ASA/PIX 8.x:MPF 컨피그레이션이 포함된 정규식을 사용하여 특정 웹 사이트(URL) 차단 예

목차

소개

이 문서에서는 특정 웹 사이트(URL)를 차단하기 위해 MPF(Modular Policy Framework)를 사용하여 정규식을 사용하는 Cisco Security Appliances ASA/PIX 8.x를 구성하는 방법에 대해 설명합니다.

참고: 이 컨피그레이션은 모든 애플리케이션 다운로드를 차단하지는 않습니다.안정적인 파일 차단을 위해서는 Ironport S Series와 같은 전용 어플라이언스 또는 ASA용 CSC 모듈과 같은 모듈을 사용해야 합니다.

참고: ASA에서는 HTTPS 필터링이 지원되지 않습니다.HTTPS에서 패킷의 콘텐츠가 암호화(SSL)되기 때문에 ASA는 HTTPS 트래픽에 대한 정규식을 기반으로 심층 패킷 검사 또는 검사를 수행할 수 없습니다.

사전 요구 사항

요구 사항

이 문서에서는 Cisco Security Appliance가 구성되어 제대로 작동한다고 가정합니다.

사용되는 구성 요소

- 소프트웨어 버전 8.0(x) 이상을 실행하는 Cisco 5500 Series ASA(Adaptive Security Appliance)
- ASA 8.x용 Cisco ASDM(Adaptive Security Device Manager) 버전 6.x

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다.이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다.현재 네트워크가 작동 중인 경우, 모든 명령어의 잠재적인 영향을 미리 숙지하시기 바랍니다.

관련 제품

이 컨피그레이션은 소프트웨어 버전 8.0(x) 이상을 실행하는 Cisco 500 Series PIX에서도 사용할 수 있습니다.

표기 규칙

문서 규칙에 대한 자세한 내용은 Cisco 기술 팁 규칙을 참조하십시오.

배경 정보

모듈식 정책 프레임워크 개요

MPF는 보안 어플라이언스 기능을 구성할 수 있는 일관되고 유연한 방법을 제공합니다.예를 들어, 모든 TCP 애플리케이션에 적용되는 것과 달리 MPF를 사용하여 특정 TCP 애플리케이션에 특정한 시간 제한 컨피그레이션을 생성할 수 있습니다.

MPF는 다음 기능을 지원합니다.

- TCP 정규화, TCP 및 UDP 연결 제한 및 시간 제한, TCP 시퀀스 번호 임의 설정
- CSC
- 애플리케이션 검사
- IPS
- QoS 입력 폴리싱
- QoS 출력 폴리싱
- QoS 우선순위 큐

MPF 컨피그레이션은 다음 네 가지 작업으로 구성됩니다.

1. 작업을 적용할 레이어 3 및 4 트래픽을 식별합니다.자세한 내용은 레이어 3/4 클래스 맵을 사용하여 트래픽 식별을 참조하십시오.
2. (애플리케이션 검사만 해당) 애플리케이션 검사 트래픽에 대한 특별 작업을 정의합니다.자세한 내용은 애플리케이션 검사를 위한 특별 작업 구성을 참조하십시오.
3. 레이어 3 및 4 트래픽에 작업을 적용합니다.자세한 내용은 레이어 3/4 정책 맵을 사용하여 작업 정의를 참조하십시오.
4. 인터페이스에서 작업을 활성화합니다.자세한 내용은 서비스 정책을 사용하여 인터페이스에 레이어 3/4 정책 적용을 참조하십시오.

정규식

정규식은 텍스트 문자열을 문자 그대로 정확한 문자열로 매칭하거나 메타 문자를 사용하여 텍스트 문자열의 여러 변형을 매칭할 수 있습니다.정규식을 사용하여 특정 애플리케이션 트래픽의 내용을 일치시킬 수 있습니다.예를 들어, HTTP 패킷 내에서 URL 문자열을 매칭할 수 있습니다.

참고: Ctrl+V를 사용하여 물음표(?) 또는 탭과 같은 CLI의 모든 특수 문자를 이스케이프합니다.예를 들어 **d[Ctrl+V]?g를** 입력하여 컨피그레이션에 **d?g**를 입력합니다.

정규식을 만들려면 **regex** 명령을 사용합니다. 이 명령은 텍스트 일치를 필요로 하는 다양한 기능에 사용할 수 있습니다.예를 들어 검사 정책 맵을 사용하는 Modular Policy Framework를 사용하여 애플리케이션 검사에 대한 특수 작업을 구성할 수 있습니다.자세한 내용은 **정책 맵 유형 inspect 명령**을 참조하십시오.검사 정책 맵에서 하나 이상의 **match** 명령이 포함된 검사 클래스 맵을 생성하거나 검사 정책 맵에서 직접 **match** 명령을 사용할 수 있는 경우 수행할 트래픽을 식별할 수 있습니다.일부 **match** 명령을 사용하면 정규식을 사용하여 패킷의 텍스트를 식별할 수 있습니다.예를 들어, HTTP 패킷 내에서 URL 문자열을 매칭할 수 있습니다.정규식 클래스 맵에서 정규식을 그룹화할 수 있습니다.자세한 내용은 **class-map type regex 명령**을 참조하십시오.

이 표는 특별한 의미를 갖는 메타 문자를 나열합니다.

문자	설명	참고
.	점	단일 문자와 일치합니다.예를 들어 **d.g는** dog, dag, dtg 및 이러한 문자가 포함된 단어(예: doggonnit)와 일치합니다.
(exp)	하위 식	하위 식은 문자를 주변 문자와 분리하므로 하위 식에서 다른 메타 문자를 사용할 수 있습니다.예를 들어 **d(o|a)g는** dog 및 dag와 일치하지만 **do|ag**는 do 및 ag와 일치합니다.하위 식을 반복 한정자와 함께 사용하여 반복에 사용되는 문자를 구분할 수도 있습니다.예를 들어 **ab(xy){3}z는 abxyxyxyz**와 일치합니다.
|	대체	분리되는 식 중 하나와 일치합니다.예를 들어 **dog|cat은** dog 또는 cat과 일치합니다.
?	물음표	이전 식의 0 또는 1이 있음을 나타내는 한정자입니다.예를 들어 lo?**se는 lse 또는 lose**와 일치합니다. **참고:** Ctrl+V를 입력한 다음 물음표를 입력하거나 도움말 기능을 호출해야 합니다.
*	별표	이전 식이 0, 1 또는 임의의 숫자임을 나타내는 한

		정자입니다.예를 들어 **lo*se는 lse**, lose, loose 등과 일치합니다.
{x}	반복 한정자	정확히 x번 반복합니다 .예를 들어 **ab(xy){3}z는 abxyxyxyz**와 일치합니다.
{x,}	최소 반복 한정자	최소 x번 반복합니다.예를 들어 **ab(xy){2,}z는 abxyxyz**, abxyxyxyz 등과 일치합니다.
[abc]	문자 클래스	대괄호의 모든 문자와 일치합니다.예를 들어 **[abc]는** a, b 또는 c와 일치합니다.
[^abc]	부정 문자 클래스	대괄호 안에 포함되지 않은 단일 문자와 일치합니다.예를 들어 **[^abc]는** a, b 또는 c 이외의 문자와 일치합니다.**[^A-Z]**는 대문자가 아닌 단일 문자와 일치합니다.
[a-c]	문자 범위 클래스	범위의 모든 문자와 일치합니다.**[a-z]는** 소문자와 일치합니다.문자 및 범위를 혼합할 수 있습니다 .**[abcq-z]**는 a, b, c, q, r, s, t, u, v, w, x, y, z 등과 **[a-cq-z]** 일치합니다.대시(-) 문자는 대괄호 안의 마지막 또는 첫 번째 문자인 경우에만 리터럴입니다 .**[abc-]** 또는 **[-abc]**.
""	따옴표	문자열의 후행 또는 선행 공백을 유지합니다.예를 들어 **" test"는** 일치 항목을 찾을 때 선행 공백을 유지합니다.
^	캐럿	줄의 시작 부분을 지정합니다.
\ \ \ \ \ \ \ \\ \ \\ \ \\ \\ \\\ \ \\\ \\\ \\ \\\ \\\\ \\\ \\\\ \\ \\\ \\\\\ \\\\\\ \\\\\ \\\\ \\\ \\\\ \\\ \\\\ \\\\\ \\\ \\ \\ \\ \\ \\\ \\ \\ \\ \\\\\\\ \\\\\\\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\\ \\ \\\\\\\\ \\ \\ \\ \\ \\ \\\\ \	이스케이프 문자	메타 문자와 함께 사용하면 리터럴 문자와 일치합니다.예를 들어, **\[**왼쪽 대괄호와 일치합니다.
문자	문자	문자가 메타 문자가 아닌 경우 리터럴 문자와 일치

		합니다.
\r	캐리지 리턴	캐리지 리턴 0x0d와 일치시킵니다.
\n	새 줄	새 줄 0x0a와 일치
\t	탭	탭 0x09와 일치
\f	서식	0x0c 양식 피드와 일치시킵니다.
\xNN	이스케이프된 16진수 숫자	정확히 두 자릿수의 16진수를 사용하는 ASCII 문자와 일치합니다.
\NNN	이스케이프된 8진수 번호	정확히 3자리의 8진수로 ASCII 문자와 일치합니다 .예를 들어 문자 040은 공백을 나타냅니다.

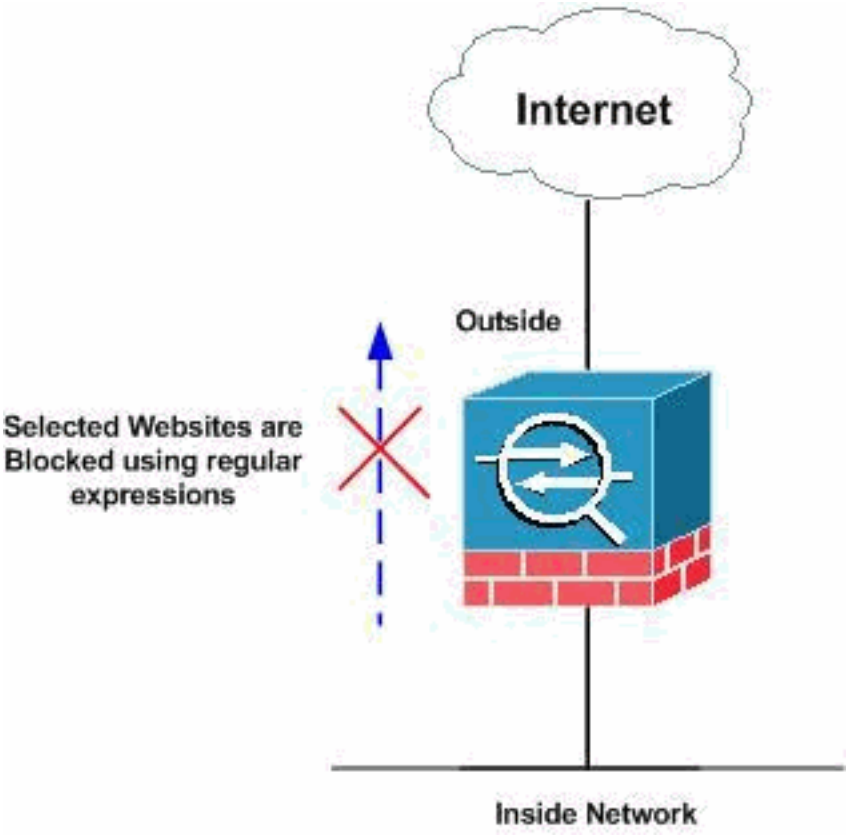
구성

이 섹션에서는 이 문서에 설명된 기능을 구성하는 정보를 제공합니다.

참고: 이 섹션에 사용된 명령에 대한 자세한 내용을 보려면 명령 조회 도구(등록된 고객만 해당)를 사용하십시오.

네트워크 다이어그램

이 문서에서는 다음 네트워크 설정을 사용합니다.



구성

이 문서에서는 다음 구성을 사용합니다.

- ASA CLI 컨피그레이션
- ASA Configuration 8.x with ASDM 6.x

ASA CLI 컨피그레이션

ASA CLI 컨피그레이션

```
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 by web
browser must be either 1.0 or 1.1 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 by web
browser must be either !--- 1.0 or 1.1 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 version being
used by web browser must be either 1.0 or 1.1 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 by web
browser must be either 1.0 or 1.1 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 !---
expressions 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
```

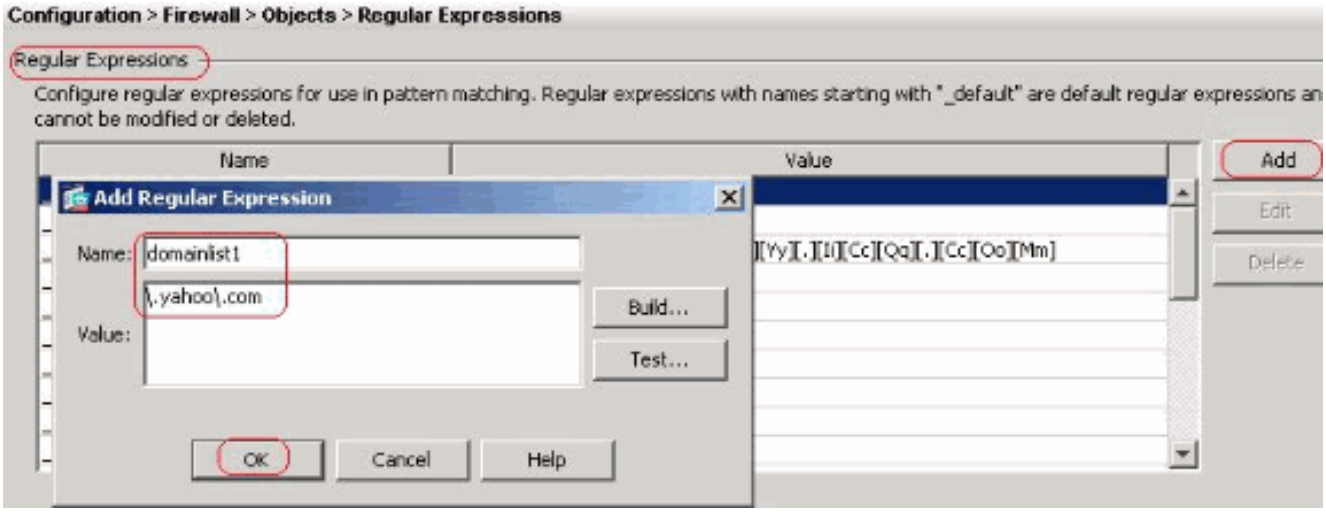
```
!--- Define the actions such as drop, reset or log !---
in the inspection policy map. policy-map global_policy
class inspection_default inspect dns preset_dns_map
inspect ftp inspect h323 h225 inspect h323 ras inspect
netbios inspect rsh inspect rtsp inspect skinny inspect
esmtp inspect sqlnet inspect 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 the !---
inside network traffic. ! 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#
```

ASA Configuration 8.x with ASDM 6.x

정규식을 구성하고 MPF에 적용하여 표시된 대로 특정 웹 사이트를 차단하려면 다음 단계를 완료합니다.

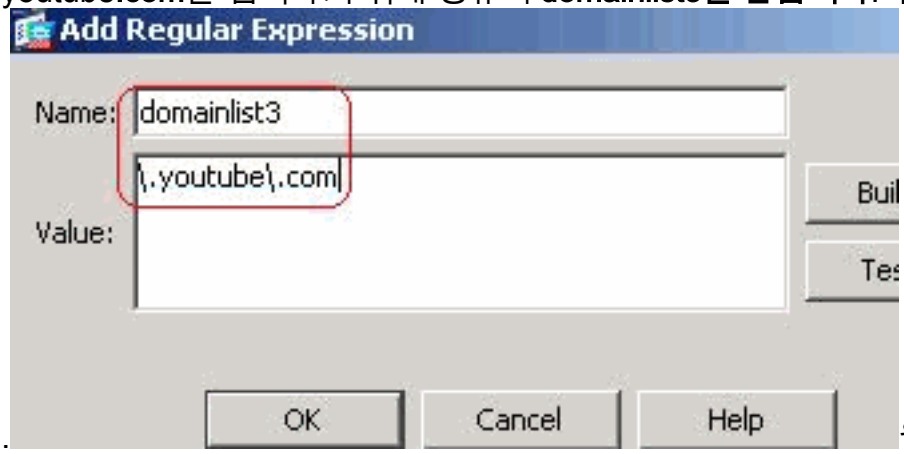
1. **정규식 만들기**Configuration(**컨피그레이션) > Firewall(방화벽) > Objects(개체) > Regular Expressions(정규식)를** 선택하고 **Regular Expression**(정규식) 탭 아래 **Add(추가)를** 클릭하여 표시된 대로 정규식을 생성합니다.도메인 이름 yahoo.**com**을 캡처하기 위해 정규식 **domainlist1**을 **생성합니다**.**확인**을 **클릭합니다**

.

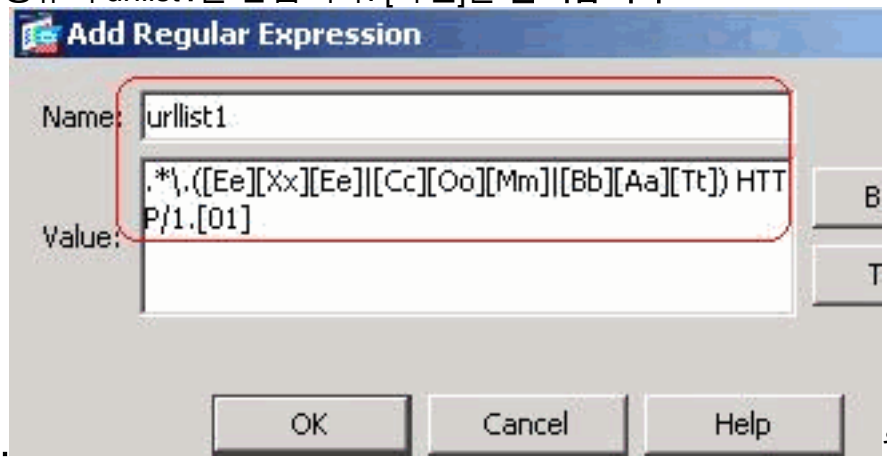


myspace.**com** 도메인 이름**을** 캡처하기 위해 정규식 domainlist**2**를 **만듭니다**.**확인**을 **클릭합니**

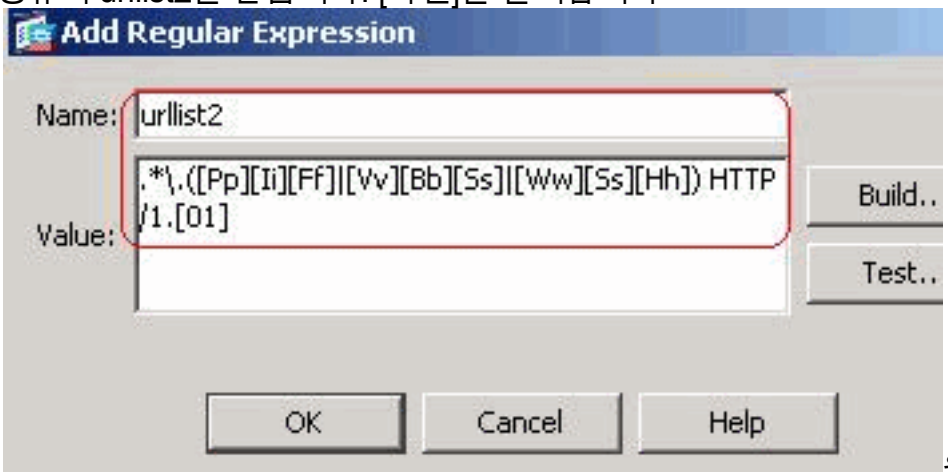
다. 도메인 이름 youtube.com을 캡처하기 위해 정규식 **domainlist3을 만듭니다.확인을 클릭합니다**



. 웹 브라우저에서 사용 중인 http 버전이 1.0 또는 1.1이어야 하는 경우 **exe**, **com** 및 **bat**와 같은 파일 확장명을 캡처하려면 정규식 urllist**1**을 만듭니다. [확인]을 **클릭합니다**

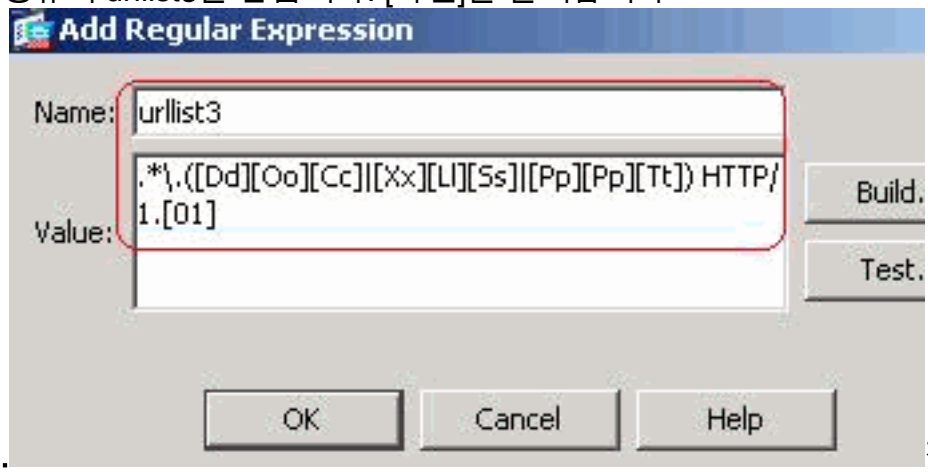


. 웹 브라우저에서 사용 중인 http 버전이 1.0 또는 1.1이어야 하는 경우 **pif**, **vbs** 및 **wsh**와 같은 파일 확장명을 캡처하려면 정규식 urllist**2**를 만듭니다. [확인]을 클릭합니다

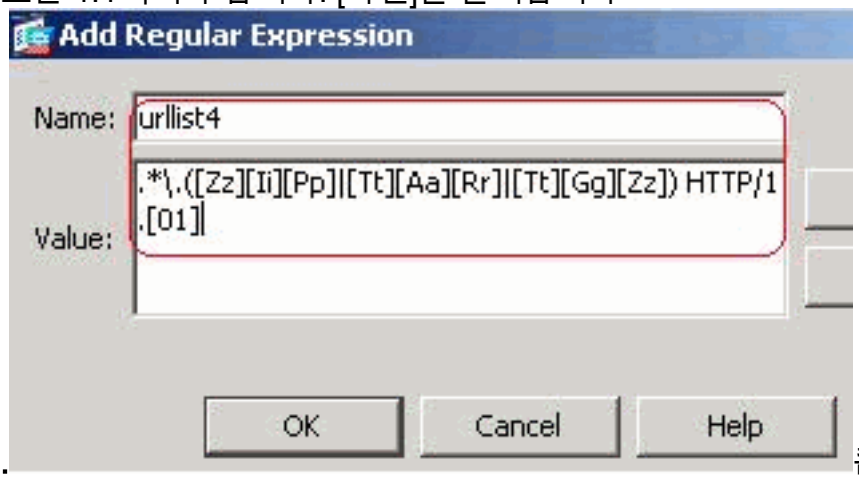


. 웹 브라우저에서 사용 중

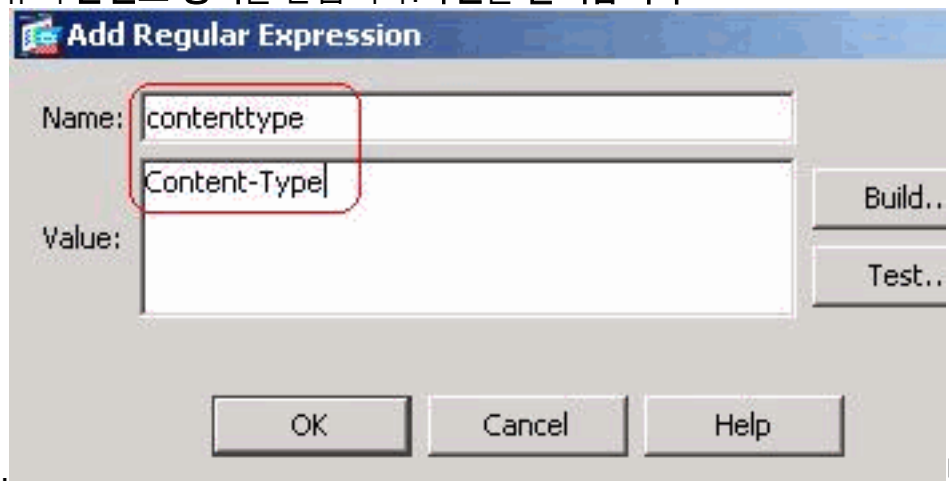
인 http 버전이 1.0 또는 1.1이어야 하는 경우 **doc**, **xls** 및 ppt와 같은 파일 확장명을 캡처하려면 정규식 urllist**3**을 만듭니다. [확인]을 클릭합니다



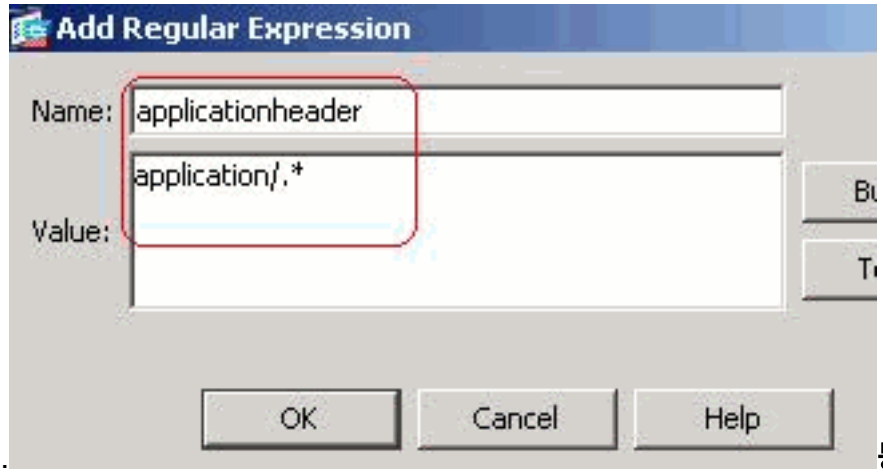
정규식 urllist**4**를 만들어 **zip**, **tar** 및 tgz와 같은 파일 확장명을 캡처합니다. 단, 웹 브라우저에서 사용 중인 http 버전이 1.0 또는 1.1이어야 합니다. [확인]을 클릭합니다



콘텐츠 형식을 캡처하기 위해 정규식 **콘텐츠 형식**을 만듭니다.**확인**을 **클릭합니다**



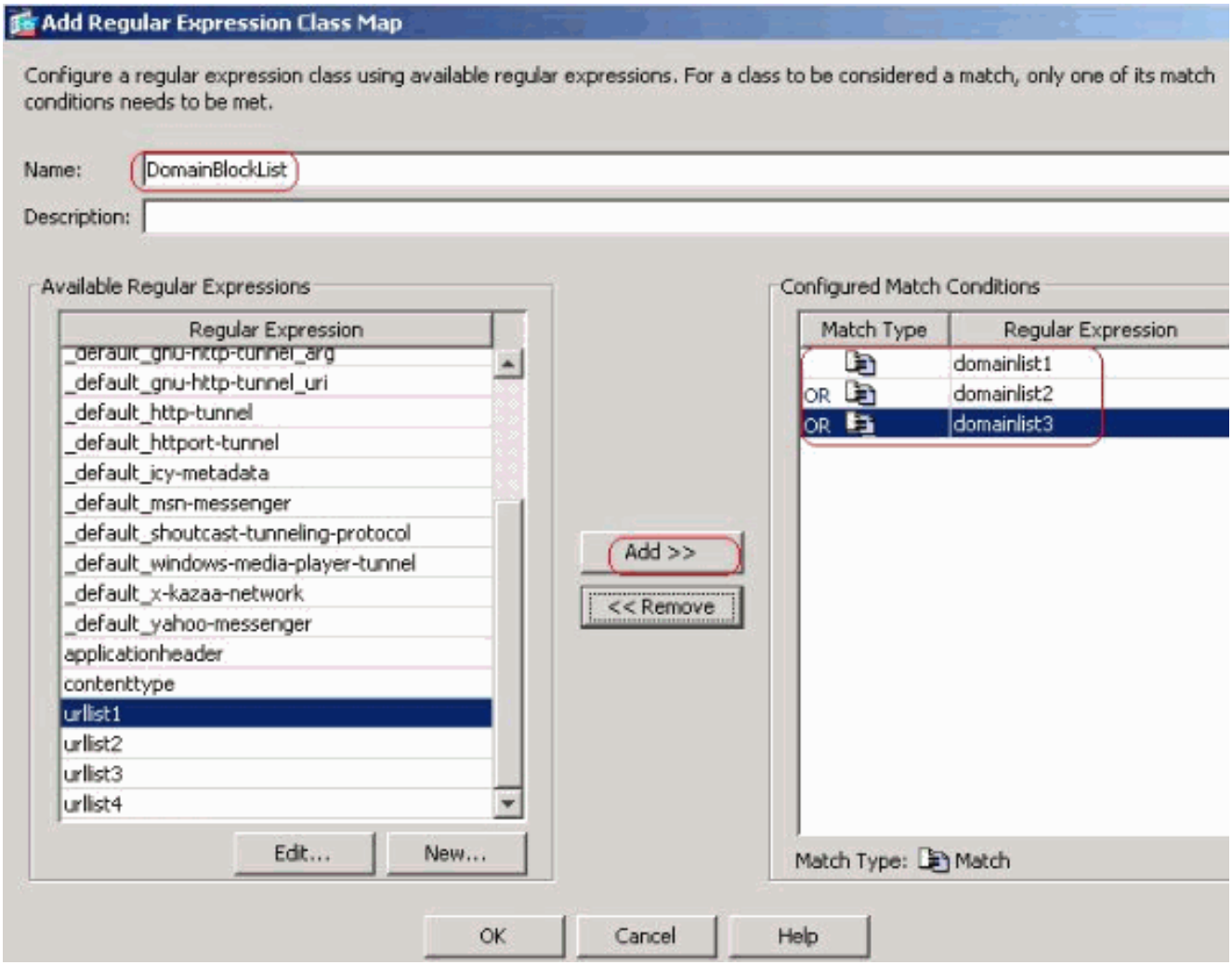
다양한 애플리케이션 헤더를 캡처하기 위해 정규식 **애플리케이션 헤더**를 생성합니다.**확인**을 **클릭합니다**



. **동등한 CLI 컨피그레이션**

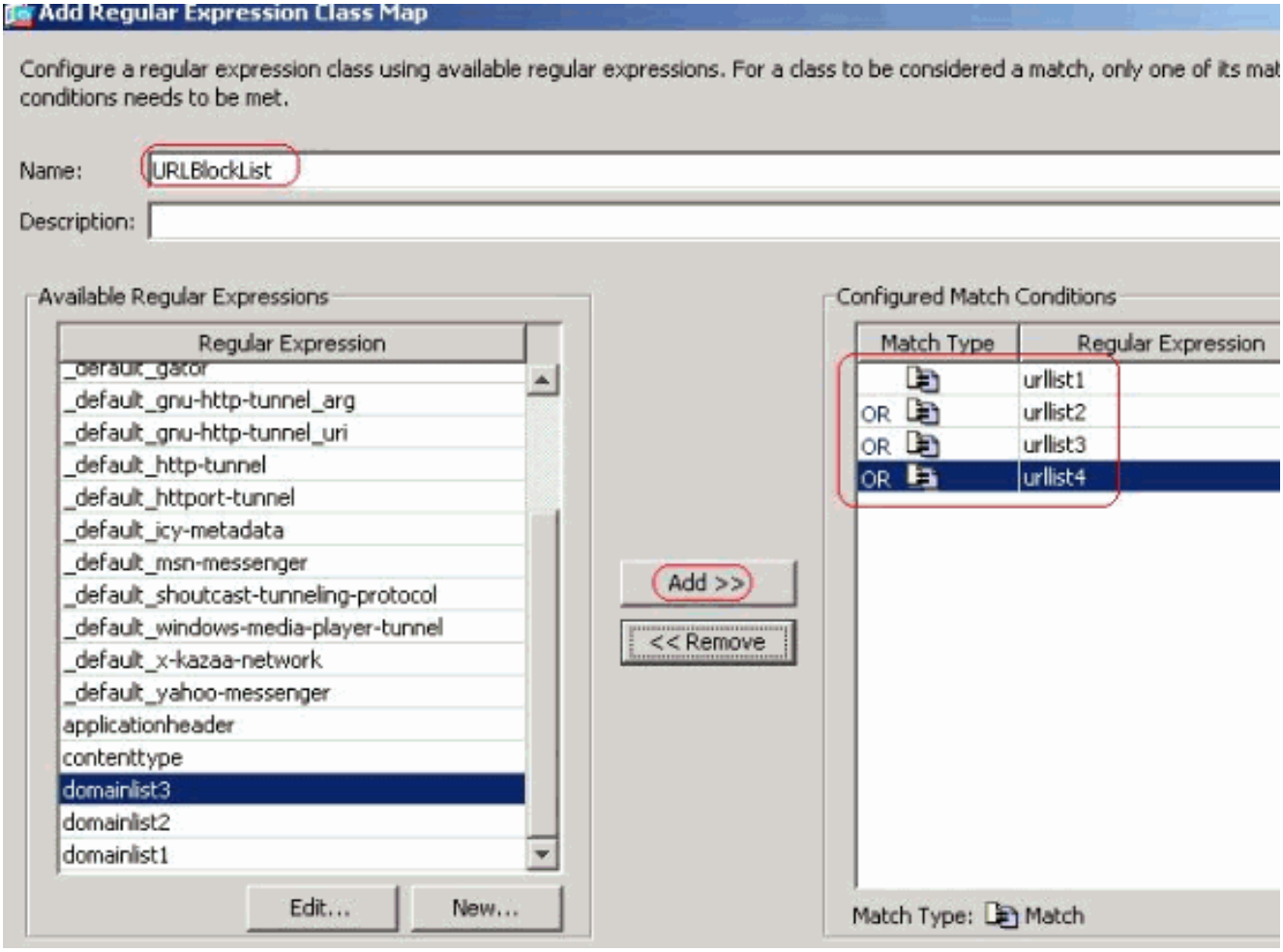
2. **정규식 클래스 만들기**Configuration(**컨피그레이션) > Firewall(방화벽) > Objects(개체) > Regular Expressions(정규식)를** 선택하고 Regular Expression Classes(정규식 클래스) 탭 **아래**에서 Add(추가)를 클릭하여 표시된 대로 다양한 클래스를 생성합니다.정규식 domainlist1, domainlist2 및 domainlist3과 일치시키기 위해 정규식 클래스 DomainBlockList를 만듭니다. **확인**을 클릭합니다

.



정규식 urllist1, urllist2, urllist3 및 urllist4**와 일치시키기** 위해 정규식 클래스 URLBlockList를 만듭니다. **확인**을 클릭합니다

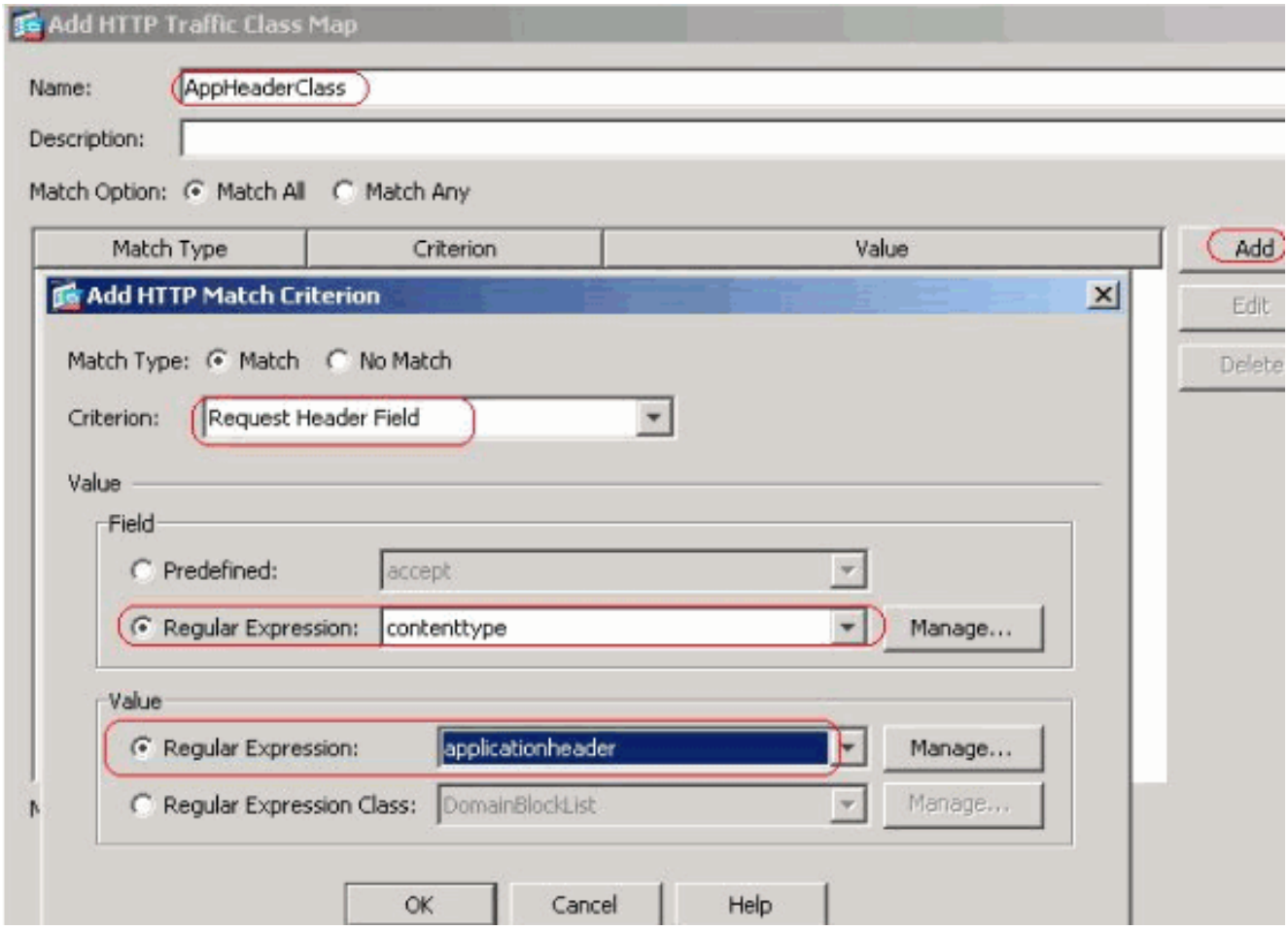
.



동등한 CLI 컨피그레이션

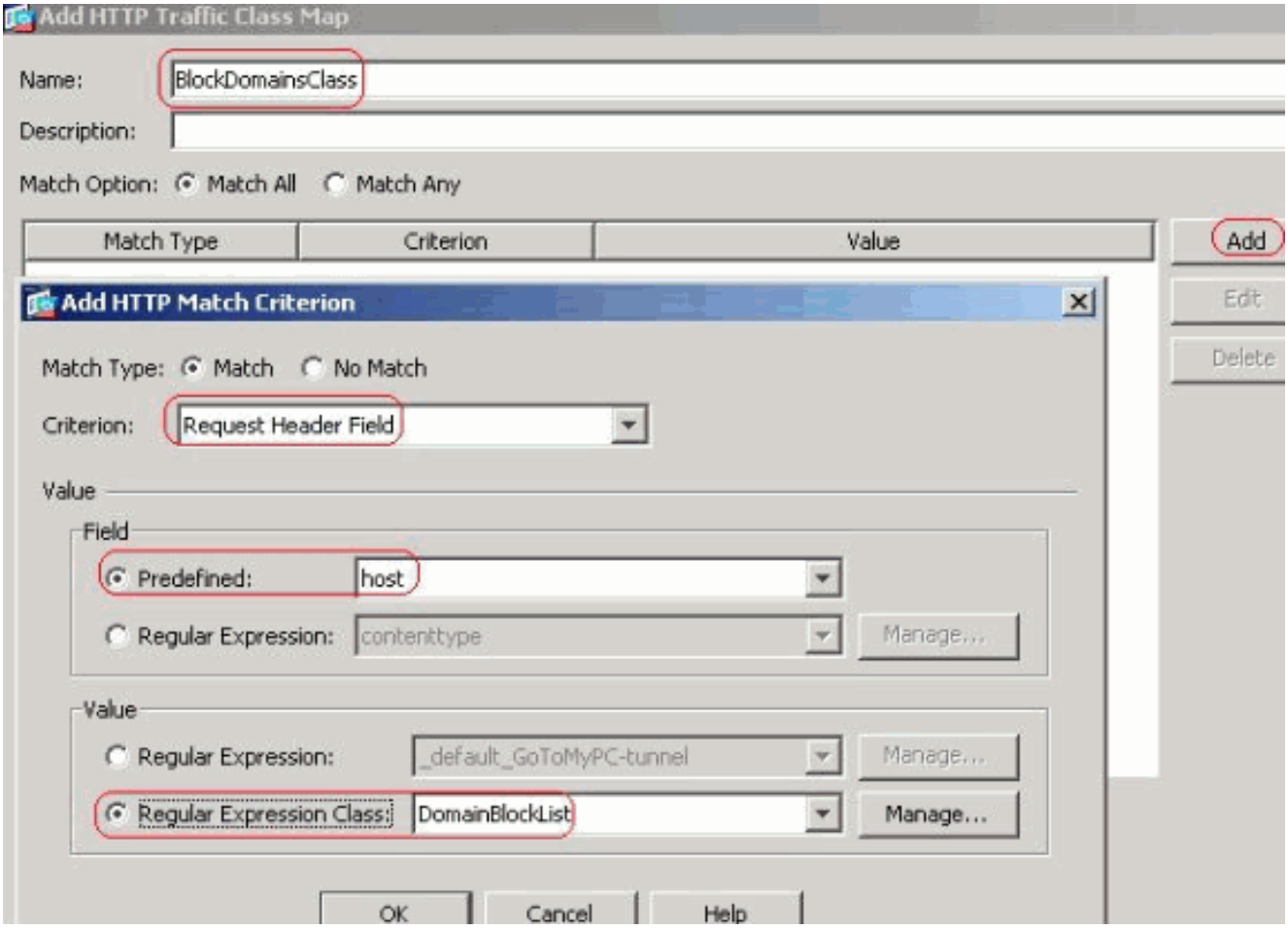
3. **클래스 맵으로 식별된 트래픽 검사**표시된 대로 다양한 정규식으로 식별된 http 트래픽을 검사하기 위한 클래스 맵을 만들려면 Configuration **> Firewall > Objects > Class Maps > HTTP > Add**를 선택합니다.응답 헤더를 정규식 캡처와 일치시키기 위해 클래스 맵 AppHeaderClass를 만듭니다

.



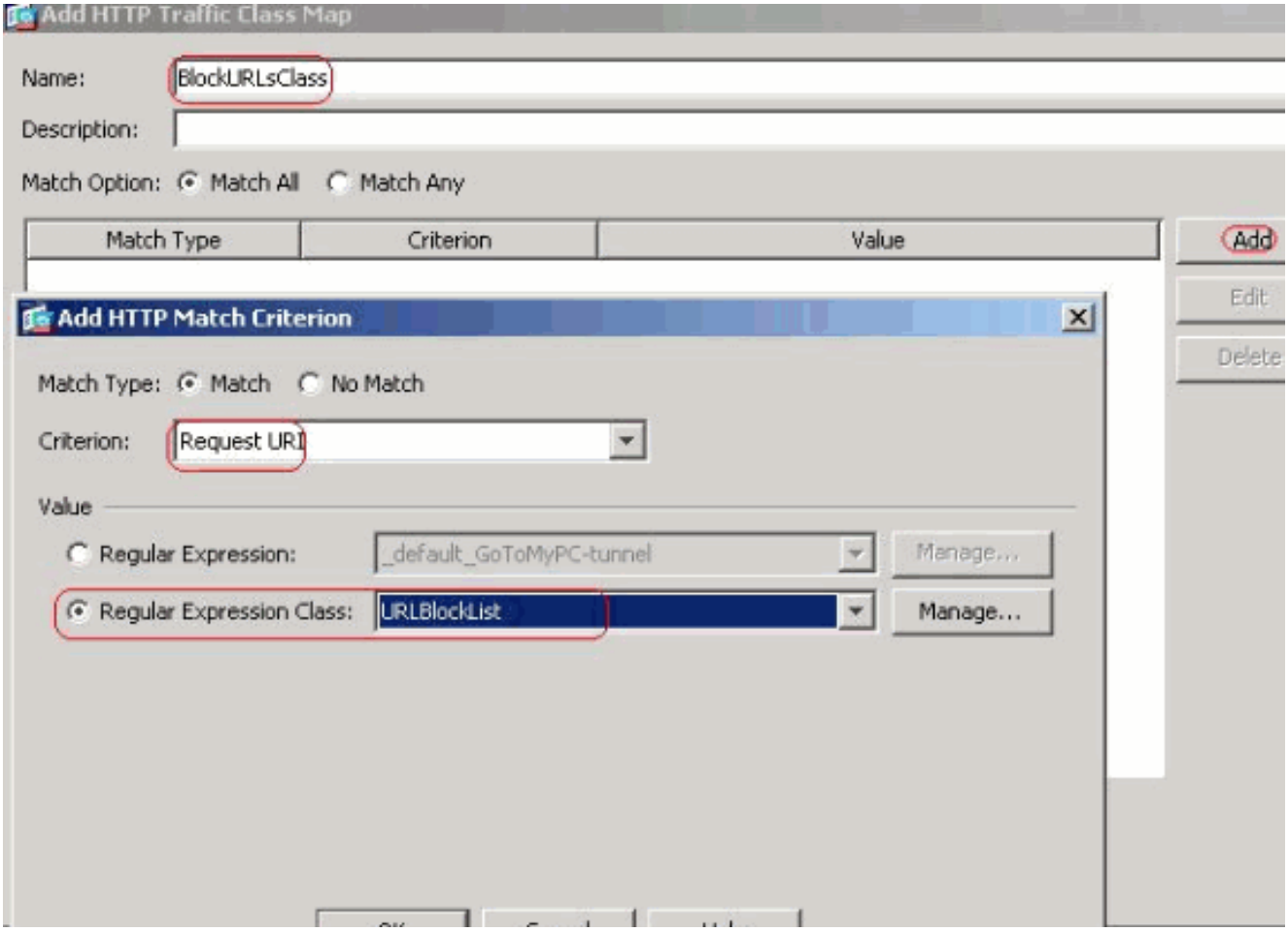
OK(확인)를 **클릭합니다.**요청 헤더를 정규식 캡처와 일치시키기 위해 클래스 맵 BlockDomainsClass를 만듭니다

.



확인을 **클릭합니다**.요청 URI와 정규식 캡처를 일치시키기 위해 클래스 맵 **BlockURLsClass**를 만듭니다

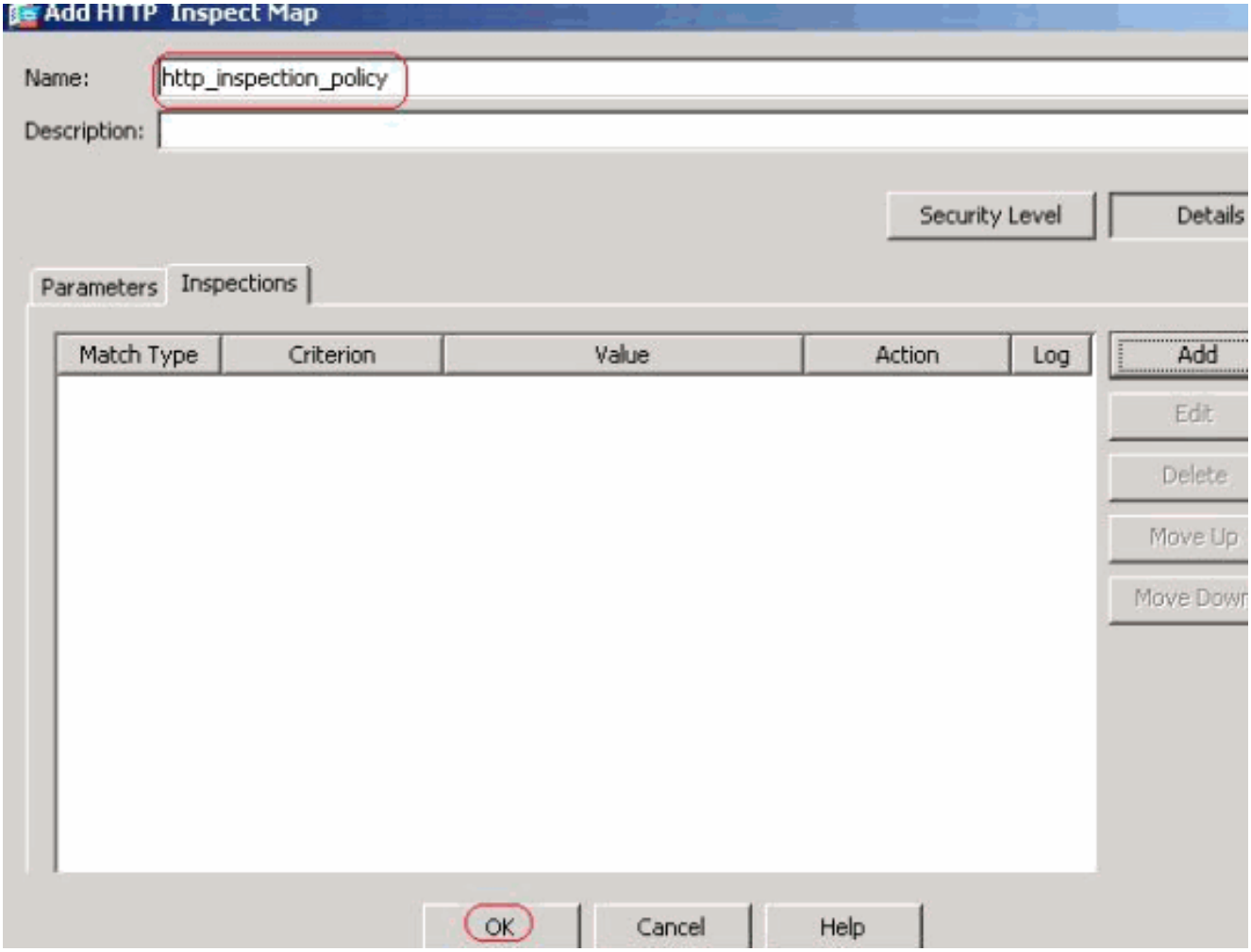
.



확인을 **클릭합니다**.**동등한 CLI 컨피그레이션**

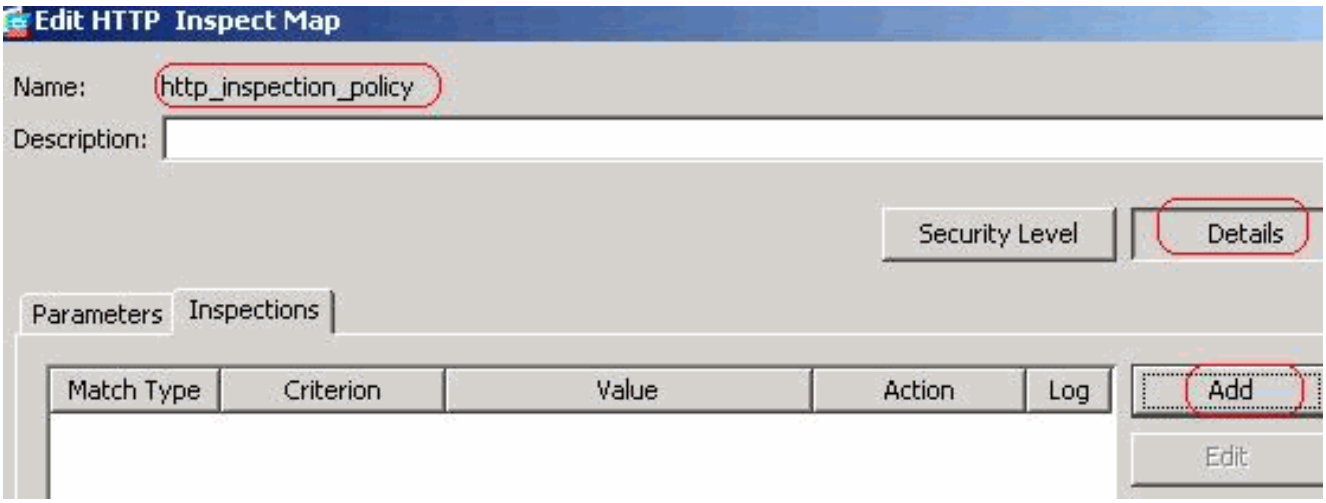
4. **검사 정책에서 일치하는 트래픽에 대한 작업을 설정합니다**.Configuration **> Firewall > Objects > Inspect Maps > HTTP**를 선택하여 표시된 대로 매칭된 트래픽에 대한 작업을 설정하기 위한 **http_inspection_policy**를 생성합니다.**확인**을 **클릭합니다**

.

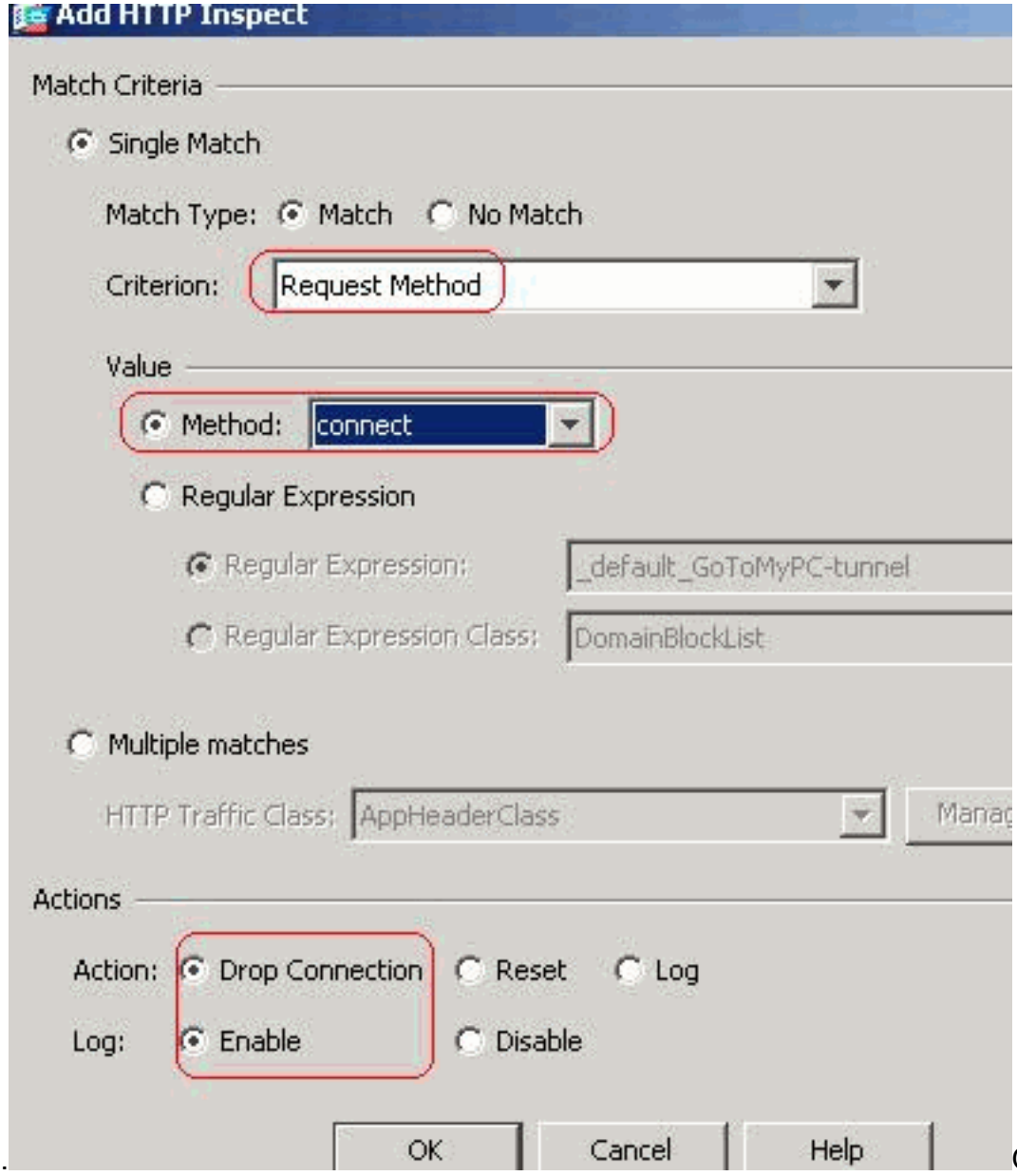


Configuration **> Firewall > Objects > Inspect Maps > HTTP > http_inspection_policy(두 번 클릭)를 선택하고 Details > Add를** 클릭하여 지금까지 생성된 다양한 클래스에 대한 작업을 설정합니다

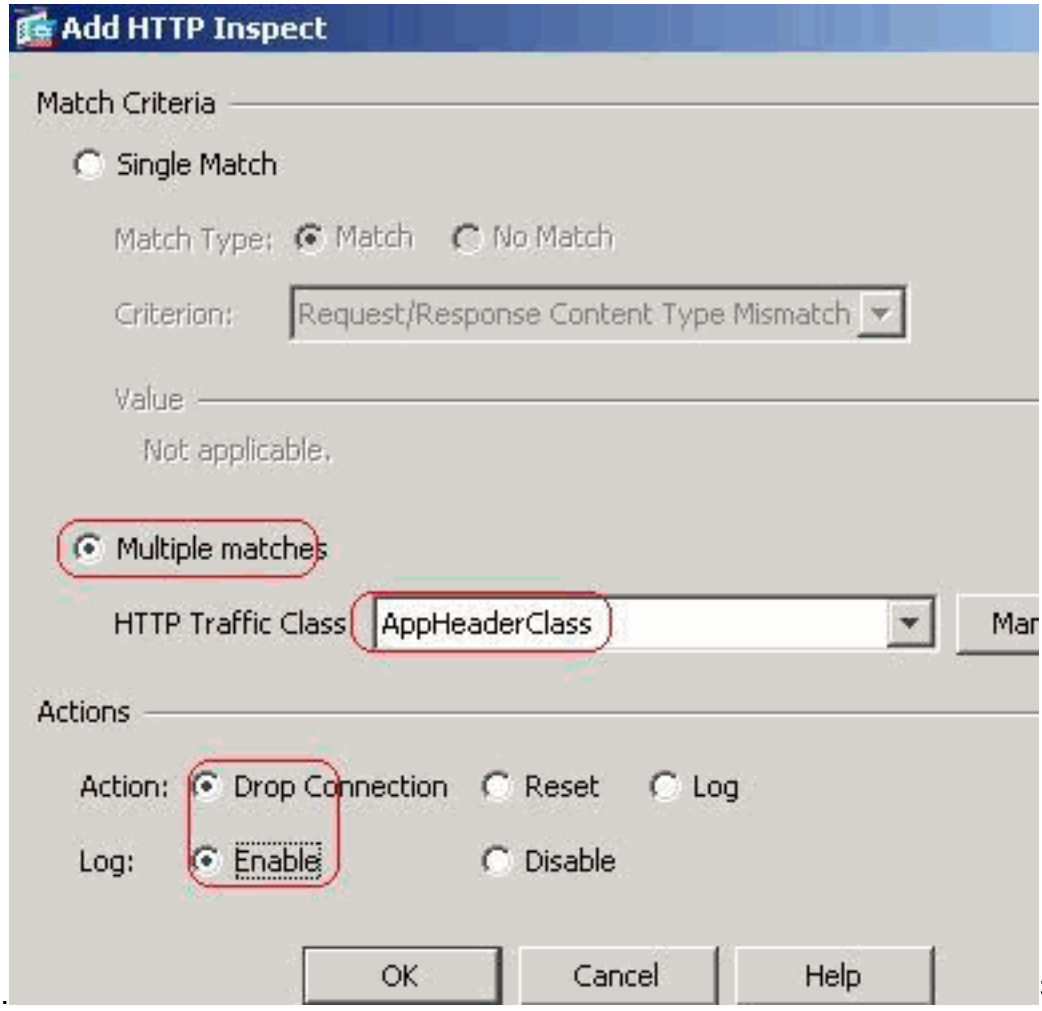
.



작업을 Drop Connection(연결 **삭제**)으로 설정하고 **Criterion**(조건)에 대한 로깅을 Request Method(요청 방법)로 설정하고 Value(값)를 Connect(연결)로 활성화합니다

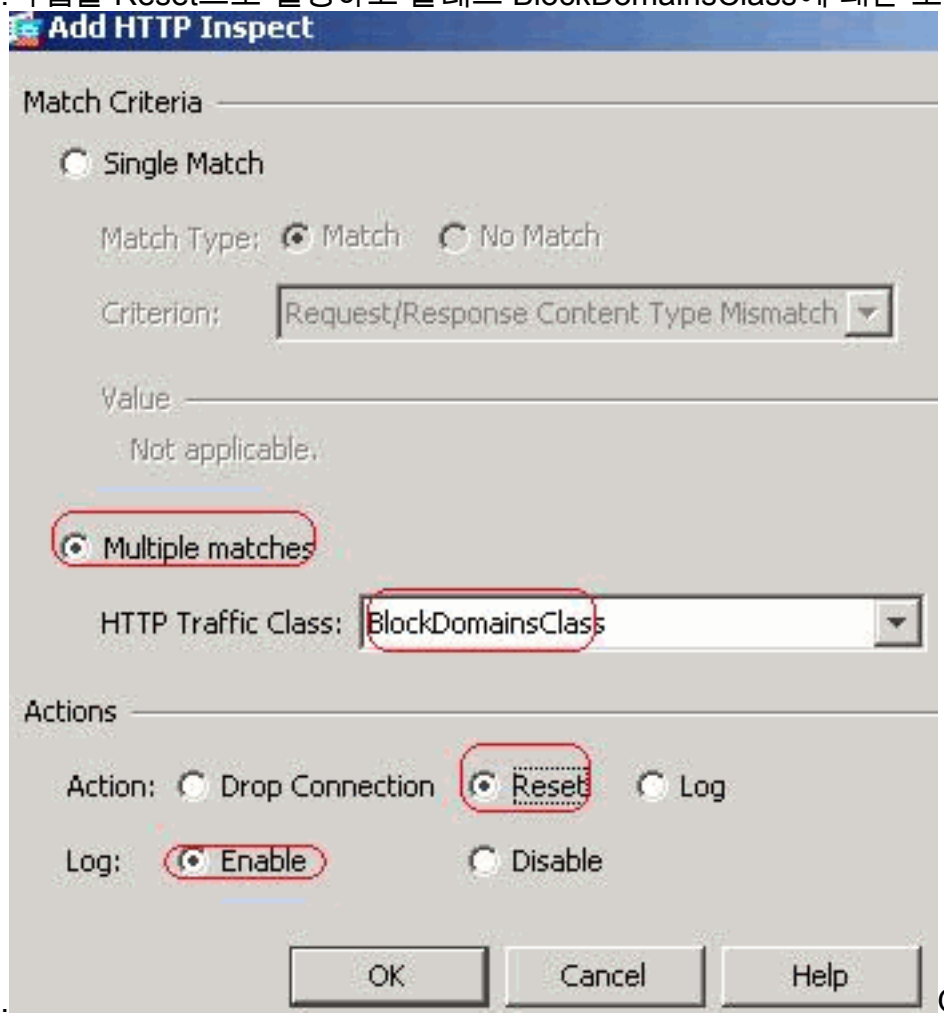


. OK(확인)를 **클릭합니다.**작업을 Drop Connection(연결 **삭제**)으로 설정하고 클래스 AppHeaderClass에 대한 로깅을 **활성화합니다**



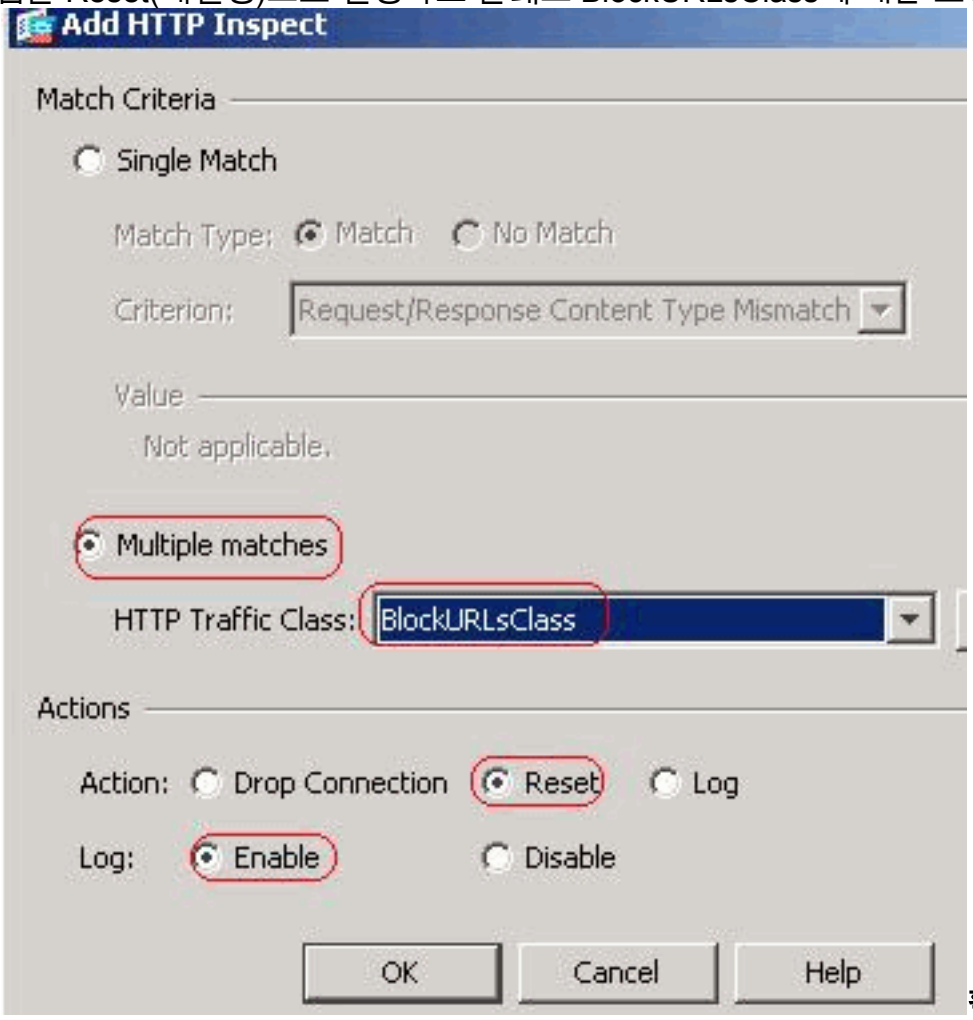
. **확인을 클릭합니다**

.작업을 Reset으로 설정하고 클래스 BlockDomainsClass에 대한 로깅**을 활성화합니다**



. OK(확인)를 **클릭합니다.**작

업을 Reset(재설정)으로 설정하고 클래스 BlockURLsClass에 대한 로깅을 **활성화합니다**

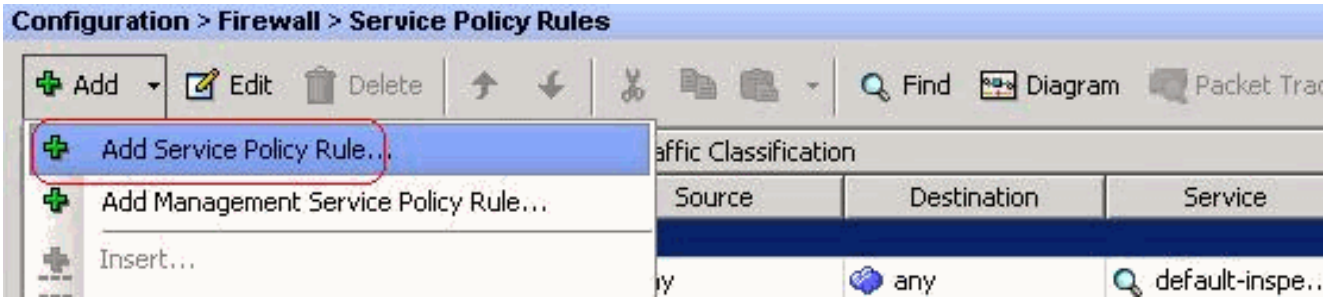


. **확인을 클릭합니다**

.Apply를 **클릭합니다.동등한 CLI 컨피그레이션**

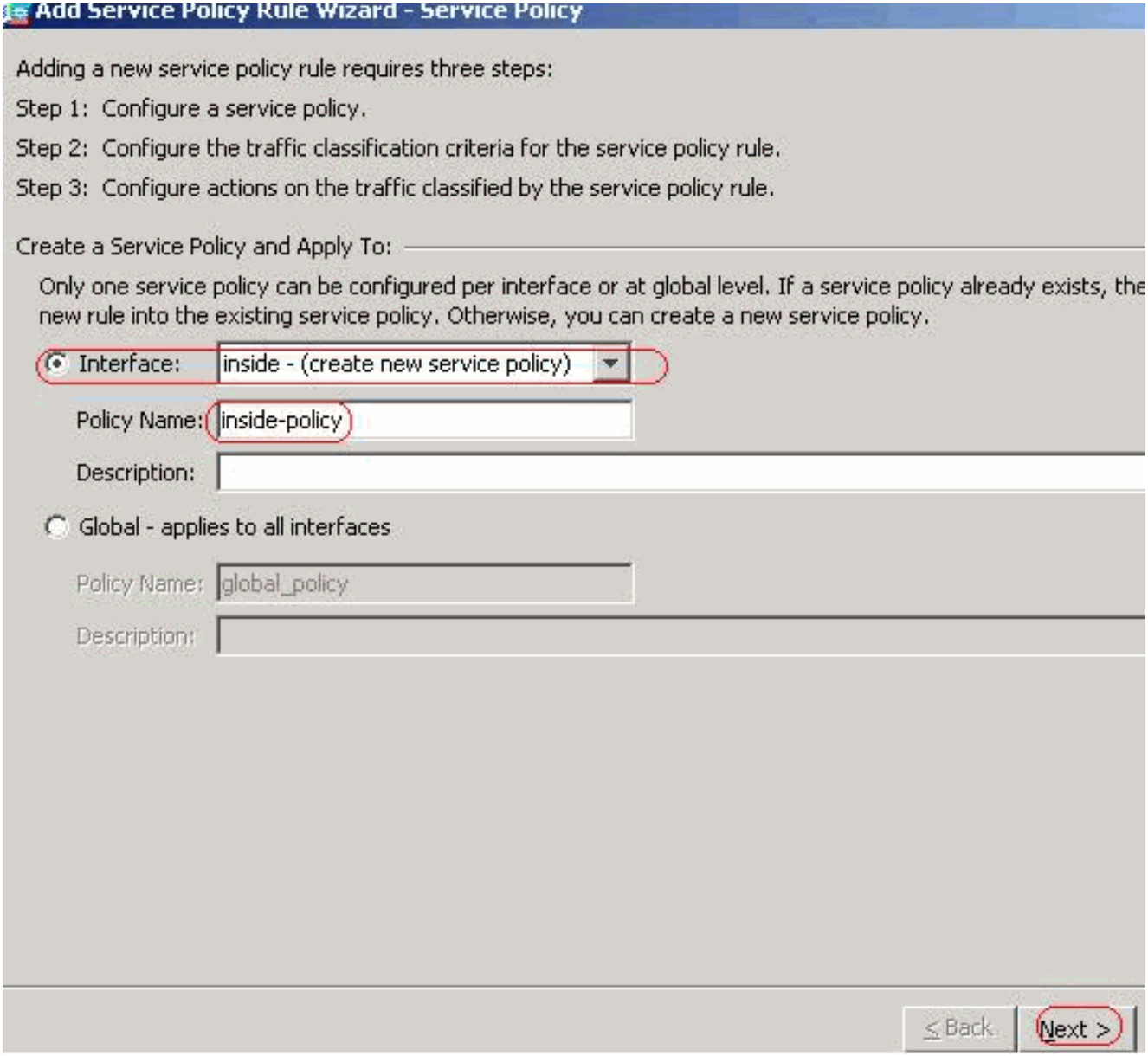
5. **인터페이스에 검사 http 정책 적용**Configuration **> Firewall > Service Policy Rules > Add > Add Service Policy Rule을 선택합니다**

.



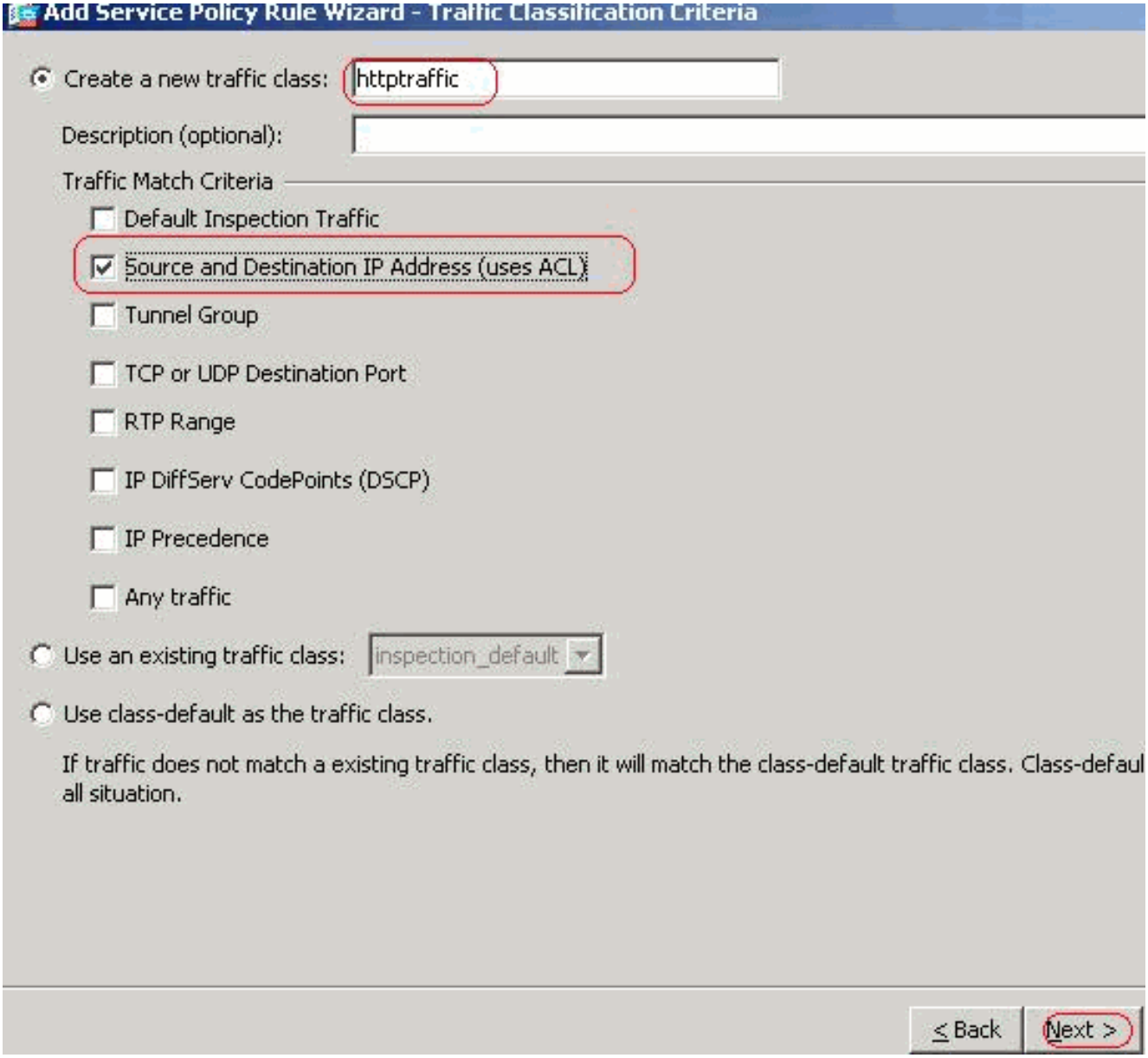
HTTP 트래픽드롭다운 메뉴에서 내부 인터페이스가 있는 **Interface** 라디오 버튼을 선택하고 Policy Name as **inside-policy를 선택합니다**.Next(**다음)를 클릭합니다**

.



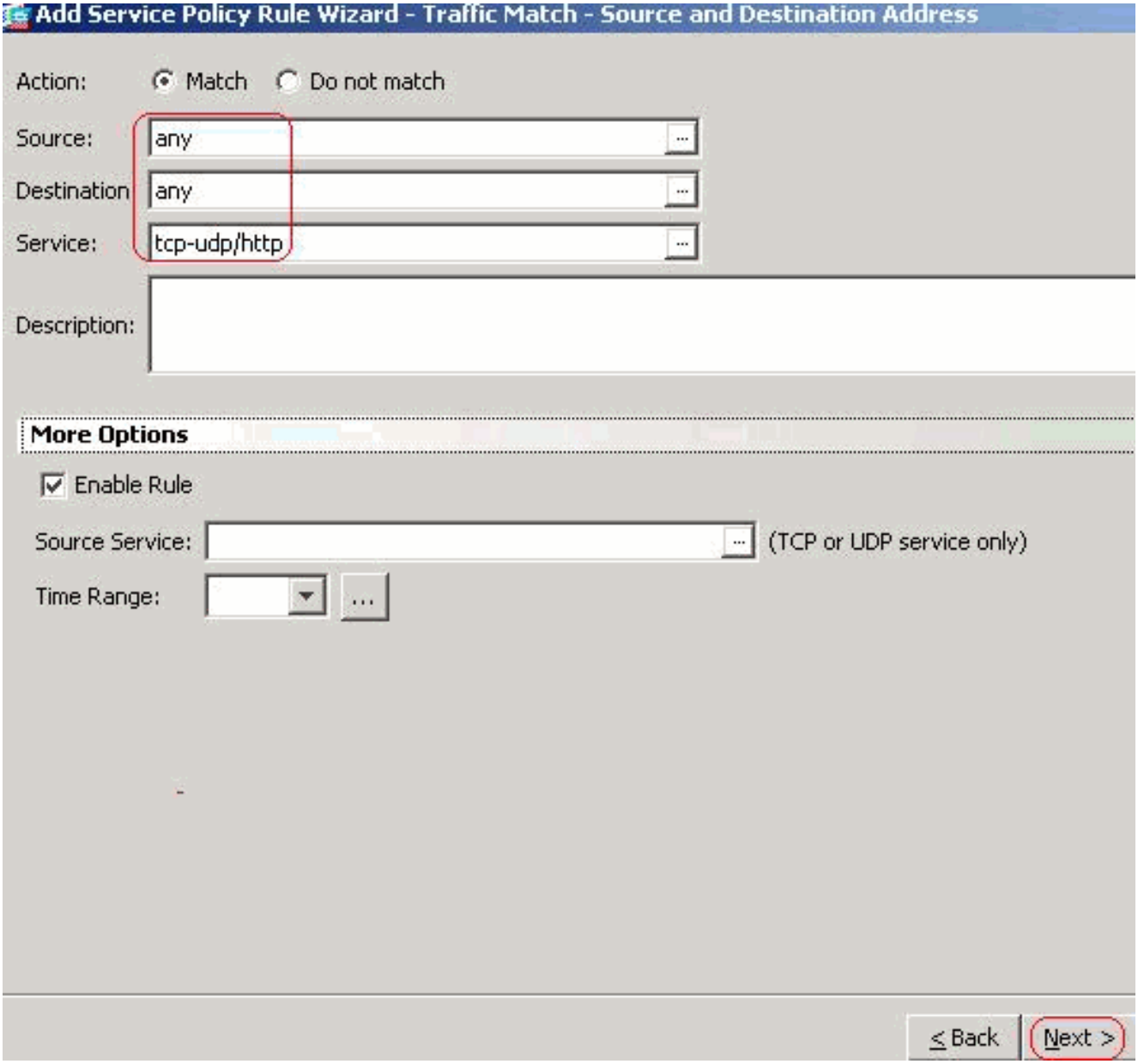
클래스 맵 httptraffic을 **생성하고 Source** 및 Destination IP Address**(ACL 사용)를 확인합니다**. Next(**다음)를 클릭합니다**

.

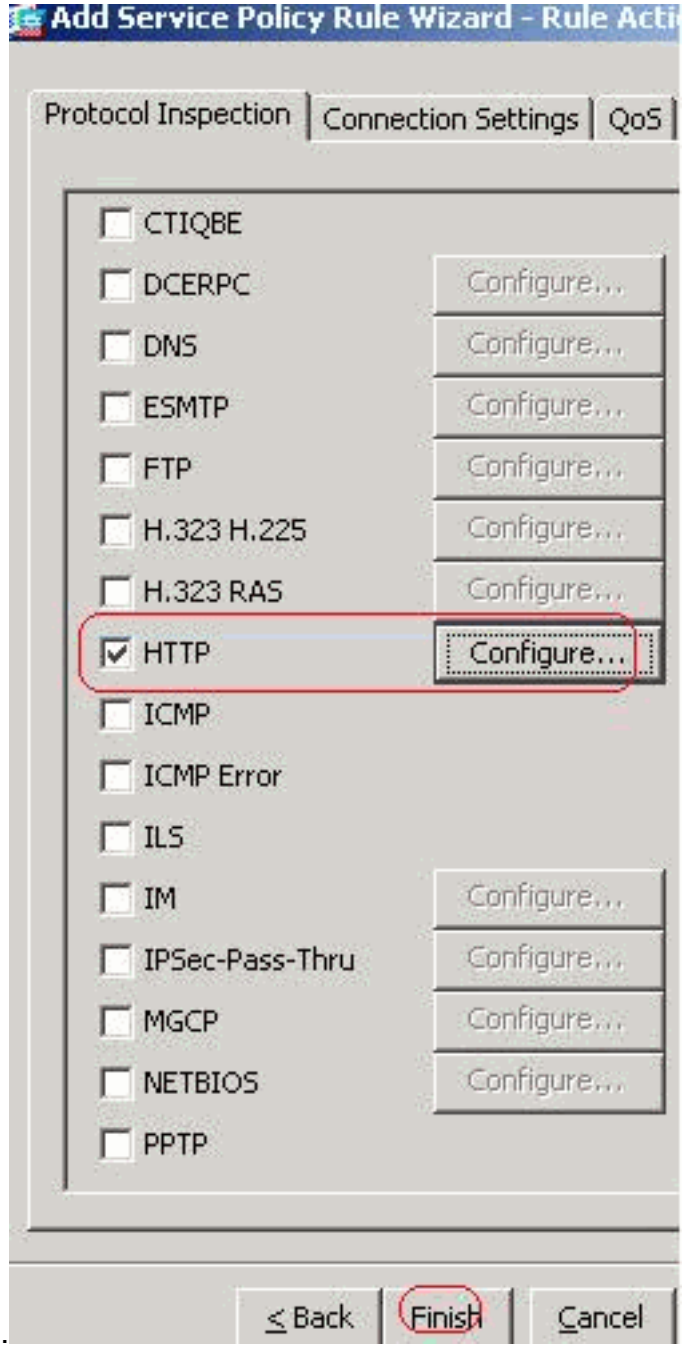


Source(소스) 및 Destination(대상)을 **tcp-udp/http**와 같은 서비스로 **선택합니다**.Next(**다음)를 클릭합니다**

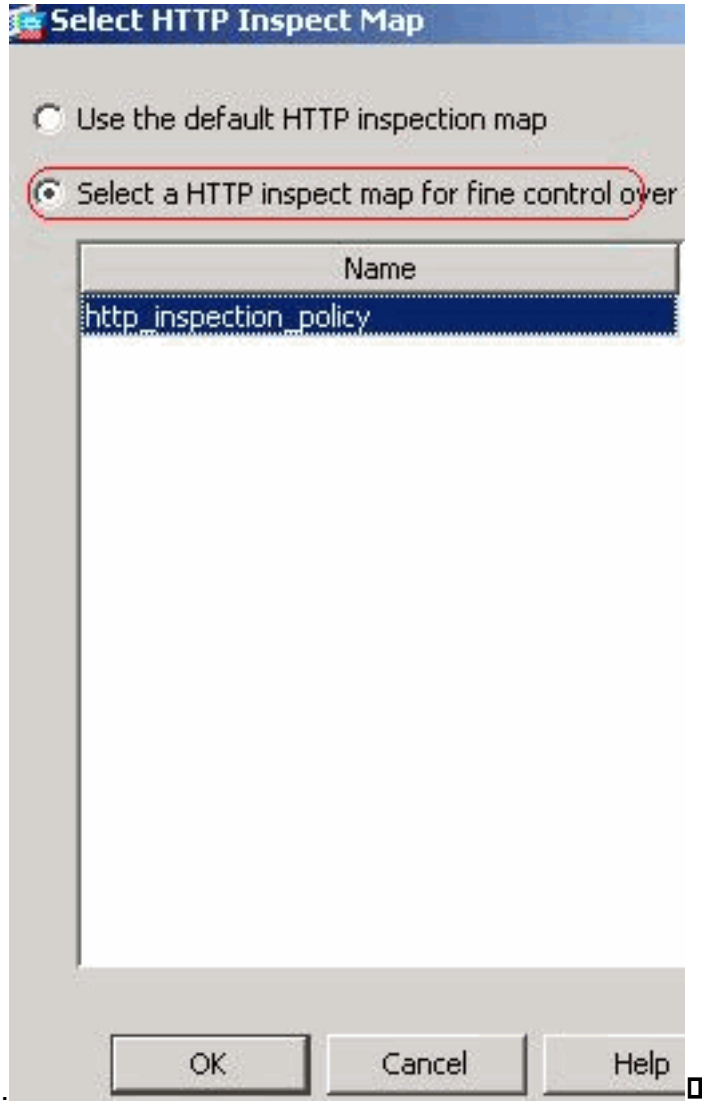
.



HTTP 라디오 버튼을 선택하고 Configure를 **클릭합니다**

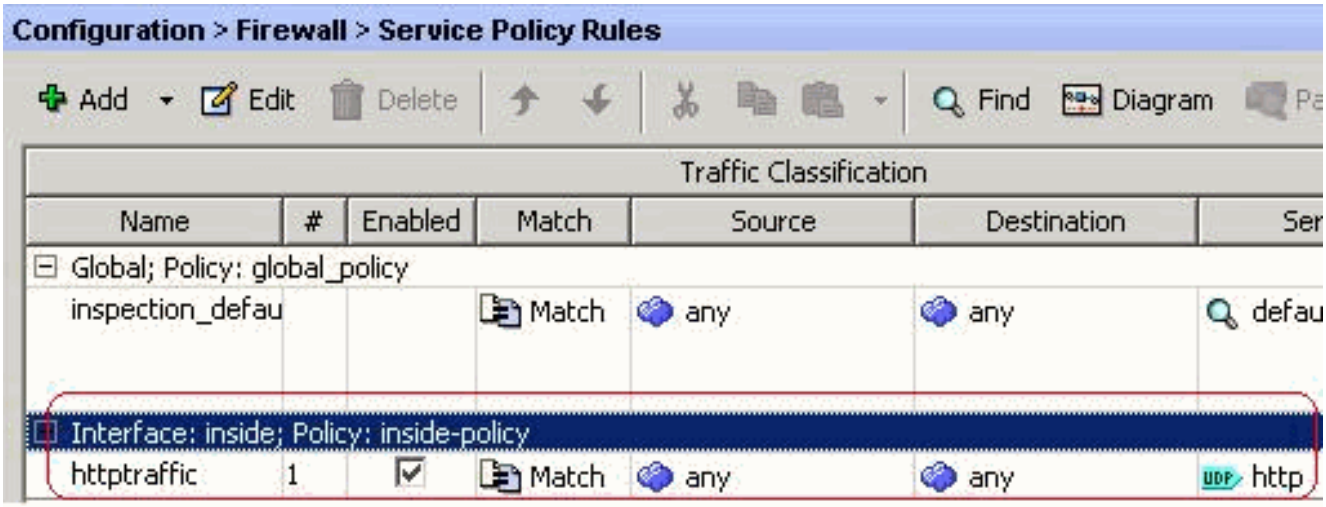


. 라디오 버튼 Select a **HTTP inspect map for the control over inspection(HTTP 검사 맵 선택)을** 참조하십시오.**확인을 클릭합니다**



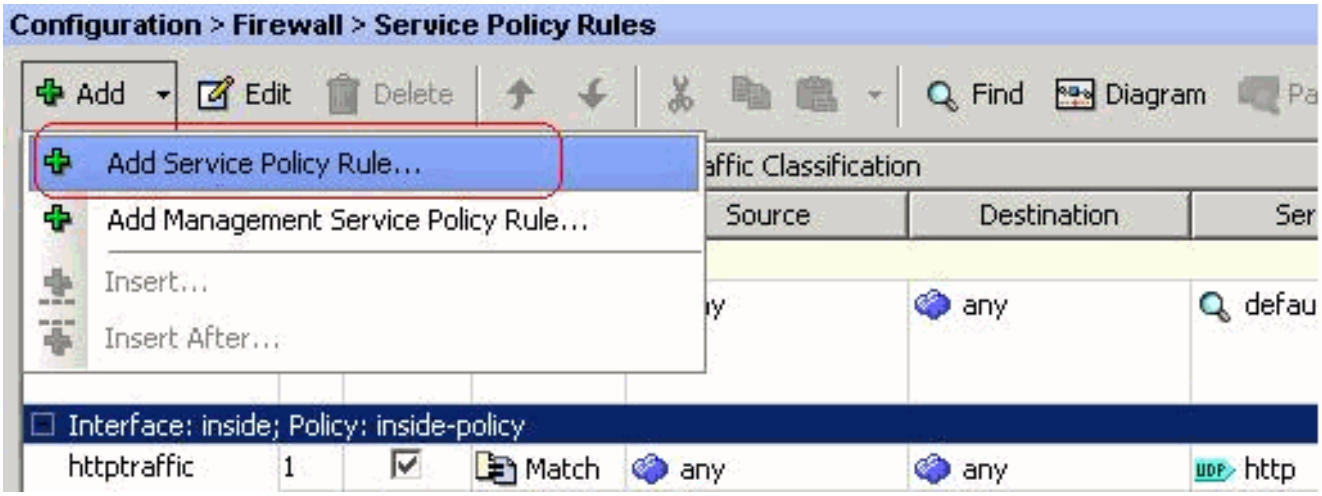
. **마침을 클릭합니다**

.



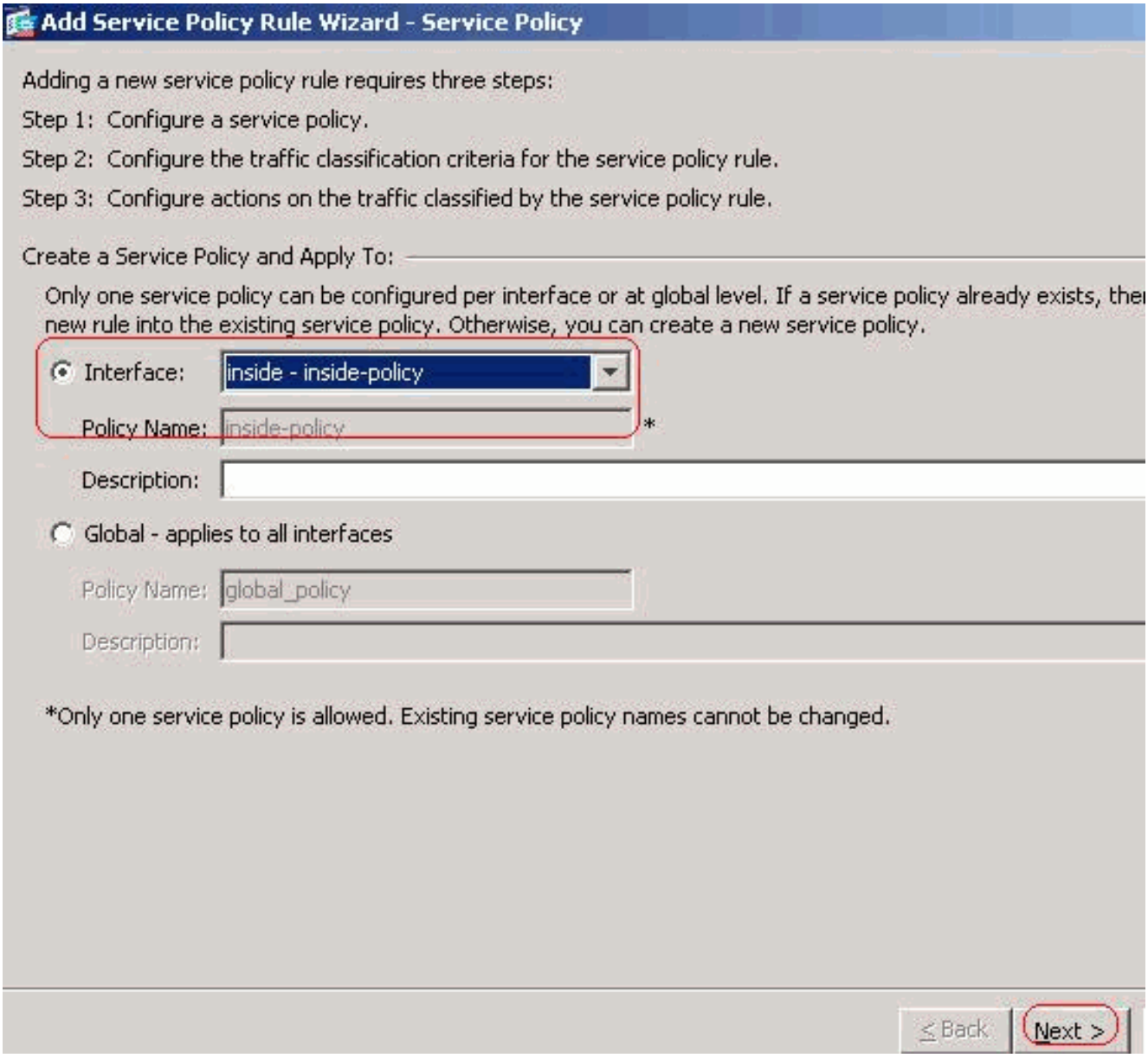
포트 8080 트래픽다시 Add(추가) **> Add Service Policy Rule(서비스 정책 규칙 추가)을** 선택합니다

.



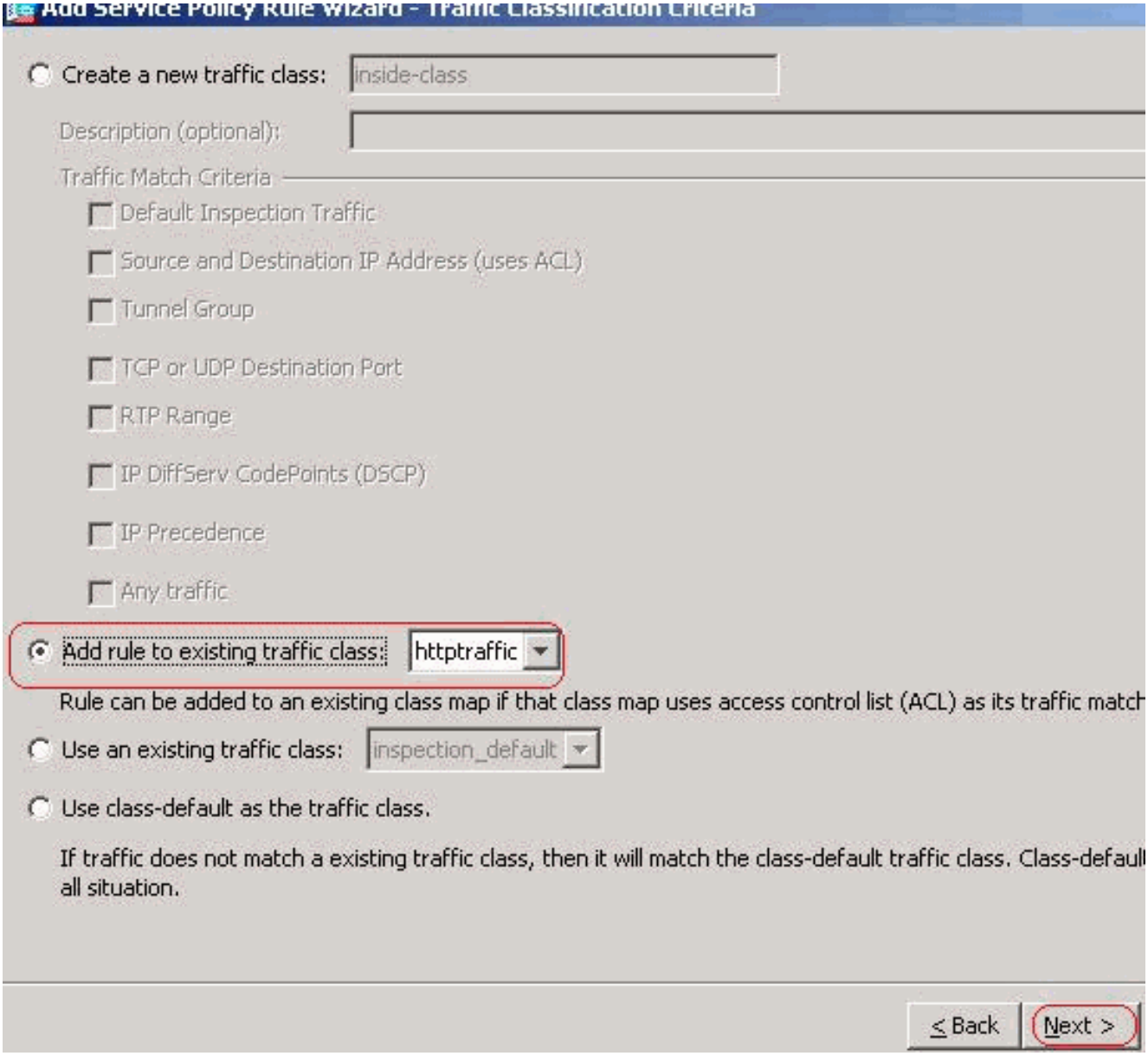
Next(**다음)를 클릭합니다**

.



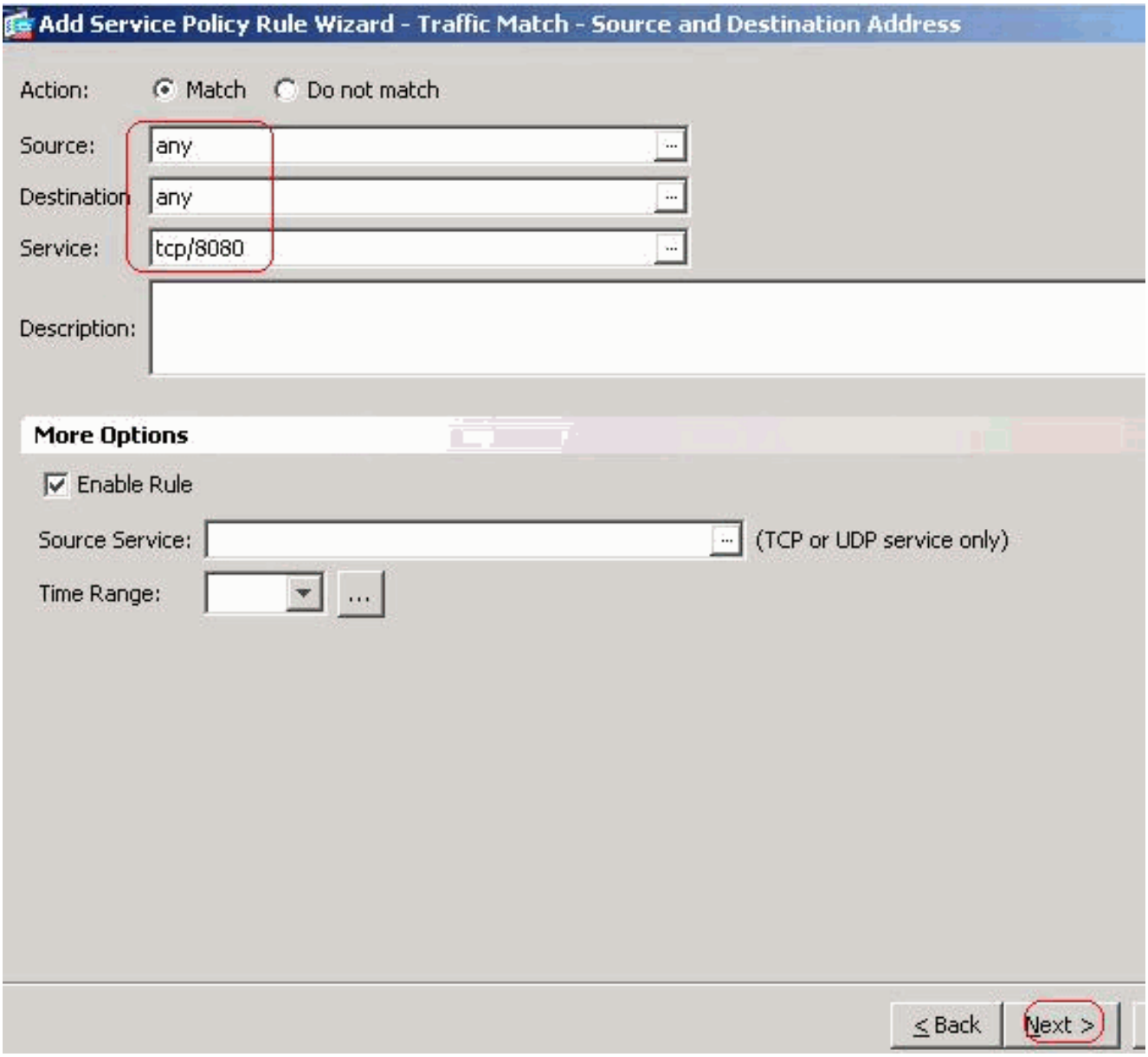
라디오 버튼 **Add rule to existing traffic class(기존 트래픽 클래스에 규칙 추가)**를 선택하고 드롭다운 메뉴에서 httptraffic을 선택합니다.Next(**다음)를 클릭합니다**

.



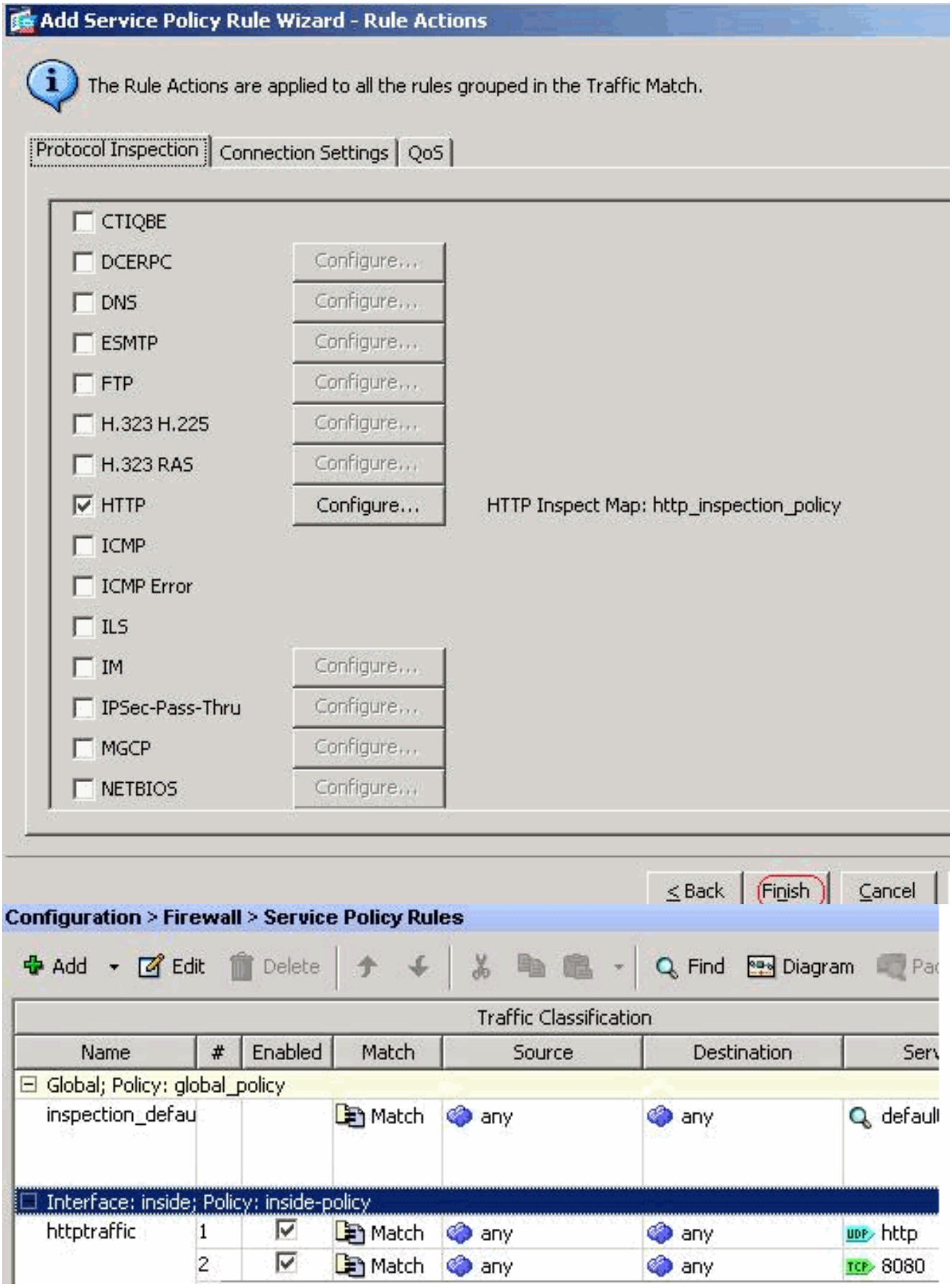
tcp/8080을 사용하여 소스 및 대상**을** 임의**로** 선택하고 **다음**을 클릭합니다

.


Add Service Policy Rule Wizard - Traffic Match - Source and Destination Address
Action:
Match
Do not match
Source:
any
Destination:
any
Service:
tcp/8080
Description:
More Options
Enable Rule
Source Service:
(TCP or UDP service only)
Time Range:
...
< Back
Next >

마침을 클릭합니다

.



Apply를 **클릭합니다.동등한 CLI 컨피그레이션**

다음을 확인합니다.

이 섹션을 사용하여 컨피그레이션이 제대로 작동하는지 확인합니다.

Output Interpreter 도구(등록된 고객만 해당)(OIT)는 특정 **show** 명령을 지원합니다.OIT를 사용하여 **show** 명령 출력의 분석을 봅니다.

- **show running-config regex** - 구성된 정규식을 표시합니다.

```
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** - 구성된 클래스 맵을 표시합니다.

```
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** - 구성된 http 트래픽을 검사하는 정책 맵을 표시합니다.

```
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** - 모든 policy-map 컨피그레이션과 기본 policy-map 컨피그레이션을 표시합니다.

```
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** - 현재 실행 중인 모든 서비스 정책 컨피그레이션을 표시합니다.

```
ciscoasa#show running-config service-policy
service-policy global_policy global
service-policy inside-policy interface inside
```

- **show running-config access-list** - 보안 어플라이언스에서 실행되는 access-list 컨피그레이션을 표시합니다.

```
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#
```

문제 해결

이 섹션에서는 컨피그레이션 문제를 해결하는 데 사용할 수 있는 정보를 제공합니다.

참고: debug **명령**을 사용하기 전에 디버그 명령에 대한 중요 정보를 참조하십시오.

- **debug http** - HTTP 트래픽에 대한 디버그 메시지를 표시합니다.

관련 정보

- Cisco ASA 5500 Series Adaptive Security Appliances 지원
- Cisco ASDM(Adaptive Security Device Manager) 지원
- Cisco PIX 500 Series 보안 어플라이언스 지원
- Cisco PIX 방화벽 소프트웨어
- Cisco Secure PIX Firewall 명령 참조
- 보안 제품 필드 알림(PIX 포함)
- RFC(Request for Comments)
- 기술 지원 및 문서 – Cisco Systems