

```
class-map type inspect match-any self—service-cmap
 match protocol tcp
 match protocol udp
 match protocol icmp
 match protocol h323
!
class-map type inspect match-all to-self-cmap
 match class-map self—service-cmap
 match access-group 120
!
class-map type inspect match-all from-self-cmap
 match class-map self—service-cmap
!
class-map type inspect match-all tftp-in-cmap
 match access-group 121
!
class-map type inspect match-all tftp-out-cmap
 match access-group 122
!
policy-map type inspect to-self-pmap
 class type inspect to-self-cmap
  inspect
 class type inspect tftp-in-cmap
  pass
!
policy-map type inspect from-self-pmap
 class type inspect from-self-cmap
  inspect
 class type inspect tftp-out-cmap
  pass
!
zone security private
zone security internet
zone-pair security priv-self source private destination self
 service-policy type inspect to-self-pmap
zone-pair security net-self source internet destination self
 service-policy type inspect to-self-pmap
zone-pair security self-priv source self destination private
 service-policy type inspect from-self-pmap
zone-pair security self-net source self destination internet
 service-policy type inspect from-self-pmap

!
interface FastEthernet 0/0
 ip address 172.16.100.10
 zone-member security internet
!
interface FastEthernet 0/1
 ip address 172.17.100.10
 zone-member security private
!
access-list 120 permit icmp 172.17.100.0 0.0.0.255 any
access-list 120 permit icmp any host 172.17.100.10 echo
access-list 120 deny icmp any any
access-list 120 permit tcp 172.17.100.0 0.0.0.255 host 172.17.100.10 eq www
access-list 120 permit tcp any any eq 443
```

```
access-list 120 permit tcp any any eq 22
access-list 120 permit udp any host 172.17.100.10 eq snmp
access-list 121 permit udp host 172.17.100.17 host 172.17.100.10
access-list 122 permit udp host 172.17.100.10 host 172.17.100.17
```

Malheureusement, la politique de zone indivuelle n'offre pas la capacité d'inspecter des transferts TFTP. Ainsi, le pare-feu doit passer tout le traffic en direction et en provenance du serveur TFTP si TFTP doit passer par le pare-feu.

Si le routeur termine des connexions VPN IPSec, vous devriez également définir une politique pour passer ESP IPSec, AH, ISAKMP et NAT-T IPSec (UDP 4500). Ceci dépend de celui qui est nécessaire selon les services que vous utiliserez. La politique suivante peut être appliquée en plus de la politique ci-dessus. Remarquez le changement des cartes-politiques où une carte-classe pour le trafic VPN a été insérée avec une action de passage. Généralement, le trafic encrypté est digne de confiance, à moins que votre politique de sécurité énonce que vous devez laisser le trafic encrypté en direction et en provenance de points finaux déterminés.

```
class-map type inspect match-all crypto-cmap
 match access-group 123
!
policy-map type inspect to-self-pmap
 class type inspect crypto-cmap
  pass
 class type inspect to-self-cmap
  inspect
 class type inspect tftp-in-cmap
  pass
!
policy-map type inspect from-self-pmap
 class type inspect crypto-cmap
  pass
 class type inspect from-self-cmap
  inspect
 class type inspect tftp-out-cmap
  pass
!
access-list 123 permit esp any any
access-list 123 permit udp any any eq 4500
access-list 123 permit ah any any
access-list 123 permit udp any any eq 500
```

Pare-feu selon des zones et services d'application de vaste domaine

Reportez-vous aux Notes de publication pour les Services d'application de réseau étendu Cisco (logiciel version 4.0.13) - Nouvelles fonctionnalités du logiciel version 4.0.13 pour des notes d'application fournissant des exemples de configuration et des conseils d'utilisation

Surveillance de la politique de pare-feu selon des zones à l'aide des commandes show et debug

ZFW met en place de nouvelles commandes afin de visualiser la politique de configuration et contrôler l'activité de pare-feu.

Affichez la description de zone et les interfaces contenues dans une zone spécifique :

show zone security *[<zone-name>]*

Quand le nom de la zone n'est pas compris, la commande affiche l'information de toutes les zones configurées.

```
Router#show zone security z1 zone z1 Description: this is test zone1 Member Interfaces:
Ethernet0/0
```

Affichez la zone source, la zone de destination et la politique liée à la zone paire :

show zone-pair security [source <*source-zone-name*>**] [destination** <*destination-zone-name*>**]**

Quand aucune source ou destination n'est spécifiée, toutes les zones-paires avec source, destination et politique associée sont affichées. Quand seulement la zone source/de destination est mentionnée, toutes les zones paires qui contiennent cette zone en tant que source/destination sont affichées.

```
Router#show zone-pair security zone-pair name zp Source-Zone z1 Destination-Zone z2 service-
policy p1
```

Affiche une carte-politique déterminée :

show policy-map type inspect [<*policy-map-name*> **[class** <*class-map-name*>**]]**

Quand le nom d'une carte-politique n'est pas spécifiée, il affiche toutes les cartes-politiques de type inspecter (y compris la carte-politique de la couche 7 qui contient un sous-type).

```
Router#show policy-map type inspect p1 Policy Map type inspect p1 Class c1 Inspect
```

Affiche le délai d'exécution des statistiques du type d'inspection de la carte-politique existant dans une zone-paire spécifique.

show policy-map type inspect zone-pair [*zone-pair-name***] [sessions]**

Quand **aucun nom de zone-paire** est mentionné, des cartes-politiques sur toutes les zones-paires sont affichées.

L'option **sessions** affiche les sessions d'inspection créées par l'application de carte-politique sur la zone-paire spécifique.

```
Router#show policy-map type inspect zone-pair zp Zone-pair: zp Service-policy : p1 Class-map: c1
(match-all) Match: protocol tcp Inspect Session creations since subsystem startup or last reset
0 Current session counts (estab/half-open/terminating) [0:0:0] Maxever session counts
(estab/half-open/terminating) [0:0:0] Last session created never Last statistic reset never Last
session creation rate 0 Last half-open session total 0 Class-map: c2 (match-all) Match: protocol
udp Pass 0 packets, 0 bytes Class-map: class-default (match-any) Match: any Drop 0 packets, 0
bytes
```

Le mot-clé **urlfilter** affiche les statistiques liées à l'urlfilter qui concernent la carte-politique spécifiée (ou des cartes politiques sur toutes les cibles quand aucun nom de zone-paire n'est spécifié) :

show policy-map type inspect zone-pair [*zone-pair-name***] [urlfilter [cache]]**

Quand le mot clé **cache** est spécifié avec **urlfilter**, il affiche le cache d'**urlfilter** (d'adresse IP).

Résumé de la commande **show policy-map** pour inspecter les cartes-politiques :

show policy-map type inspect inspect { <policy name> [class <class name>] | zone-pair [<zone-pair name>] [sessions | urlfilter cache] }

Ajustement de la protection de déni de service de la politique de pare-feu selon des .zones

ZFW offre la protection DoS pour alerter des ingénieurs réseau aux changements excessifs de l'activité réseau, et pour atténuer l'activité non désirée pour réduire l'incidence des modifications d'activité réseau. ZFW met à jour un compteur distinct pour chaque carte-classe de carte-politique. Ainsi, si une carte-classe est utilisée pour deux cartes-politiques de zones-paires différentes, deux ensembles différents de compteurs de protection DoS seront appliqués.

ZFW assure la réduction d'attaque DoS par défaut sur des versions du logiciel Cisco IOS antérieures à 12.4(11)T. Le comportement de protection DoS par défaut a changé avec le logiciel Cisco IOS version 12.4(11)T. Reportez-vous à l'Ajustement de la protection du déni de service du pare-feu Cisco IOS pour plus de détails et une procédure pour ajuster la protection DoS ZFW.

Reportez-vous à la Définir des politiques pour protéger contre des attaques de déni de service SYN TCP pour plus d'informations sur les attaques DoS SYN TCP.

Annexe

Annexe A : Configuration de base

```
ip subnet-zero
ip cef
!
bridge irb
!
interface FastEthernet0
 ip address 172.16.1.88 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet1
 ip address 172.16.2.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet2
 switchport access vlan 2
!
interface FastEthernet3
 switchport access vlan 2
!
interface FastEthernet4
 switchport access vlan 1
!
interface FastEthernet5
 switchport access vlan 1
!
interface FastEthernet6
 switchport access vlan 1
!
interface FastEthernet7
 switchport access vlan 1
!
interface Vlan1
 no ip address
```

```
 bridge-group 1
!
interface Vlan2
 no ip address
 bridge-group 1
!
interface BVI1
 ip address 192.168.1.254 255.255.255.0
 ip route-cache flow
!
ip classless
ip route 0.0.0.0 0.0.0.0 172.16.1.1
!
bridge 1 protocol ieee
bridge 1 route ip
!
end
```

Annexe B : Configuration finale (complète)

```
ip subnet-zero
ip cef
!
ip port-map user-Xwindows port tcp from 6900 to 6910
!
class-map type inspect match-any L4-inspect-class
 match protocol tcp
 match protocol udp
 match protocol icmp
class-map type inspect match-any L7-inspect-class
  match protocol ssh
  match protocol ftp
  match protocol pop
  match protocol imap
  match protocol esmtp
  match protocol http
class-map type inspect match-any dns-http-class
 match protocol dns
 match protocol http
class-map type inspect match-any smtp-class
 match protocol smtp
class-map type inspect match-all dns-http-acl-class
 match access-group 110
 match class-map dns-http-class
class-map type inspect match-all smtp-acl-class
 match access-group 111
 match class-map smtp-class
class-map type inspect match-any Xwindows-class
 match protocol user-Xwindows
class-map type inspect match-any internet-traffic-class
 match protocol http
 match protocol https
 match protocol dns
 match protocol icmp
class-map type inspect http match-any bad-http-class
 match  port-misuse all
 match  strict-http
!
policy-map type inspect clients-servers-policy
 class type inspect L4-inspect-class
  inspect
policy-map type inspect private-dmz-policy
  class type inspect L7-inspect-class
   inspect
```

```
policy-map type inspect internet-dmz-policy
 class type inspect dns-http-acl-class
  inspect
 class type inspect smtp-acl-class
  inspect
policy-map type inspect servers-clients-policy
 class type inspect Xwindows-class
  inspect
policy-map type inspect private-internet-policy
  class type inspect internet-traffic-class
   inspect
  class type inspect bad-http-class
   drop
!
zone security clients
zone security servers
zone security private
zone security internet
zone security dmz
zone-pair security private-internet source private destination internet
  service-policy type inspect private-internet-policy
zone-pair security servers-clients source servers destination clients
  service-policy type inspect servers-clients-policy
zone-pair security clients-servers source clients destination servers
  service-policy type inspect clients-servers-policy
zone-pair security private-dmz source private destination dmz
  service-policy type inspect private-dmz-policy
zone-pair security internet-dmz source internet destination dmz
  service-policy type inspect internet-dmz-policy
!
bridge irb
!
interface FastEthernet0
 ip address 172.16.1.88 255.255.255.0
 zone-member internet
!
interface FastEthernet1
 ip address 172.16.2.1 255.255.255.0
 zone-member dmz
!
interface FastEthernet2
 switchport access vlan 2
!
interface FastEthernet3
 switchport access vlan 2
!
interface FastEthernet4
 switchport access vlan 1
!
interface FastEthernet5
 switchport access vlan 1
!
interface FastEthernet6
 switchport access vlan 1
!
interface FastEthernet7
 switchport access vlan 1
!
interface Vlan1
 no ip address
 zone-member clients
 bridge-group 1
!
interface Vlan2
```

```
 no ip address
 zone-member servers
 bridge-group 1
!
interface BVI1
 ip address 192.168.1.254 255.255.255.0
 zone-member private
!
ip classless
ip route 0.0.0.0 0.0.0.0 172.16.1.1
!
access-list 110 permit ip any host 172.16.2.2 access-list 111 permit ip any host 172.16.2.3 !
bridge 1 protocol ieee bridge 1 route ip ! End
```

Annexe C : Configuration de la politique de pare-feu selon des zones pour deux zones

Ceci exemple fournit une configuration simple comme base pour le test de caractéristique pour des améliorations du ZFW du logiciel Cisco IOS. Cette configuration est une configuration modèle pour deux zones, telle que configurée sur un routeur 1811. La zone privée est appliquée aux ports du commutateur fixe du routeur, ainsi tous les serveurs sur les ports de commutation sont connectés au VLAN 1. La zone publique est appliquée sur FastEthernet 0.

```
class-map type inspect match-any private-allowed-class
 match protocol tcp
 match protocol udp
 match protocol icmp
class-map type inspect match-all http-class
 match protocol http
!
policy-map type inspect private-allowed-policy
 class type inspect http-class
  inspect my-parameters class type inspect private-allowed-class inspect ! zone security private
zone security public zone-pair security priv-pub source private destination public service-
policy type inspect private-allowed-policy ! interface fastethernet 0 zone-member security
public ! Interface VLAN 1 zone-member security private
```

Informations connexes

- Support et documentation techniques - Cisco Systems