Politique de service

Les politiques de service qui utilisent Modular Policy Framework offrent un moyen cohérent et flexible de configurer les fonctionnalités de l'ASA. Par exemple, vous pouvez utiliser une politique de service pour créer une configuration de délai d'expiration qui est spécifique à une application TCP particulière, par opposition à une configuration qui s'applique à toutes les applications TCP. Une politique de service comprend plusieurs actions ou règles appliquées à une interface ou appliquées globalement.

À propos des politiques de service

Les rubriques suivantes décrivent le fonctionnement des politiques de service.

Les composants d'une politique de service

Les politiques de service permettent d'appliquer des services avancés au trafic que vous autorisez. Tout trafic autorisé par les règles d'accès peut faire l'objet de politiques de service et donc recevoir un traitement spécial, comme être redirigé vers un module de service ou faire l'objet d'une inspection d'application.

Vous pouvez avoir les types de politiques de service suivants :

- Une politique globale qui s'applique à toutes les interfaces.
- Une politique de service appliquée par interface. La politique peut être un ensemble de cartes pour le trafic passant par le périphérique et le trafic de gestion dirigé vers l'interface de l'ASA plutôt que de passer par elle.

Chaque politique de service est composée des éléments suivants :

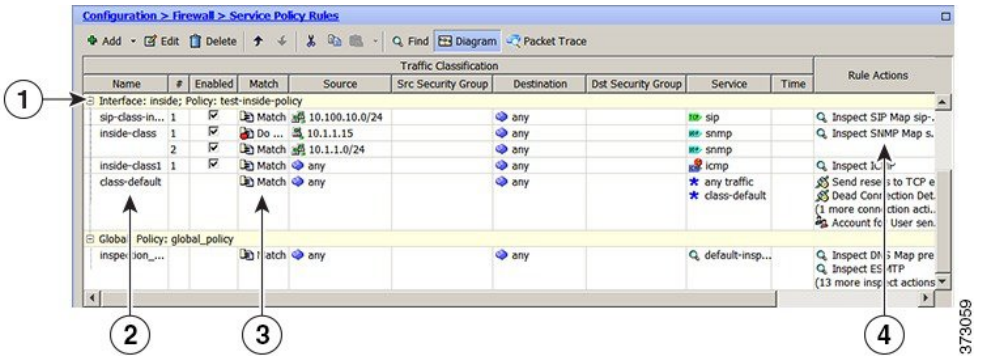
1. La liste des politiques de service, qui est l'ensemble ordonné de règles, et est nommée dans la commande **service-policy**. Dans ASDM, la liste des politiques est représentée sous forme de dossier sur la page des règles de politique de service.

2. Des règles, chaque règle étant une commande **class** dans la liste des politiques de service et les commandes associées à la commande **class**. Dans ASDM, chaque règle est affichée sur une ligne distincte, et le nom de la règle est le nom de la carte.

   La commande **class** définit les critères de correspondance de trafic pour la règle.

   Les commandes associées à la carte, telles que **inspect**, **set connection timeout**, etc., définissent les services et les contraintes à appliquer au trafic correspondant. Notez que les commandes d'inspection peuvent pointer vers des listes des politiques d'inspection, qui définissent les actions à appliquer au trafic inspecté. Gardez à l'esprit que les listes des politiques d'inspection ne sont pas la même chose que les listes des politiques de service.

L'exemple suivant compare la façon dont les politiques de service s'affichent dans l'interface de ligne de commande avec la façon dont elles s'affichent dans ASDM. Notez qu'il n'y a pas de mappage un à un entre les appels de figure et les lignes dans l'interface de ligne de commande.



L'interface de ligne de commande suivante est générée par les règles affichées dans la figure ci-dessus.

```
: Access lists used in class maps.
: In ASDM, these map to call-out 3, from the Match to the Time fields.
access-list inside_mpc line 1 extended permit tcp 10.100.10.0 255.255.255.0 any eq sip
access-list inside_mpc_1 line 1 extended deny udp host 10.1.1.15 any eq snmp
access-list inside_mpc_1 line 2 extended permit udp 10.1.1.0 255.255.255.0 any eq snmp
access-list inside_mpc_2 line 1 extended permit icmp any any
: SNMP map for SNMP inspection. Denies all but v3.
: In ASDM, this maps to call-out 4, rule actions, for the class-inside policy.
snmp-map snmp-v3only
  deny version 1
  deny version 2
  deny version 2c
: Inspection policy map to define SIP behavior.
: The sip-high inspection policy map must be referred to by an inspect sip command
: in the service policy map.
: In ASDM, this maps to call-out 4, rule actions, for the sip-class-inside policy.
policy-map type inspect sip sip-high
  parameters
    rtp-conformance enforce-payloadtype
    no traffic-non-sip
    software-version action mask log
    uri-non-sip action mask log
    state-checking action drop-connection log
    max-forwards-validation action drop log
    strict-header-validation action drop log
: Class map to define traffic matching for the inside-class rule.
: In ASDM, this maps to call-out 3, from the Match to the Time fields.
class-map inside-class
```

```
  match access-list inside_mpc_1
: Class map to define traffic matching for the sip-class-inside rule.
: In ASDM, this maps to call-out 3, from the Match to the Time fields.
class-map sip-class-inside
  match access-list inside_mpc
: Class map to define traffic matching for the inside-class1 rule.
: In ASDM, this maps to call-out 3, from the Match to the Time fields.
class-map inside-class1
  match access-list inside_mpc_2
: Policy map that actually defines the service policy rule set named test-inside-policy.
: In ASDM, this corresponds to the folder at call-out 1.
policy-map test-inside-policy
: First rule in test-inside-policy, named sip-class-inside. Inspects SIP traffic.
: The sip-class-inside rule applies the sip-high inspection policy map to SIP inspection.
: In ASDM, each rule corresponds to call-out 2.
  class sip-class-inside
    inspect sip sip-high
: Second rule, inside-class. Applies SNMP inspection using an SNMP map.
  class inside-class
    inspect snmp snmp-v3only
: Third rule, inside-class1. Applies ICMP inspection.
  class inside-class1
    inspect icmp
: Fourth rule, class-default. Applies connection settings and enables user statistics.
  class class-default
    set connection timeout embryonic 0:00:30 half-closed 0:10:00 idle 1:00:00
reset dcd 0:15:00 5
    user-statistics accounting
: The service-policy command applies the policy map rule set to the inside interface.
: This command activates the policies.
service-policy test-inside-policy interface inside
```

Fonctionnalités configurées avec les politiques de service

Le tableau suivant répertorie les fonctionnalités que vous configurez à l'aide des politiques de service.

Tableau 1 : Fonctionnalités configurées avec les politiques de service

Fonctionnalités	Pour le trafic de transit?	Pour le trafic de gestion?	Voir :
inspection des applications (plusieurs types)	**Tous** sauf la comptabilité RADIUS	**Comptabilité RADIUS** uniquement	• Premiers pas avec l'inspection du protocole de la couche application. • Inspection des protocoles Internet de base. • Inspection des protocoles vocaux et vidéo. • Inspection des réseaux mobiles.
Filtrage de journalisation des événements sécurisés NetFlow	**Oui**	**Oui**	Consultez le guide de mise en œuvre de NetFlow.
Régulation d'entrée et de sortie QoS	**Oui**	Non	Qualité de service.
File d'attente prioritaire standard QoS	**Oui**	Non	Qualité de service.

Fonctionnalités	Pour le trafic de transit?	Pour le trafic de gestion?	Voir :
Limites et délais d'expiration de connexion TCP et UDP et répartition aléatoire du numéro de séquence TCP	**Oui**	**Oui**	Paramètres de connexion.
normalisation TCP	**Oui**	Non	Paramètres de connexion.
contournement de l'état du TCP	**Oui**	Non	Paramètres de connexion.
Statistiques d'utilisateurs pour le pare-feu d'identité	**Oui**	**Oui**	Consultez la commande **user-statistics** dans la référence de commande.

Sens d'application des fonctionnalités

Les actions sont appliquées au trafic de manière bidirectionnelle ou unidirectionnelle, selon la fonctionnalité. Pour les fonctionnalités appliquées de manière bidirectionnelle, tout le trafic qui entre ou sort de l'interface à laquelle vous appliquez la liste des politiques est affecté si le trafic correspond à la carte de trafic dans les deux sens.

Remarque Lorsque vous utilisez une politique globale, toutes les fonctionnalités sont unidirectionnelles ; les fonctionnalités qui sont normalement bidirectionnelles lorsqu'elles sont appliquées à une seule interface ne s'appliquent qu'au trafic entrant de chaque interface lorsqu'elles sont appliquées globalement. Comme la politique est appliquée à toutes les interfaces, la politique sera appliquée dans les deux sens, de sorte que la bidirectionnalité dans ce cas est redondante.

Pour les fonctionnalités appliquées de manière unidirectionnelle, par exemple la file d'attente de priorité QoS, seul le trafic qui entre (ou quitte, selon la fonctionnalité) l'interface à laquelle vous appliquez la liste des politiques est affecté. Consultez le tableau suivant pour connaître le sens d'application de chaque fonctionnalité.

Tableau 2 : Sens d'application des fonctionnalités

Fonctionnalités	Sens d'application sur une interface unique	Sens d'application global
Inspection des applications (plusieurs types)	Bidirectionnel	Entrée
Filtrage NetFlow Secure Event Logging	S. O.	Entrée
Régulation du trafic entrant QoS	Entrée	Entrée
Régulation du trafic sortant QoS	Sortie	Sortie
File d'attente prioritaire standard QoS	Sortie	Sortie
Limites et délais d'expiration de connexion TCP et UDP et répartition aléatoire du numéro de séquence TCP	Bidirectionnel	Entrée

Fonctionnalités	Sens d'application sur une interface unique	Sens d'application global
normalisationTCP	Bidirectionnel	Entrée
contournement de l'état du TCP	Bidirectionnel	Entrée
Statistiques d'utilisateurs pour le pare-feu d'identité	Bidirectionnel	Entrée

Correspondance des fonctionnalités dans une politique de service

Un paquet correspond de mappage de classe dans une mise en correspondance de politiques pour une interface donnée selon les règles suivantes :

1. Un paquet ne peut correspondre qu'à une seule carte de classe dans la et chaque type de fonctionnalité.
2. Lorsque le paquet correspond à une de carte de trafic pour un type de fonctionnalité, l'ASA ne tente pas de le mettre en correspondance avec les de cartes de trafic ultérieures pour ce type de fonctionnalité.
3. Si le paquet correspond à une de carte de trafic ultérieure pour un type de fonctionnalité différent, l'ASA applique également les actions pour la de carte de trafic ultérieure, si elle est prise en charge. Consultez Incompatibilité de certaines actions de fonctionnalité, à la page 6 pour obtenir plus d'informations sur les combinaisons non prises en charge.

Remarque L'inspection d'application comprend plusieurs types d'inspection, et la plupart s'excluent mutuellement. Pour les inspections qui peuvent être combinées, chaque inspection est considérée comme une fonctionnalité distincte.

Exemples de mise en correspondance de paquets

Par exemple :

- Si un paquet correspond à une de carte de trafic pour les limites de connexion, ainsi qu'à une de carte de trafic pour une inspection d'application, les deux actions sont appliquées.
- Si un paquet correspond à une de carte de trafic pour l'inspection HTTP, mais correspond également à une autre de carte de trafic qui inclut l'inspection HTTP, les actions de la deuxième de carte de trafic ne sont pas appliquées.
- Si un paquet correspond à une de carte de trafic pour l'inspection HTTP, mais correspond également à une autre de carte de trafic qui comprend l'inspection FTP, les actions de la deuxième de carte de trafic ne sont pas appliquées, car les inspections HTTP et FTP ne peuvent pas être combinées.
- Si un paquet correspond à une de carte de trafic pour l'inspection HTTP, mais correspond également à une autre de carte de trafic qui inclut l'inspection IPv6, les deux actions sont appliquées, car l'inspection IPv6 peut être combinée avec tout autre type d'inspection.

Ordre dans lequel plusieurs actions de fonctionnalité sont appliquées

L'ordre dans lequel les différents types d'actions dans une politique de service de sont effectués est indépendant de l'ordre dans lequel les actions apparaissent dans le de la liste des politiques.

Les actions sont effectuées dans l'ordre suivant :

1. régulation d'entrée QoS
2. Normalisation TCP, limites et délais d'expiration de connexion TCP et UDP, randomisation du numéro de séquence TCP et contournement de l'état TCP.

Remarque Lorsque l'ASA effectue un service de proxy (comme AAA) ou qu'il modifie la charge utile TCP (comme l'inspection FTP), le normaliseur TCP agit en mode double, où il est appliqué avant et après le service de modification du proxy ou de la charge utile.

3. Inspections d'application qui peuvent être combinées avec d'autres inspections :
    1. IPv6
    2. Options d'adresse IP
    3. WAAS
4. Inspections d'application qui ne peuvent pas être combinées avec d'autres inspections. Consultez Incompatibilité de certaines actions de fonctionnalité, à la page 6 pour de plus amples renseignements.
5. Régulation de sortie QoS
6. File d'attente prioritaire standard QoS

Remarque Le filtrage de la journalisation des événements sécurisés NetFlow et les statistiques d'utilisateur pour le pare-feu d'identité sont indépendants de l'ordre.

Incompatibilité de certaines actions de fonctionnalité

Certaines fonctionnalités ne sont pas compatibles entre elles pour le même trafic. La liste suivante peut ne pas inclure toutes les incompatibilités ; pour en savoir plus sur la compatibilité de chaque fonctionnalité, consultez le chapitre ou la section correspondant à la fonctionnalité :

- Vous ne pouvez pas configurer la mise en file d'attente prioritaire QoS et la régulation QoS pour le même ensemble de trafic.
- La plupart des inspections ne doivent pas être combinées avec une autre inspection, donc l'ASA n'applique qu'une seule inspection si vous configurez plusieurs inspections pour le même trafic. Les exceptions sont répertoriées dans Ordre dans lequel plusieurs actions de fonctionnalité sont appliquées, à la page 6.

Remarque Lacommande **match default-inspection-traffic**, qui est utilisée dans la politique globale par défaut, est un raccourci d'interface de ligne de commande spécial pour faire correspondre les ports par défaut pour toutes les inspections. Lorsqu'elle est utilisée dans une liste des politiques, cette carte de trafic garantit que l'inspection correcte est appliquée à chaque paquet, en fonction du port de destination du trafic. Par exemple, lorsque le trafic UDP pour le port 69 atteint l'ASA, l'ASA applique l'inspection TFTP; lorsque le trafic TCP pour le port 21 arrive, l'ASA applique l'inspection FTP. Ainsi, dans ce cas uniquement, vous pouvez configurer plusieurs inspections pour la même carte de trafic. Normalement, l'ASA n'utilise pas le numéro de port pour déterminer quelle inspection appliquer, vous donnant ainsi la flexibilité d'appliquer des inspections à des ports non standard, par exemple.

Un exemple d'erreur de configuration est si vous configurez plusieurs inspections dans la même liste des politiques et n'utilisez pas le raccourci default-inspection-traffic. Dans le premier exemple, le trafic destiné au port 21 est configuré par erreur pour l'inspection FTP et HTTP. Dans le deuxième exemple, le trafic destiné au port 80 est configuré par erreur pour l'inspection FTP et HTTP. Dans les deux cas d'exemples de configuration incorrecte, seule l'inspection FTP est appliquée, car FTP précède HTTP dans l'ordre des inspections appliquées.

Exemple 1 : Mauvaise configuration pour les paquets FTP : l'inspection HTTP a également été configurée

```
class-map ftp
   match port tcp eq 21
class-map http
   match port tcp eq 21 [it should be 80]
policy-map test
   class ftp
     inspect ftp
   class http
     inspect http
```

Exemple 2 : Mauvaise configuration pour les paquets HTTP : l'inspection FTP a également été configurée

```
class-map ftp
   match port tcp eq 80 [it should be 21]
class-map http
   match port tcp eq 80
policy-map test
   class ftp
     inspect ftp
   class http
     inspect http
```

Mise en correspondance des fonctionnalités pour plusieurs politiques de service

Pour le trafic TCP et UDP (et ICMP lorsque vous activez l'inspection ICMP dynamique), les politiques de service fonctionnent sur les flux de trafic, et pas seulement sur les paquets individuels. Si le trafic fait partie d'une connexion existante qui correspond à une fonctionnalité dans une politique sur une interface, ce flux de trafic ne peut pas également correspondre à la même fonctionnalité dans une politique sur une autre interface; seule la première politique est utilisée.

Par exemple, si le trafic HTTP correspond à une politique sur l'interface interne pour inspecter le trafic HTTP, et que vous disposez d'une politique distincte sur l'interface externe pour l'inspection HTTP, ce trafic n'est pas également inspecté à la sortie de l'interface externe. De même, le trafic de retour pour cette connexion ne sera pas inspecté par la politique d'entrée de l'interface externe ni par la politique de sortie de l'interface interne.

Pour le trafic qui n'est pas traité comme un flux, par exemple ICMP lorsque vous n'activez pas l'inspection ICMP avec état, le trafic de retour peut correspondre à une carte de politiques différente sur l'interface de retour.

Lignes directrices pour les politiques de service

Lignes directrices relatives à l'inspection

Il existe une rubrique distincte qui fournit des lignes directrices détaillées sur les politiques de service d'inspection des applications. Consultez Lignes directrices relatives à l'inspection des applications.

Lignes directrices pour IPv6

Prend en charge IPv6 pour les fonctionnalités suivantes :

- Inspection d'application pour plusieurs protocoles, mais pas tous. Pour de plus amples renseignements, consultez la section Lignes directrices relatives à l'inspection des applications.
- Filtrage NetFlow Secure Event Logging
- Contournement de l'état du SCTP
- Limites et délais d'expiration de connexion TCP et UDP et répartition aléatoire du numéro de séquence TCP
- normalisationTCP
- contournement de l'état du TCP
- Statistiques des utilisateurs pour le pare-feu d'identité

Lignes directrices relatives aux cartes de trafic (classes de trafic)

Le nombre maximal de cartes de trafic (classes de trafic) de tous les types est de 255 en mode unique ou par contexte en mode multiple. Les cartes de trafic comprennent les types suivants :

- Classes de trafic de couche 3/4 (pour le trafic de transit et le trafic de gestion).
- Cartes de trafic d'inspection
- Cartes de trafic d'expressions régulières
- Commandes **match** utilisées directement sous une liste des politiques d'inspection

Cette limite inclut également les cartes de trafics par défaut de tous les types, ce qui limite les cartes de trafics configurées par l'utilisateur à environ 235.

Lignes directrices relatives aux listes des politiques

Consultez les lignes directrices suivantes concernant l'utilisation des listes des politiques :

- Vous ne pouvez appliquer qu'une seule liste de politiques à chaque interface. Pour les pare-feu bas de gamme, vous pouvez créer jusqu'à 64 listes des politiques dans la configuration; pour des pare-feu plus puissants, vous pouvez créer jusqu'à 128 listes des politiques.
- Vous pouvez appliquer la même liste des politiques à plusieurs interfaces.
- Vous pouvez identifier jusqu'à 63 cartes de trafic de couche 3/4 dans une liste des politiques de couche 3/4.
- Pour chaque carte de trafic, vous pouvez attribuer plusieurs actions à partir d'un ou de plusieurs types de fonctionnalités, le cas échéant. Consultez Incompatibilité de certaines actions de fonctionnalité, à la page 6.

Lignes directrices relatives aux politiques de service

- Les politiques de service d'interface sur les interfaces d'entrée prévalent sur la politique de service globale pour une fonctionnalité donnée. Par exemple, si vous avez une politique globale avec inspection FTP et une politique d'interface avec normalisation TCP, l'inspection FTP et la normalisation TCP sont appliquées à l'interface. Toutefois, si vous avez une politique globale avec inspection FTP et une politique d'interface d'entrée avec inspection FTP, seule l'inspection FTP de la politique d'interface d'entrée est appliquée à cette interface. Si aucune politique d'entrée ou globale ne met en œuvre une fonctionnalité, une politique de service d'interface sur l'interface de sortie qui spécifie la fonctionnalité est appliquée.
- Vous ne pouvez appliquer qu'une seule politique globale. Par exemple, vous ne pouvez pas créer une politique globale qui inclut l'ensemble de fonctionnalités 1 et une politique globale distincte qui inclut l'ensemble de fonctionnalités 2. Toutes les fonctionnalités doivent être incluses dans une seule politique.
- Lorsque vous apportez des modifications à la configuration de la politique de service, toutes les *nouvelles* connexions utilisent la nouvelle politique de service. Les connexions existantes continuent d'utiliser la politique qui était configurée au moment de l'établissement de la connexion. La sortie pour la commande **show** n'inclura pas de données sur les anciennes connexions.

  Par exemple, si vous supprimez une politique de service QoS d'une interface, puis ajoutez une version modifiée, alors la commande **show service-policy** n'affiche que les compteurs QoS associés aux nouvelles connexions qui correspondent à la nouvelle politique de service; les connexions existantes sur l'ancienne politique ne s'affichent plus dans la sortie de la commande.

  Pour vous assurer que toutes les connexions utilisent la nouvelle politique, vous devez déconnecter les connexions actuelles afin qu'elles puissent se reconnecter à l'aide de la nouvelle politique. Utilisez les commandes **clear conn** ou **clear local-host**.

Par défaut pour les politiques de service

Les rubriques suivantes décrivent les paramètres par défaut des politiques de service et le cadre de politique modulaire.

Configuration de la politique de service par défaut

Par défaut, la configuration comprend une politique qui correspond à tout le trafic d'inspection d'application par défaut et applique certaines inspections au trafic sur toutes les interfaces (politique globale). Toutes les inspections ne sont pas activées par défaut. Vous ne pouvez appliquer qu'une seule politique globale, donc si vous souhaitez modifier la politique globale, vous devez soit modifier la politique par défaut, soit la désactiver et en appliquer une nouvelle. (Une politique d'interface remplace la politique globale pour une fonctionnalité particulière.)

La politique par défaut inclut les inspections d'application suivantes :

- DNS
- FTP
- H323 (H225)
- H323 (RAS)
- RSH
- RTSP
- ESMTP
- SQLnet
- Skinny (SCCP)
- SunRPC
- SIP
- NetBios
- TFTP
- Options d'adresse IP

Cartes de trafics par défaut (classes de trafic)

La configuration comprend une carte de trafic de couche 3/4 par défaut (classe de trafic) que l'ASA utilise dans la politique globale par défaut appelée default-inspection-traffic ; elle correspond au trafic d'inspection par défaut. Cette classe, qui est utilisée dans la politique globale par défaut, est un raccourci spécial pour faire correspondre les ports par défaut pour toutes les inspections.

Lorsqu'elle est utilisée dans une politique, cette classe garantit que l'inspection correcte est appliquée à chaque paquet, en fonction du port de destination du trafic. Par exemple, lorsque le trafic UDP pour le port 69 atteint l'ASA, l'ASA applique l'inspection TFTP; lorsque le trafic TCP pour le port 21 arrive, l'ASA applique l'inspection FTP. Ainsi, dans ce cas uniquement, vous pouvez configurer plusieurs inspections pour la même carte de trafic. Normalement, l'ASA n'utilise pas le numéro de port pour déterminer quelle inspection appliquer, ce qui vous donne ainsi la possibilité d'appliquer des inspections à des ports non standard, par exemple.

```
class-map inspection_default
 match default-inspection-traffic
```

Une autre carte de trafic qui existe dans la configuration par défaut est appelée class-default, et elle correspond à tout le trafic. Cette carte de trafic apparaît à la fin de toutes les listes des politiques de couche 3/4 et indique essentiellement à l'ASA de n'effectuer aucune action sur tout autre trafic. Si vous utilisez la classe par défaut (la classe Tout trafic dans ASDM), utilisez-la à des fins plus larges comme l'activation de l'aléatoire des séquences TCP, du décrément des TTL ou toute autre modification d'attribut facultatif. Ne l'utilisez pas pour l'inspection, car cela pourrait perturber le trafic. Certaines fonctionnalités sont uniquement disponibles pour class-default.

```
class-map class-default
 match any
```

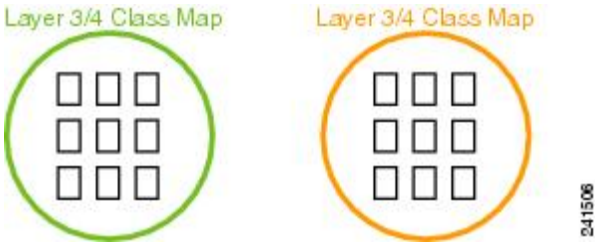
Configurer les politiques de service

Pour configurer les politiques de service à l'aide du cadre de politique modulaire, procédez comme suit :

Procédure

Étape 1 Identifiez le trafic sur lequel vous souhaitez agir en créant des cartes de trafic 3/4, comme décrit dans Identifier le trafic (cartes de trafic de couche 3/4), à la page 13.

Par exemple, vous pourriez vouloir effectuer des actions sur tout le trafic qui passe par l'ASA ; ou vous pourriez ne souhaitez effectuer que certaines actions sur le trafic de 10.1.1.0/24 vers n'importe quelle adresse de destination.



Étape 2 Vous pouvez également effectuer des actions supplémentaires sur une partie du trafic d'inspection.

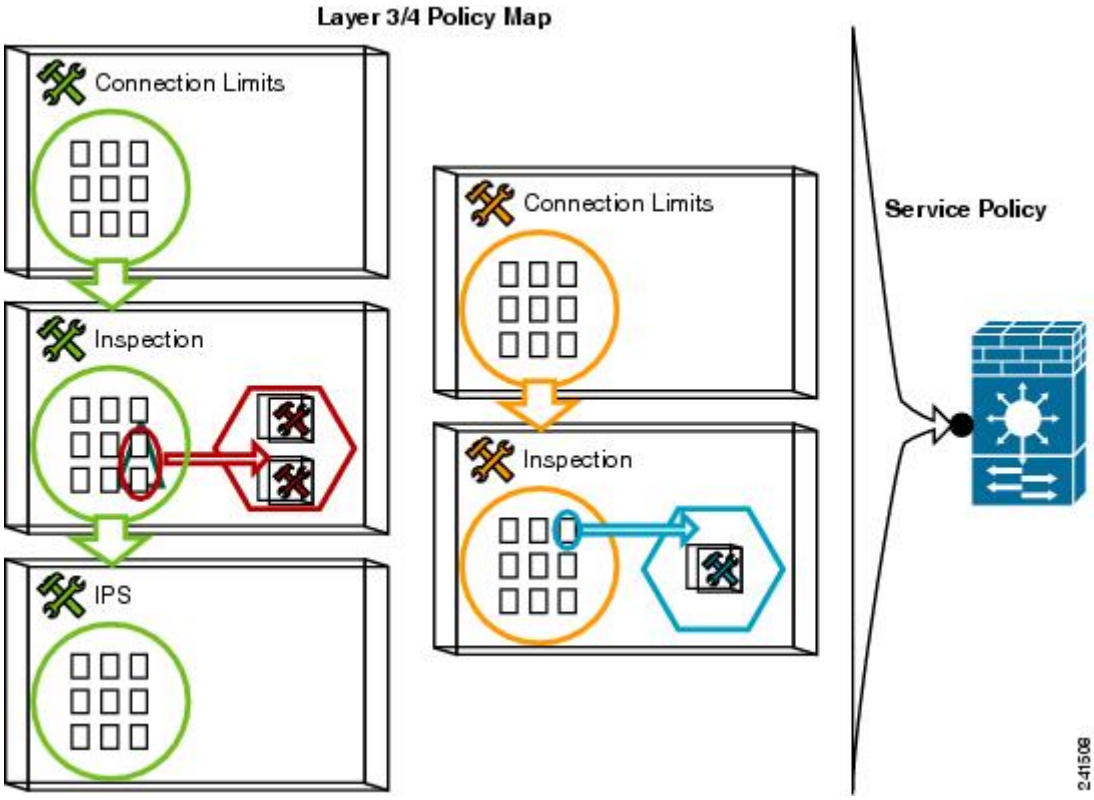
Si l'une des actions que vous souhaitez effectuer est l'inspection des applications et que vous souhaitez effectuer des actions supplémentaires sur une partie du trafic d'inspection, créez une carte de politique d'inspection. La liste des politiques d'inspection identifie le trafic et précise ce qu'il faut en faire.

Par exemple, vous pourriez vouloir abandonner toutes les requêtes HTTP dont la longueur de corps est supérieure à 1 000 octets.

Vous pouvez créer une liste des politiques d'inspection autonome qui identifie le trafic directement à l'aide de commandes **match**, ou vous pouvez créer une carte de classe d'inspection pour la réutilisation ou pour une correspondance plus complexe. Par exemple, vous pouvez mettre en correspondance du texte dans des paquets inspectés à l'aide d'une expression régulière ou d'un groupe d'expressions régulières (une carte de trafic d'expression régulière) et cibler des actions en fonction de critères plus restreints. Par exemple, vous pourriez vouloir abandonner toutes les requêtes HTTP avec une URL comprenant le texte « example.com ».

Consultez Configurer l'inspection du protocole de couche d'application.

Étape 3 Définissez les actions que vous souhaitez effectuer sur chaque carte de classe de couche 3/4 en créant une carte de politiques de couche 3/4, comme décrit dans Définir les actions (liste des politiques de couche 3/4), à la page 16.



Étape 4 Déterminez sur quelles interfaces vous souhaitez appliquer la liste des politiques ou appliquez-la globalement, comme décrit dans Appliquer les actions à une interface (politique de service), à la page 18.

Identifier le trafic (cartes de trafic de couche 3/4)

Une carte de classe de couche 3/4 identifie le trafic de couche 3 et 4 auquel vous souhaitez appliquer des actions. Vous pouvez créer plusieurs cartes de classe de couche 3/4 pour chaque carte de politique de couche 3/4.

Créer une carte de trafic de couche 3/4 pour le trafic de transit

Une carte de trafic de couche 3/4 établit une correspondance du trafic en fonction des protocoles, des ports, des adresses IP et d'autres attributs de couche 3 ou 4.

Astuces Nous vous suggérons d'inspecter le trafic uniquement sur les ports sur lesquels vous attendez un trafic d'application; si vous inspectez tout le trafic, par exemple à l'aide de **match any**, les performances de l'ASA peuvent être affectées.

Procédure

Étape 1 Créer une carte de trafic de couche 3/4 : **class-map** *class_map_name*

Où *class_map_name* est une chaîne d'une longueur maximale de 40 caractères.

Le nom « class-default » est réservé. Tous les types de cartes de trafic utilisent le même espace de nom, de sorte que vous ne pouvez pas réutiliser un nom déjà utilisé par un autre type de carte de trafic. L'interface de ligne de commande passe en mode de configuration class-map.

Exemple :

```
hostname(config)# class-map all_udp
```

Étape 2 (Facultatif) Ajoutez une description à la carte de trafic.

description *string*

Exemple :

```
hostname(config-cmap)# description All UDP traffic
```

Étape 3 Établissez une correspondance du trafic à l'aide de l'une des commandes suivantes. Sauf indication contraire, vous ne pouvez inclure qu'une seule commande **match** dans la carte de trafic.

- **match any (tout)** : établit une correspondance avec tout le trafic.

  ```
  hostname(config-cmap)# match any
  ```

- **match access-list** *access_list_name* : établit une correspondance avec le trafic spécifié par une ACL étendue.

  ```
  hostname(config-cmap)# match access-list udp
  ```

- **match port** {**tcp** | **udp** | **sctp**} {**eq** *port_num* | **range** *port_num port_num*} : correspond aux ports de destination, qu'il s'agisse d'un port unique ou d'une plage de ports contigus, pour le protocole indiqué. Pour les applications qui utilisent plusieurs ports non contigus, utilisez la commande **match access-list** et définissez une ACE pour chaque port.

```
hostname(config-cmap)# match tcp eq 80
```

- **match default-inspection-traffic** : établit une correspondance avec le trafic d'inspection par défaut : les ports TCP et UDP par défaut utilisés par toutes les applications que l'ASA peut inspecter.

```
hostname(config-cmap)# match default-inspection-traffic
```

  Cette commande, qui est utilisée dans la politique globale par défaut, est un raccourci spécial de l'interface de ligne de commande qui, lorsqu'il est utilisé dans une liste des politiques, garantit que l'inspection correcte est appliquée à chaque paquet, en fonction du port de destination du trafic. Par exemple, lorsque le trafic UDP pour le port 69 atteint l'ASA, l'ASA applique l'inspection TFTP ; lorsque le trafic TCP pour le port 21 arrive, l'ASA applique l'inspection FTP. Ainsi, dans ce cas uniquement, vous pouvez configurer plusieurs inspections pour la même carte de trafic (à l'exception de l'inspection WAAS, qui peut être configurée avec d'autres inspections. Consultez Incompatibilité de certaines actions de fonctionnalité, à la page 6 pour obtenir plus d'informations sur la combinaison d'actions). Normalement, l'ASA n'utilise pas le numéro de port pour déterminer l'inspection appliquée, vous permettant ainsi d'appliquer des inspections à des ports non standard, par exemple.

  Consultez Inspections par défaut et limites de la NAT pour obtenir la liste des ports par défaut. Toutes les applications dont les ports sont inclus dans la commande **match default-inspection-traffic** ne sont pas activées par défaut dans la liste des politiques.

  Vous pouvez spécifier une commande **match access-list** avec la commande **match default-inspection-traffic** afin de restreindre le trafic correspondant. Comme la commande **match default-inspection-traffic** spécifie les ports et les protocoles à mettre en correspondance, tous les ports et protocoles figurant dans l'ACL sont ignorés.

- **match dscp** *value1* [*value2*] [...] [*value8*] : correspond à la valeur DSCP dans un en-tête IP, jusqu'à huit valeurs DSCP.

```
hostname(config-cmap)# match dscp af43 cs1 ef
```

- **match precedence** *value1* [*value2*] [*value3*] [*value4*] : correspond à quatre valeurs de priorité, représentées par l'octet TOS dans l'en-tête IP, où les valeurs de priorité peuvent être comprises entre 0 et 7.

```
hostname(config-cmap)# match precedence 1 4
```

- **match rtp** *starting_port range* : correspond au trafic RTP, où le *starting_port* spécifie un port de destination UDP pair entre 2000 et 65534. La *range* spécifie le nombre de ports UDP supplémentaires à mettre en correspondance au-dessus du *starting_port*, entre 0 et 16 383.

```
hostname(config-cmap)# match rtp 4004 100
```

- **match tunnel-group** *name :* établit une correspondance avec le trafic du groupe de tunnels VPN auquel vous souhaitez appliquer la QoS.

Vous pouvez également spécifier une autre commande **match** afin d'affiner la correspondance du trafic. Vous pouvez spécifier n'importe laquelle des commandes précédentes, à l'exception de **match any**, **match access-list**, ou **match default-inspection-traffic**. Vous pouvez également saisir la commande **match flow ip destination-address** pour faire correspondre les flux dans le groupe de tunnels vers chaque adresse IP.

```
hostname(config-cmap)# match tunnel-group group1
hostname(config-cmap)# match flow ip destination-address
```

Exemples

Voici un exemple pour la commande **class-map** :

```
hostname(config)# access-list udp permit udp any any
hostname(config)# access-list tcp permit tcp any any
hostname(config)# access-list host_foo permit ip any 10.1.1.1 255.255.255.255

hostname(config)# class-map all_udp
hostname(config-cmap)# description "This class-map matches all UDP traffic"
hostname(config-cmap)# match access-list udp

hostname(config-cmap)# class-map all_tcp
hostname(config-cmap)# description "This class-map matches all TCP traffic"
hostname(config-cmap)# match access-list tcp

hostname(config-cmap)# class-map all_http
hostname(config-cmap)# description "This class-map matches all HTTP traffic"
hostname(config-cmap)# match port tcp eq http

hostname(config-cmap)# class-map to_server
hostname(config-cmap)# description "This class-map matches all traffic to server 10.1.1.1"
hostname(config-cmap)# match access-list host_foo
```

Créer une carte de trafic de couche 3/4 pour le trafic de gestion

Pour le trafic de gestion vers l'ASA, vous pouvez effectuer des actions spécifiques à ce type de trafic. Vous pouvez spécifier une carte de trafic de gestion qui peut correspondre à une ACL ou à des ports TCP ou UDP. Les types d'actions disponibles pour une carte de trafic de gestion dans la liste des politiques sont spécialisées dans le trafic de gestion.

Procédure

Étape 1 Créez une carte de trafic de gestion : **class-map type management** *class_map_name*

Où *class_map_name* est une chaîne d'une longueur maximale de 40 caractères.

Le nom « class-default » est réservé. Tous les types de cartes de trafic utilisent le même espace de nom, de sorte que vous ne pouvez pas réutiliser un nom déjà utilisé par un autre type de carte de trafic. L'interface de ligne de commande passe en mode de configuration class-map.

Exemple :

```
hostname(config)# class-map management all_udp
```

Étape 2 (Facultatif) Ajoutez une description à la carte de trafic.

description *string*

Exemple :

```
hostname(config-cmap)# description All UDP traffic
```

Étape 3 Établissez une correspondance du trafic à l'aide de l'une des commandes suivantes.

- **match access-list** *access_list_name* : établit une correspondance avec le trafic spécifié par une ACL étendue.

  ```
  hostname(config-cmap)# match access-list udp
  ```

- **match port** {**tcp** | **udp** | **sctp**} {**eq** *port_num* | **range** *port_num port_num*} : correspond aux ports de destination, qu'il s'agisse d'un port unique ou d'une plage de ports contigus, pour le protocole indiqué. Pour les applications qui utilisent plusieurs ports non contigus, utilisez la commande **match access-list** et définissez une ACE pour chaque port.

  ```
  hostname(config-cmap)# match tcp eq 80
  ```

Définir les actions (liste des politiques de couche 3/4)

Après avoir configuré les cartes de trafic de couche 3/4 pour identifier le trafic, utilisez une liste des politiques de couche 3/4 pour associer des actions à ces cartes.

Astuces Le nombre maximal de listes des politiques est de 64, mais vous ne pouvez appliquer qu'une seule liste des politiques par interface.

Procédure

Étape 1 Ajoutez la liste des politiques : **policy-map** *policy_map_name*

Où *policy_map_name* est le nom de la liste des politiques, jusqu'à 40 caractères de longueur. Tous les types de listes des politiques utilisent le même espace de noms, de sorte que vous ne pouvez pas réutiliser un nom déjà utilisé par un autre type de liste des politiques. L'interface de ligne de commande passe en mode de configuration policy-map.

Exemple :

```
hostname(config)# policy-map global_policy
```

Étape 2 Préciser une carte de trafic de couche 3/4 précédemment configurée : **class** *class_map_name*

Où *nclass_map_name* est le nom de la carte de trafic.

Consultez Identifier le trafic (cartes de trafic de couche 3/4), à la page 13 pour ajouter une carte de trafic.

Exemple :

```
hostname(config-pmap)# class all_http
```

Étape 3 Précisez une ou plusieurs actions pour cette carte de trafic.

Consultez Fonctionnalités configurées avec les politiques de service, à la page 3.

Remarque
S'il n'y a pas de commande **match default-inspection-traffic** dans une carte de trafic, au plus une commande **inspect** peut être configurée sous la classe.

Étape 4 Répétez le processus pour chaque carte de trafic que vous souhaitez inclure dans cette liste des politiques.

Exemples

Voici un exemple de commande **policy-map** pour une politique de connexion. Il limite le nombre de connexions autorisées pour le serveur Web 10.1.1.1 :

```
hostname(config)# access-list http-server permit tcp any host 10.1.1.1
hostname(config)# class-map http-server
hostname(config-cmap)# match access-list http-server

hostname(config)# policy-map global-policy
hostname(config-pmap)# description This policy map defines a policy concerning
connection to http server.
hostname(config-pmap)# class http-server
hostname(config-pmap-c)# set connection conn-max 256
```

L'exemple suivant montre le fonctionnement de la correspondance multiple dans une liste des politiques :

```
hostname(config)# class-map inspection_default
hostname(config-cmap)# match default-inspection-traffic
hostname(config)# class-map http_traffic
hostname(config-cmap)# match port tcp eq 80

hostname(config)# policy-map outside_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect http http_map
hostname(config-pmap-c)# inspect sip
hostname(config-pmap)# class http_traffic
hostname(config-pmap-c)# set connection timeout idle 0:10:0
```

L'exemple suivant montre comment le trafic correspond à la première carte de trafic disponible et ne correspondra pas aux cartes de trafic ultérieures qui précisent les actions dans le même domaine de fonctionnalité :

```
hostname(config)# class-map telnet_traffic
hostname(config-cmap)# match port tcp eq 23
hostname(config)# class-map ftp_traffic
hostname(config-cmap)# match port tcp eq 21
hostname(config)# class-map tcp_traffic
hostname(config-cmap)# match port tcp range 1 65535
hostname(config)# class-map udp_traffic
hostname(config-cmap)# match port udp range 0 65535
hostname(config)# policy-map global_policy
hostname(config-pmap)# class telnet_traffic
hostname(config-pmap-c)# set connection timeout idle 0:0:0
hostname(config-pmap-c)# set connection conn-max 100
hostname(config-pmap)# class ftp_traffic
hostname(config-pmap-c)# set connection timeout idle 0:5:0
hostname(config-pmap-c)# set connection conn-max 50
hostname(config-pmap)# class tcp_traffic
hostname(config-pmap-c)# set connection timeout idle 2:0:0
hostname(config-pmap-c)# set connection conn-max 2000
```

Lorsqu'une connexion Telnet est lancée, elle correspond à **la classe telnet_traffic**. De même, si une connexion FTP est initiée, elle correspond à **la classe ftp_traffic**. Pour toute connexion TCP autre que Telnet et FTP, elle correspondra à la **classe tcp_traffic**. Même si une connexion Telnet ou FTP peut correspondre à la classe **tcp_traffic**, l'ASA n'établit pas cette correspondance, car elles ont déjà correspondu à d'autres classes.

Appliquer les actions à une interface (politique de service)

Pour activer la carte de politiques de couche 3/4, créez une politique de service qui l'applique à une ou plusieurs interfaces ou qui l'applique globalement à toutes les interfaces. Utilisez la commande suivante :

service-policy *policy_map_name* {**global** | **interface** *interface_name*} [**fail-close**]

Lieu :

- *policy_map_name* est le nom de la liste des politiques.
- **global** crée une politique de service qui s'applique à toutes les interfaces qui n'ont pas de politique particulière.

  Vous ne pouvez appliquer qu'une seule politique globale, donc si vous souhaitez modifier la politique globale, vous devez soit modifier la politique par défaut, soit la désactiver et en appliquer une nouvelle. Par défaut, la configuration comprend une politique globale qui correspond à tout le trafic d'inspection d'application par défaut et applique l'inspection au trafic globalement. La politique de service par défaut comprend la commande suivante : **service-policy global_policy global**.
- **interface** *interface_name* crée une politique de service en associant une carte de politiques à une interface.
- **fail-close** génère un journal système (767001) pour le trafic IPv6 qui est abandonné par les inspections d'applications qui ne prennent pas en charge le trafic IPv6. Par défaut, les journaux système ne sont pas générés.

Exemples

Par exemple, la commande suivante active la carte de politiques inbound_policy sur l'interface externe :

```
hostname(config)# service-policy inbound_policy interface outside
```

Les commandes suivantes désactivent la politique globale par défaut et en activent une nouvelle appelée new_global_policy.

```
hostname(config)# no service-policy global_policy global
hostname(config)# service-policy new_global_policy global
```

Surveillance des politiques de service

Pour surveiller les politiques de service, entrez la commande suivante :

- **show service-policy**

  Affiche les statistiques de la politique de service.

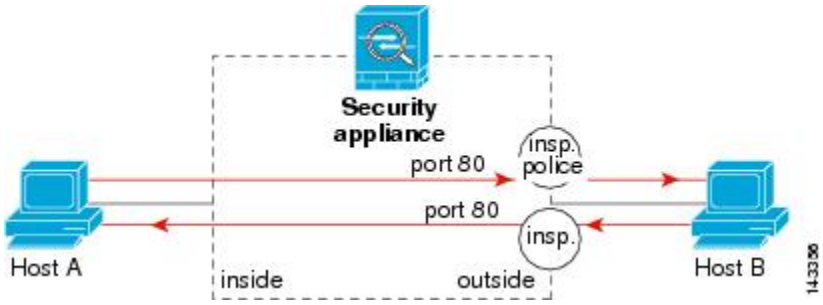
Exemples de politiques de service (cadre de politique modulaire)

Cette section comprend plusieurs exemples de Modular Policy Framework

Application de l'inspection et de la régulation QoS au trafic HTTP

Dans cet exemple, toute connexion HTTP (trafic TCP sur le port 80) qui entre dans l'ASA ou en sort par l'interface externe est classée pour l'inspection HTTP. Tout trafic HTTP qui quitte l'interface externe est classé pour le contrôle.

Illustration 1 : Inspection HTTP et politique QoS



Consultez les commandes suivantes pour cet exemple :

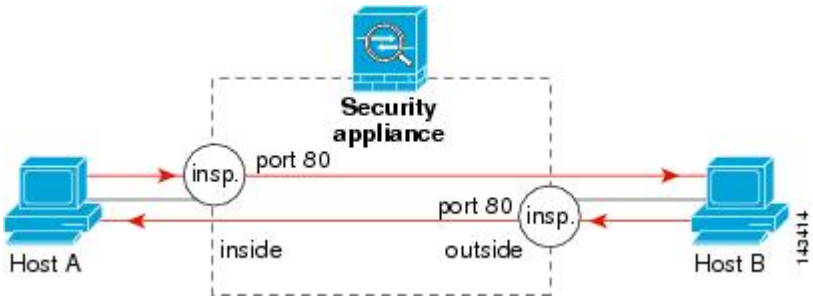
```
hostname(config)# class-map http_traffic
hostname(config-cmap)# match port tcp eq 80
```

```
hostname(config)# policy-map http_traffic_policy
hostname(config-pmap)# class http_traffic
hostname(config-pmap-c)# inspect http
hostname(config-pmap-c)# police output 250000
hostname(config)# service-policy http_traffic_policy interface outside
```

Application de l'inspection au trafic HTTP globalement

Dans cet exemple, toute connexion HTTP (trafic TCP sur le port 80) qui entre dans l'ASA par l'intermédiaire d'une interface est classée pour l'inspection HTTP. Comme la politique est une politique globale, l'inspection se produit uniquement lorsque le trafic entre dans chaque interface.

Illustration 2 : Inspection HTTP globale



Consultez les commandes suivantes pour cet exemple :

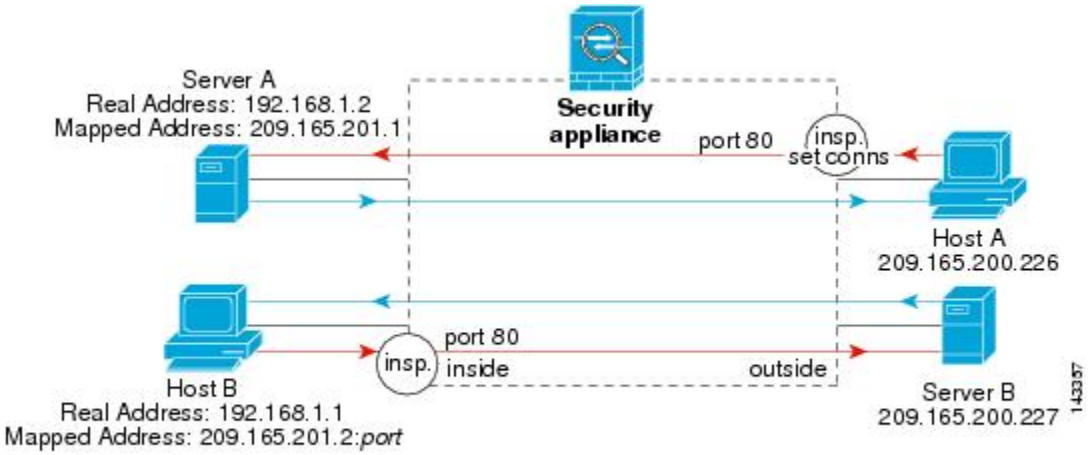
```
hostname(config)# class-map http_traffic
hostname(config-cmap)# match port tcp eq 80

hostname(config)# policy-map http_traffic_policy
hostname(config-pmap)# class http_traffic
hostname(config-pmap-c)# inspect http
hostname(config)# service-policy http_traffic_policy global
```

Application des limites d'inspection et de connexion au trafic HTTP vers des serveurs spécifiques

Dans cet exemple, toute connexion HTTP destinée au serveur A (trafic TCP sur le port 80) qui entre dans l'ASA par l'intermédiaire de l'interface externe est classée pour l'inspection HTTP et les limites de connexion maximales. Les connexions initiées du serveur A à l'hôte A ne correspondent pas à l'ACL dans la carte de trafic, elles ne sont donc pas affectées.

Toute connexion HTTP destinée au serveur B qui entre dans l'ASA par l'intermédiaire de l'interface interne est classée pour l'inspection HTTP. Les connexions initiées du serveur B à l'hôte B ne correspondent pas à l'ACL dans la carte de trafic, elles ne sont donc pas affectées.

Illustration 3 : Limites d'inspection et de connexion HTTP pour des serveurs spécifiques



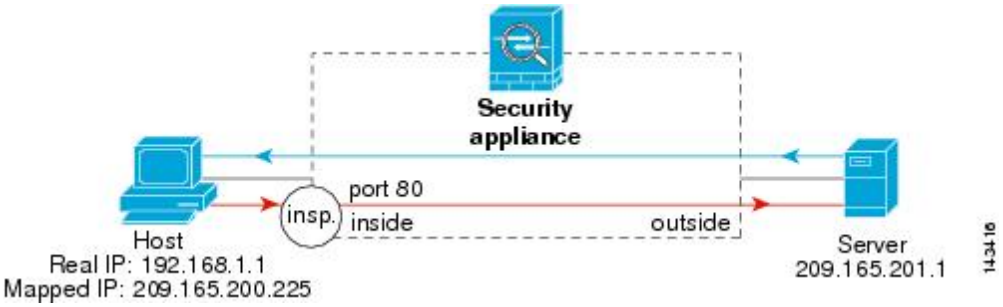
Consultez les commandes suivantes pour cet exemple :

```
hostname(config)# object network obj-192.168.1.2
hostname(config-network-object)# host 192.168.1.2
hostname(config-network-object)# nat (inside,outside) static 209.165.201.1
hostname(config)# object network obj-192.168.1.0
hostname(config-network-object)# subnet 192.168.1.0 255.255.255.0
hostname(config-network-object)# nat (inside,outside) dynamic 209.165.201.2
hostname(config)# access-list serverA extended permit tcp any host 209.165.201.1 eq 80
hostname(config)# access-list ServerB extended permit tcp any host 209.165.200.227 eq 80

hostname(config)# class-map http_serverA
hostname(config-cmap)# match access-list serverA
hostname(config)# class-map http_serverB
hostname(config-cmap)# match access-list serverB

hostname(config)# policy-map policy_serverA
hostname(config-pmap)# class http_serverA
hostname(config-pmap-c)# inspect http
hostname(config-pmap-c)# set connection conn-max 100
hostname(config)# policy-map policy_serverB
hostname(config-pmap)# class http_serverB
hostname(config-pmap-c)# inspect http

hostname(config)# service-policy policy_serverB interface inside
hostname(config)# service-policy policy_serverA interface outside
```

Application de l'inspection au trafic HTTP avec NAT

Dans cet exemple, l'hôte du réseau interne a deux adresses : l'une est l'adresse IP réelle 192.168.1.1, et l'autre est une adresse IP mappée utilisée sur le réseau externe, 209.165.200.225. Vous devez utiliser l'adresse IP réelle dans la liste de contrôle d'accès dans la carte de trafic. Si vous l'appliquiez à l'interface externe, vous utiliseriez également l'adresse réelle.

Illustration 4 : Inspection HTTP avec NAT



Consultez les commandes suivantes pour cet exemple :

```
hostname(config)# object network obj-192.168.1.1
hostname(config-network-object)# host 192.168.1.1
hostname(config-network-object)# nat (VM1,outside) static 209.165.200.225

hostname(config)# access-list http_client extended permit tcp host 192.168.1.1 any eq 80

hostname(config)# class-map http_client
hostname(config-cmap)# match access-list http_client

hostname(config)# policy-map http_client
hostname(config-pmap)# class http_client
hostname(config-pmap-c)# inspect http

hostname(config)# service-policy http_client interface inside
```

Historique des politiques de service

Nom de la caractéristique	Versions	Description
Cadre de politique modulaire	7.0(1)	Le cadre de politique modulaire a été lancé.
Carte de trafic de gestion à utiliser avec le trafic de comptabilité RADIUS	7.2(1)	La carte de trafic de gestion a été introduite pour une utilisation avec le trafic de comptabilité RADIUS. Les commandes suivantes ont été introduites : **class-map type management**, et **inspect radius-accounting**.
listes des politiques d'inspection	7.2(1)	La liste des politiques d'inspection a été introduite. La commande suivante a été introduite : **class-map type inspect**.
Expressions régulières et listes des politiques	7.2(1)	Les expressions régulières et les listes des politiques ont été introduites pour être utilisées dans les listes des politiques d'inspection. Les commandes suivantes ont été introduites : **class-map type regex**, **regex**, **match regex**.
Match any pour les listes des politiques d'inspection	8.0(2)	Le mot-clé **match any** a été introduit pour une utilisation avec des listes des politiques d'inspection : le trafic peut correspondre à un ou plusieurs critères pour correspondre à la carte de trafic. Auparavant, uniquement **match all** était disponible.

À propos de la traduction

Cisco peut fournir des traductions du présent contenu dans la langue locale pour certains endroits. Veuillez noter que des traductions sont fournies à titre informatif seulement et, en cas d’incohérence, la version anglaise du présent contenu prévaudra.

À propos de la traduction

Cisco peut fournir des traductions du présent contenu dans la langue locale pour certains endroits. Veuillez noter que des traductions sont fournies à titre informatif seulement et, en cas d’incohérence, la version anglaise du présent contenu prévaudra.

À propos de la traduction

Cisco peut fournir des traductions du présent contenu dans la langue locale pour certains endroits. Veuillez noter que des traductions sont fournies à titre informatif seulement et, en cas d’incohérence, la version anglaise du présent contenu prévaudra.