ASA/PIX 8.x : exemple de configuration du blocage de certains sites Web (URL) à l'aide d'expressions régulières avec MPF

Table des matières

Introduction

Ce document décrit comment configurer les dispositifs de sécurité Cisco ASA/PIX 8.x à l'aide d'expressions standards dans le cadre de règles modulaires (MPF) dans le but de bloquer certains sites Web (URL).

Remarque : cette configuration ne bloque pas tous les téléchargements d'applications. Pour un blocage fiable des fichiers, il est recommandé d'utiliser un appareil dédié tel que Ironport série S ou un module tel que le module CSC pour l'ASA.

Remarque : le filtrage HTTPS n'est pas pris en charge sur ASA. ASA ne peut pas effectuer d'inspection approfondie des paquets ou d'inspection basée sur une expression régulière pour le trafic HTTPS, car dans HTTPS, le contenu du paquet est chiffré (SSL).

Conditions préalables

Exigences

Ce document suppose que l'appliance de sécurité Cisco est configurée et fonctionne correctement.

Composants utilisés

- Appareil de sécurité adaptatif (ASA) de la gamme Cisco 5500 qui exécute le logiciel version 8.0(x) et ultérieure
- Cisco Adaptive Security Device Manager (ASDM) version 6.x pour ASA 8.x

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Produits connexes

Cette configuration peut également être utilisée avec le PIX de la gamme Cisco 500 qui exécute le logiciel version 8.0(x) et ultérieure.

Conventions

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

Informations générales

Présentation du cadre de stratégie modulaire

Le protocole MPF offre un moyen cohérent et flexible de configurer les fonctions des dispositifs de sécurité. Par exemple, vous pouvez utiliser MPF pour créer une configuration de délai d'attente spécifique à une application TCP particulière, par opposition à une configuration qui s'applique à toutes les applications TCP.

MPF prend en charge les fonctionnalités suivantes :

- Normalisation TCP, limites et délais de connexion TCP et UDP, et randomisation des numéros de séquence TCP
- CSC
- Inspection d'application
- IPS
- Contrôle d'entrée QoS
- Contrôle de sortie QoS

- File prioritaire QoS

La configuration du MPF se compose de quatre tâches :

1. Identifiez le trafic des couches 3 et 4 auquel vous souhaitez appliquer des actions. Référez-vous à Identification du trafic à l'aide d'un mappage de classe de couche 3/4 pour plus d'informations.

2. (Contrôle des applications uniquement) Définissez des actions spéciales pour le trafic de contrôle des applications. Référez-vous à Configuration d'actions spéciales pour les inspections d'application pour plus d'informations.

3. Appliquez des actions au trafic des couches 3 et 4. Référez-vous à Définition d'actions à l'aide d'une carte de stratégie de couche 3/4 pour plus d'informations.

4. Activez les actions sur une interface. Référez-vous à Application d'une politique de couche 3/4 à une interface à l'aide d'une politique de service pour plus d'informations.

Expression régulière

Une expression régulière fait correspondre des chaînes de texte littéralement sous la forme d'une chaîne exacte, ou à l'aide de métacaractères afin que vous puissiez faire correspondre plusieurs variantes d'une chaîne de texte. Vous pouvez utiliser une expression régulière pour faire correspondre le contenu d'un certain trafic d'application ; par exemple, vous pouvez faire correspondre une chaîne d'URL à l'intérieur d'un paquet HTTP.

Remarque : utilisez Ctrl+V afin d'échapper tous les caractères spéciaux de la CLI, tels que le point d'interrogation (?) ou une tabulation. Par exemple, tapez d[Ctrl+V]?g afin d'entrer d?g dans la configuration.

Pour la création d'une expression régulière, utilisez la commande regex, qui peut être utilisée pour diverses fonctionnalités qui nécessitent une correspondance de texte. Par exemple, vous pouvez configurer des actions spéciales pour l'inspection des applications à l'aide du cadre de stratégie modulaire qui utilise une carte de stratégie d'inspection. Référez-vous à la commande policy map type inspect pour plus d'informations. Dans la carte de stratégie d'inspection, vous pouvez identifier le trafic sur lequel vous voulez agir si vous créez une carte de classe d'inspection qui contient une ou plusieurs commandes match ou vous pouvez utiliser des commandes match directement dans la carte de stratégie d'inspection. Certaines commandes match vous permettent d'identifier le texte d'un paquet à l'aide d'une expression régulière ; par exemple, vous pouvez faire correspondre des chaînes d'URL dans des paquets HTTP. Vous pouvez regrouper des expressions régulières dans une carte de classe d'expression régulière. Référez-vous à la commande class-map type regex pour plus d'informations.

Ce tableau répertorie les métacaractères qui ont des significations spéciales.

Caractère	Description	Remarques
.	Point	Correspond à n'importe quel caractère unique. Par

		exemple, d.g correspond à « dog », « dag », « dtg » et à tout mot contenant ces caractères, tel que « doggonnit ».
(exp)	Sous-Expression	Une sous-expression sépare les caractères des caractères environnants, de sorte que vous pouvez utiliser d'autres métacaractères sur la sous-expression. Par exemple, d(o|a)g correspond à dog et dag, mais do|ag correspond à do et ag. Une sous-expression peut également être utilisée avec des quantificateurs de répétition pour différencier les caractères destinés à la répétition. Par exemple, ab(xy){3}z correspond à abxyxyxyz.
|	Alternance	Correspond à l'expression qu'elle sépare. Par exemple, dog|cat correspond à dog ou cat.
?	Point d'interrogation	Quantificateur qui indique qu'il existe 0 ou 1 dans l'expression précédente. Par exemple, lo?se correspond à lse ou à lose. Remarque : vous devez saisir Ctrl+V, puis le point d'interrogation, sinon la fonction d'aide est appelée.
*	Astérisque	Quantificateur qui indique qu'il existe 0, 1 ou n'importe quel nombre de l'expression précédente. Par exemple, lo*se correspond à lse, lose, lose, etc.
{x}	Quantificateur	Répétez exactement x fois.

	de répétition	Par exemple, ab(xy){3}z correspond à abxyxyxyz.
{x,}	Quantificateur de répétition minimal	Répétez au moins x fois. Par exemple, ab(xy){2,}z correspond à abxyxyz, abxyxyxyz, etc.
[abc]	Classe Character	Correspond à n'importe quel caractère entre crochets. Par exemple, [abc] correspond à a, b ou c.
[^abc]	Classe de caractères inversée	Correspond à un caractère unique qui n'est pas contenu entre crochets. Par exemple, [^abc] correspond à tout caractère autre que a, b ou c. [^A-Z] correspond à tout caractère unique qui n'est pas une lettre majuscule.
[a-c]	Classe de plage de caractères	Correspond à n'importe quel caractère de la plage. [a-z] correspond aux minuscules. Vous pouvez mélanger des caractères et des plages : [abcq-z] correspond à a, b, c, q, r, s, t, u, v, w, x, y, z et [a-cq-z] correspond à ce paramètre. Le tiret (-) est littéral uniquement s'il s'agit du dernier ou du premier caractère entre crochets : [abc-] ou [-abc].
""	Guillemets	Conserve les espaces de fin ou de début dans la chaîne. Par exemple, " test" préserve l'espace d'en-tête lorsqu'il recherche une correspondance.
^	Caret	Spécifie le début d'une ligne
\	Caractère d'échappement	Utilisé avec un métacaractère, correspond à un caractère littéral. Par

		exemple, \[correspond au crochet gauche.
carboniser	Caractère	Lorsque caractère n'est pas un métacaractère, fait correspondre le caractère littéral.
\r	Retour de chariot	Correspond à un retour chariot 0x0d
\n	Nouvelle Ligne	Correspond à une nouvelle ligne 0x0a
\t	Onglet	Correspond à un onglet 0x09
\f	Alimentation	Correspond à un flux de formulaire 0x0c
\xNN	Nombre hexadécimal échappé	Correspond à un caractère ASCII qui utilise un hexadécimal de deux chiffres
\NNN	Nombre octal échappé	Correspond à un caractère ASCII octal qui est exactement à trois chiffres. Par exemple, le caractère 040 représente un espace.

Configurer

Cette section vous fournit des informations pour configurer les fonctionnalités décrites dans ce document.

Remarque : Utilisez l'outil de recherche de commandes (clients enregistrés seulement) pour en savoir plus sur les commandes employées dans cette section.

Diagramme du réseau

Ce document utilise la configuration réseau suivante :

Configurations

Ce document utilise les configurations suivantes :

- Configuration de l'interface de ligne de commande ASA
- Configuration ASA 8.x avec ASDM 6.x

Configuration de l'interface de ligne de commande ASA

Configuration de l'interface de ligne de commande A

<#root>

```
ciscoasa#
show running-config

: Saved
:
ASA Version 8.0(2)
!
hostname ciscoasa
domain-name default.domain.invalid
enable password 8Ry2YjIyt7RRXU24 encrypted
names
!
interface Ethernet0/0
 nameif inside
 security-level 100
 ip address 10.1.1.1 255.255.255.0
!
interface Ethernet0/1
 nameif outside
 security-level 0
 ip address 192.168.1.5 255.255.255.0
!
interface Ethernet0/2
 nameif DMZ
 security-level 90
 ip address 10.77.241.142 255.255.255.192
!
interface Ethernet0/3
 shutdown
 no nameif
 no security-level
 no ip address
!
interface Management0/0
 shutdown
 no nameif
 no security-level
 no ip address
!
passwd 2KFQnbNIdI.2KYOU encrypted

regex urllist1 ".*\.([Ee][Xx][Ee]|[Cc][Oo][Mm]|[Bb][Aa][Tt]) HTTP/1.[01]"


!--- Extensions such as .exe, .com, .bat to be captured and !--- provided the http version being used


regex urllist2 ".*\.([Pp][Ii][Ff]|[Vv][Bb][Ss]|[Ww][Ss][Hh]) HTTP/1.[01]"

!--- Extensions such as .pif, .vbs, .wsh to be captured !--- and provided the http version being used
```

```
regex urllist3 ".*\.([Dd][Oo][Cc]|[Xx][Ll][Ss]|[Pp][Pp][Tt]) HTTP/1.[01]"

!--- Extensions such as .doc(word), .xls(ms-excel), .ppt to be captured and provided !--- the http ver

regex urllist4 ".*\.([Zz][Ii][Pp]|[Tt][Aa][Rr]|[Tt][Gg][Zz]) HTTP/1.[01]"

!--- Extensions such as .zip, .tar, .tgz to be captured and provided !--- the http version being used

regex domainlist1 "\.yahoo\.com"
regex domainlist2 "\.myspace\.com"
regex domainlist3 "\.youtube\.com"

!--- Captures the URLs with domain name like yahoo.com, !--- youtube.com and myspace.com

regex contenttype "Content-Type"
regex applicationheader "application/.*"

!--- Captures the application header and type of !--- content in order for analysis

boot system disk0:/asa802-k8.bin
ftp mode passive
dns server-group DefaultDNS
 domain-name default.domain.invalid

access-list inside_mpc extended permit tcp any any eq www

access-list inside_mpc extended permit tcp any any eq 8080

!--- Filters the http and port 8080 !--- traffic in order to block the specific traffic with regular

pager lines 24
mtu inside 1500
mtu outside 1500
mtu DMZ 1500
no failover
icmp unreachable rate-limit 1 burst-size 1
asdm image disk0:/asdm-602.bin
no asdm history enable
arp timeout 14400
route DMZ 0.0.0.0 0.0.0.0 10.77.241.129 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 icmp 0:00:02
```

```
timeout sunrpc 0:10:00 h323 0:05:00 h225 1:00:00 mgcp 0:05:00 mgcp-pat 0:05:00
timeout sip 0:30:00 sip_media 0:02:00 sip-invite 0:03:00 sip-disconnect 0:02:00
timeout uauth 0:05:00 absolute
dynamic-access-policy-record DfltAccessPolicy
http server enable
http 0.0.0.0 0.0.0.0 DMZ
no snmp-server location
no snmp-server contact
snmp-server enable traps snmp authentication linkup linkdown coldstart
no crypto isakmp nat-traversal
telnet timeout 5
ssh timeout 5
console timeout 0
threat-detection basic-threat
threat-detection statistics access-list
!
class-map type regex match-any DomainBlockList
 match regex domainlist1
 match regex domainlist2
 match regex domainlist3


!--- Class map created in order to match the domain names !--- to be blocked


class-map type inspect http match-all BlockDomainsClass
 match request header host regex class DomainBlockList


!--- Inspect the identified traffic by class !--- "DomainBlockList".


class-map type regex match-any URLBlockList
 match regex urllist1
 match regex urllist2
 match regex urllist3
 match regex urllist4


!--- Class map created in order to match the URLs !--- to be blocked


class-map inspection_default
 match default-inspection-traffic


class-map type inspect http match-all AppHeaderClass
 match response header regex contenttype regex applicationheader


!--- Inspect the captured traffic by regular !--- expressions "content-type" and "applicationheader".


class-map httptraffic
 match access-list inside_mpc
```

```
!--- Class map created in order to match the !--- filtered traffic by ACL


class-map type inspect http match-all BlockURLsClass
 match request uri regex class URLBlockList
!
!--- Inspect the identified traffic by class !--- "URLBlockList".


!
policy-map type inspect dns preset_dns_map
 parameters
  message-length maximum 512
policy-map type inspect http http_inspection_policy
 parameters
  protocol-violation action drop-connection
 class AppHeaderClass
  drop-connection log
 match request method connect
  drop-connection log
 class BlockDomainsClass
  reset log
 class BlockURLsClass
  reset log


!--- Define the actions such as drop, reset or log !--- in the inspection policy map.


policy-map global_policy
 class inspection_default
  inspect dns preset_dns_map
  inspect ftp
  inspect h323 h225
  inspect h323 ras
  inspect netbios
  inspect rsh
  inspect rtsp
  inspect skinny
  inspect esmtp
  inspect sqlnet
  inspect sunrpc
  inspect tftp
  inspect sip
  inspect xdmcp


policy-map inside-policy
 class httptraffic
  inspect http http_inspection_policy


!--- Map the inspection policy map to the class !--- "httptraffic" under the policy map created for th
```

```
!
service-policy global_policy global
service-policy inside-policy interface inside


!--- Apply the policy to the interface inside where the websites are blocked.


prompt hostname context
Cryptochecksum:e629251a7c37af205c289cf78629fc11
: end
ciscoasa#
```

Configuration ASA 8.x avec ASDM 6.x

Complétez ces étapes afin de configurer les expressions régulières et de les appliquer dans MPF pour bloquer les sites Web spécifiques comme indiqué.

1. Créer des expressions régulières

   Choisissez Configuration > Firewall> Objects > Regular Expressions et cliquez sur Add sous l'onglet Regular Expression afin de créer des expressions régulières comme indiqué.

   a. Créez une expression régulière domainlist1 afin de capturer le nom de domaine yahoo.com. Click OK.

   b. Créez une expression régulière domainlist2 afin de capturer le nom de domaine myspace.com. Click OK.

   c. Créez une expression régulière domainlist3 afin de capturer le nom de domaine youtube.com. Click OK.

   d. Créez une expression régulière urllist1 afin de capturer les extensions de fichiers telles que exe, com et bat à condition que la version http utilisée par le navigateur Web soit 1.0 ou 1.1. Click OK.

   e. Créez une expression régulière urllist2 afin de capturer les extensions de fichier telles que pif, vbs et wsh à condition que la version http utilisée par le navigateur Web soit 1.0 ou 1.1. Click OK.

   f. Créez une expression régulière urllist3 afin de capturer les extensions de fichier telles que doc, xls et ppt à condition que la version http utilisée par le navigateur Web soit 1.0 ou 1.1. Click OK.

   g. Créez une expression régulière urllist4 afin de capturer les extensions de fichier telles que zip, tar et tgz à condition que la version http utilisée par le navigateur Web soit 1.0 ou 1.1. Click OK.

   h. Créez une expression régulière contenttype afin de capturer le type de contenu. Click

OK.

i. Créez une expression régulière applicationheader afin de capturer les divers en-têtes d'application. Click OK.

Configuration CLI équivalente

Configuration de l'interface de ligne de commande ASA
<#root> ciscoasa# **configure terminal** ciscoasa(config)# **regex urllist1** ".*\.([Ee][Xx][Ee]|[Cc][Oo][Mm]|[Bb][Aa][Tt])$ ciscoasa(config)# **regex urllist2** ".*\.([Pp][Ii][Ff]|[Vv][Bb][Ss]|[Ww][Ss][Hh])$ ciscoasa(config)# **regex urllist3** ".*\.([Dd][Oo][Cc]|[Xx][Ll][Ss]|[Pp][Pp][Tt])$ ciscoasa(config)# **regex urllist4** ".*\.([Zz][Ii][Pp]|[Tt][Aa][Rr]|[Tt][Gg][Zz])$ ciscoasa(config)# **regex domainlist1** "\.yahoo\.com" ciscoasa(config)# **regex domainlist2** "\.myspace\.com" ciscoasa(config)# **regex domainlist3** "\.youtube\.com" ciscoasa(config)# **regex contenttype** "Content-Type" ciscoasa(config)# **regex applicationheader** "application/.*"

2. Créer des classes d'expression régulière

   Choisissez Configuration > Firewall > Objects > Regular Expressions et cliquez sur Add sous l'onglet Regular Expression Classes afin de créer les différentes classes comme indiqué.

   a. Créez une classe d'expression régulière DomainBlockList afin de correspondre à l'une des expressions régulières domainlist1, domainlist2 et domainlist3. Click OK.

   b. Créez une classe d'expression régulière URLBlockList afin de correspondre à l'une des expressions régulières urllist1, urllist2, urllist3 et urllist4. Click OK.

      Configuration CLI équivalente

Configuration de l'interface de ligne de commande ASA
<#root> ciscoasa# **configure terminal** ciscoasa(config)# **class-map type inspect http match-all BlockDomainsClass** ciscoasa(config-cmap)# **match request header host regex class DomainBlockList** ciscoasa(config-cmap)# **exit** ciscoasa(config)# **class-map type regex match-any URLBlockList** ciscoasa(config-cmap)# **match regex urllist1** ciscoasa(config-cmap)# **match regex urllist2** ciscoasa(config-cmap)# **match regex urllist3** ciscoasa(config-cmap)# **match regex urllist4** ciscoasa(config-cmap)# **exit**

3. Inspecter le trafic identifié avec des cartes de classe

Choisissez Configuration > Firewall > Objects > Class Maps > HTTP > Add afin de créer une carte de classe pour inspecter le trafic http identifié par diverses expressions régulières comme illustré.

a. Créez une carte de classe AppHeaderClass afin de faire correspondre l'en-tête de réponse avec les captures d'expressions régulières.

   Click OK

b. Créez un mappage de classe BlockDomainsClass afin de faire correspondre l'en-tête de requête avec les captures d'expressions régulières.

   Click OK.

c. Créez une carte de classe BlockURLsClass afin de faire correspondre l'URI de requête avec les captures d'expressions régulières.

   Click OK.

   Configuration CLI équivalente

Configuration de l'interface de ligne de commande ASA
<#root> ciscoasa# **configure terminal** ciscoasa(config)# **class-map type inspect http match-all AppHeaderClass** ciscoasa(config-cmap)# **match response header regex contenttype regex applicationheader** ciscoasa(config-cmap)# **exit** ciscoasa(config)# **class-map type inspect http match-all BlockDomainsClass** ciscoasa(config-cmap)# **match request header host regex class DomainBlockList** ciscoasa(config-cmap)# **exit** ciscoasa(config)# **class-map type inspect http match-all BlockURLsClass** ciscoasa(config-cmap)#

```
match request uri regex class URLBlockList

ciscoasa(config-cmap)#

exit
```

4. Définir les actions pour le trafic correspondant dans la stratégie d'inspection

   Choisissez Configuration > Firewall > Objects > Inspect Maps > HTTP afin de créer une http_inspection_policy pour définir l'action pour le trafic correspondant comme indiqué. Click OK.

   a. Choisissez Configuration > Firewall > Objects > Inspect Maps > HTTP > http_inspection_policy (double-clic) et cliquez sur Details > Add afin de définir les actions pour les différentes classes créées jusqu'à présent.

   b. Définissez l'action sur Abandonner la connexion et Activez la journalisation pour le critère comme méthode de requête et la valeur comme connexion.

      Click OK

   c. Définissez l'action sur Drop Connection et Enable pour la classe AppHeaderClass .

      Click OK.

   d. Définissez l'action sur Reset et Enable pour la journalisation de la classe BlockDomainsClass.

      Click OK

   e. Définissez l'action sur Reset et Enable pour la journalisation de la classe BlockURLsClass.

      Click OK.

      Cliquez sur Apply.

      Configuration CLI équivalente

Configuration de l'interface de ligne de commande ASA
<#root> ciscoasa# **configure terminal** ciscoasa(config)#

```
policy-map type inspect http http_inspection_policy
ciscoasa(config-pmap)#
parameters
ciscoasa(config-pmap-p)#
match request method connect
ciscoasa(config-pmap-c)#
drop-connection log
ciscoasa(config-pmap-c)#
class AppHeaderClass
ciscoasa(config-pmap-c)#
drop-connection log
ciscoasa(config-pmap-c)#
class BlockDomainsClass
ciscoasa(config-pmap-c)#
reset log
ciscoasa(config-pmap-c)#
class BlockURLsClass
ciscoasa(config-pmap-c)#
reset log
ciscoasa(config-pmap-c)#
exit
ciscoasa(config-pmap)#
exit
```

5. Appliquez la stratégie d'inspection http à l'interface

   Choisissez Configuration > Firewall > Service Policy Rules > Add > Add Service Policy Rule.

   a. Trafic HTTP

      a. Sélectionnez la case d'option Interface avec interface interne dans le menu déroulant et Nom de la stratégie comme stratégie interne. Cliquez sur Next (Suivant).

      b. Créez une carte de classe httptraffic et vérifiez les adresses IP source et de destination (utilise la liste de contrôle d'accès). Cliquez sur Next (Suivant).

      c. Choisissez Source et Destination as any avec service as tcp-udp/http. Cliquez

sur Next (Suivant).

d. Cochez la case d'option HTTP et cliquez sur Configure.

e. Cochez la case d'option Sélectionnez une carte d'inspection HTTP pour le contrôle de l'inspection comme indiqué. Click OK.

f. Cliquez sur Finish (Terminer).

b. Trafic du port 8080

a. Choisissez à nouveau Add > Add Service Policy Rule.

b. Cliquez sur Next (Suivant).

c. Sélectionnez la case d'option Add rule to existing traffic class et choisissez httptraffic dans le menu déroulant. Cliquez sur Next (Suivant).

d. Choisissez la Source et la Destination comme tout avec tcp/8080. Cliquez sur Next (Suivant).

e. Cliquez sur Finish (Terminer).

Cliquez sur Apply.

Configuration CLI équivalente

Configuration de l'interface de ligne de commande ASA
<#root> ciscoasa# **configure terminal** ciscoasa(config)# **access-list inside_mpc extended permit tcp any any eq www** ciscoasa(config)# **access-list inside_mpc extended permit tcp any any eq 8080** ciscoasa(config)# **class-map httptraffic** ciscoasa(config-cmap)# **match access-list inside_mpc** ciscoasa(config-cmap)# **exit** ciscoasa(config)#

```
policy-map inside-policy
ciscoasa(config-pmap)#
class httptraffic
ciscoasa(config-pmap-c)#
inspect http http_inspection_policy
ciscoasa(config-pmap-c)#
exit
ciscoasa(config-pmap)#
exit
ciscoasa(config)#
service-policy inside-policy interface inside
```

Vérifier

Utilisez cette section pour confirmer que votre configuration fonctionne correctement.

L'Outil Interpréteur de sortie (clients enregistrés uniquement) (OIT) prend en charge certaines commandes show. Utilisez l'OIT pour afficher une analyse de la sortie de la commande show .

- show running-config regex : affiche les expressions régulières qui ont été configurées

<#root>

```
ciscoasa#
show running-config regex
regex urllist1 ".*\.([Ee][Xx][Ee]|[Cc][Oo][Mm]|[Bb][Aa][Tt]) HTTP/1.[01]"
regex urllist2 ".*\.([Pp][Ii][Ff]|[Vv][Bb][Ss]|[Ww][Ss][Hh]) HTTP/1.[01]"
regex urllist3 ".*\.([Dd][Oo][Cc]|[Xx][Ll][Ss]|[Pp][Pp][Tt]) HTTP/1.[01]"
regex urllist4 ".*\.([Zz][Ii][Pp]|[Tt][Aa][Rr]|[Tt][Gg][Zz]) HTTP/1.[01]"
regex domainlist1 "\.yahoo\.com"
regex domainlist2 "\.myspace\.com"
regex domainlist3 "\.youtube\.com"
regex contenttype "Content-Type"
regex applicationheader "application/.*"
ciscoasa#
```

- show running-config class-map : affiche les cartes de classe qui ont été configurées

<#root>

```
ciscoasa#
```

```
show running-config class-map

!
class-map type regex match-any DomainBlockList
 match regex domainlist1
 match regex domainlist2
 match regex domainlist3
class-map type inspect http match-all BlockDomainsClass
 match request header host regex class DomainBlockList
class-map type regex match-any URLBlockList
 match regex urllist1
 match regex urllist2
 match regex urllist3
 match regex urllist4
class-map inspection_default
 match default-inspection-traffic
class-map type inspect http match-all AppHeaderClass
 match response header regex contenttype regex applicationheader
class-map httptraffic
 match access-list inside_mpc
class-map type inspect http match-all BlockURLsClass
 match request uri regex class URLBlockList
!
ciscoasa#
```

- show running-config policy-map type inspect http : affiche les cartes de stratégie qui inspectent le trafic http qui ont été configurées

```
<#root>

ciscoasa#

show running-config policy-map type inspect http

!
policy-map type inspect http http_inspection_policy
 parameters
  protocol-violation action drop-connection
 class AppHeaderClass
  drop-connection log
 match request method connect
  drop-connection log
 class BlockDomainsClass
  reset log
 class BlockURLsClass
  reset log
!
ciscoasa#
```

- show running-config policy-map : affiche toutes les configurations policy-map ainsi que la configuration policy-map par défaut

```
<#root>
```

```
ciscoasa#
```

show running-config policy-map

```
!
policy-map type inspect dns preset_dns_map
 parameters
  message-length maximum 512
policy-map type inspect http http_inspection_policy
 parameters
  protocol-violation action drop-connection
 class AppHeaderClass
  drop-connection log
 match request method connect
  drop-connection log
 class BlockDomainsClass
  reset log
 class BlockURLsClass
  reset log
policy-map global_policy
 class inspection_default
  inspect dns preset_dns_map
  inspect ftp
  inspect h323 h225
  inspect h323 ras
  inspect netbios
  inspect rsh
  inspect rtsp
  inspect skinny
  inspect esmtp
  inspect sqlnet
  inspect sunrpc
  inspect tftp
  inspect sip
  inspect xdmcp
policy-map inside-policy
 class httptraffic
  inspect http http_inspection_policy
!
ciscoasa#
```

- show running-config service-policy : affiche toutes les configurations de stratégie de service en cours d'exécution

<#root>

```
ciscoasa#
```

show running-config service-policy

```
service-policy global_policy global
service-policy inside-policy interface inside
```

- show running-config access-list : affiche la configuration de la liste d'accès qui s'exécute sur l'appliance de sécurité

```
<#root>

ciscoasa#

show running-config access-list

access-list inside_mpc extended permit tcp any any eq www

access-list inside_mpc extended permit tcp any any eq 8080
ciscoasa#
```

Dépannage

Cette section fournit des informations que vous pouvez utiliser pour dépanner votre configuration.

Remarque : Consulter les renseignements importants sur les commandes de débogage avant d'utiliser les commandes de débogage.

- debug http : affiche les messages de débogage pour le trafic HTTP

Informations connexes

- Assistance des dispositifs de sécurité adaptatifs dédiés de la gamme Cisco ASA 5500
- Prise en charge de Cisco Adaptive Security Device Manager (ASDM)
- Assistance des dispositifs de sécurité de la gamme Cisco PIX 500
- Logiciels pare-feu Cisco PIX
- Références des commandes du pare-feu Cisco Secure PIX
- Notices de champs relatives aux produits de sécurité (y compris PIX)
- Demandes de commentaires (RFC)
- Assistance et documentation techniques - Cisco Systems

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.