Дизайн и руководство по Zone-Based Policy межсетевому экрану

Содержание

Введение
Предварительные условия
Требования
Используемые компоненты
Условные обозначения
Обзор зональной политики
Модель конфигурации зональной политики
Правила для применения межсетевого экрана зональной политики
Проектирование безопасности сети для зональной политики
Использование IPSec VPN с межсетевым экраном зональной политики
Настройка языка политики Cisco (CPL)
Настройка карт классов межсетевого экрана зональной политики
Настройка карт политик межсетевого экрана зональной политики
Настройка карт параметров межсетевого экрана зональной политики
Применение ведения журнала для политик межсетевого экрана зональной политики
Изменение карт классов и карт политик межсетевого экрана зональной политики
Примеры конфигураций
Межсетевой экран маршрутизации проверки с отслеживанием состояния соединения
Прозрачный межсетевой экран проверки с отслеживанием состояния соединения
Регулировка скорости для межсетевого экрана зональной политики
Фильтрация URL-адресов
Управление доступом к маршрутизатору
Зональный межсетевой экран и Wide-Area Application Services
Наблюдение за межсетевым экраном зональной политики с помощью команд show и debug
Настройка защиты против DoS-атак межсетевого экрана зональной политики
Приложение
Приложение А: Основная конфигурация
Приложение Б: Финальная (полная) конфигурация
Приложение В: Основная конфигурация межсетевого экрана зональной политики для двух зон
Дополнительные сведения

Введение

В состав операционной системы Cisco IOS® версии 12.4(6)T входит межсетевой экран зональной политики (ZFW), новая модель конфигурации для набора функций межсетевого

экрана Cisco IOS. Эта новая модель конфигурации обладает интуитивно понятными политиками для межсетевых экранов с несколькими интерфейсами, повышенной детализацией приложения политики межсетевого экрана и политикой запрета всего трафика между зонами безопасности межсетевого экрана, которая применяется ко всему желательному трафику вплоть до применения явной политики.

Почти все классические функции межсетевого экрана Cisco IOS, реализованные перед появлением Cisco IOS версии 12.4(6)T, поддерживаются новым интерфейсом инспекции зональной политики:

- Динамический анализ пакетов
- Межсетевой экран Cisco IOS, следящий за маршрутизацией и пересылкой
- Фильтрация URL-адресов
- Смягчение последствий DoS-атак

В Cisco IOS версии 12.4(9)T добавлена поддержка ZFW для сеансов или подключений по классам и ограничений пропускной способности, а также проверки приложений и управления ими:

- HTTP
- Протокол POP3, протокол IMAP, протоколы SMTP и ESMTP
- Удаленный вызов процедур Sun (RPC)
- Приложения для обмена мгновенными сообщениями:Microsoft MessengerYahoo! MessengerAOL Instant Messenger
- Обмен файлами в одноранговой сети:BittorrentKaZaAGnutellaeDonkey

В Cisco IOS версии 12.4(11)T добавлена статистика, позволяющая упростить настройку защиты от DoS-атак.

Ряд классических функций и возможностей межсетевого экрана в Cisco IOS еще не поддерживается в ZFW в Cisco IOS версии 12.4(15)T:

- Прокси-сервер аутентификации
- Сбой межсетевого экрана с контролем состояния
- MIB унифицированного межсетевого экрана
- Проверка IPv6 с контролем состояния
- Поддержка некорректного использования TCP

ZFW обычно повышает производительность Cisco IOS для большинства действий по инспекции межсетевого экрана.

Ни Cisco IOS ZFW, ни классический межсетевой экран не включают поддержку проверки с контролем состояния для многоадресного трафика.

Предварительные условия

Требования

Для этого документа отсутствуют особые требования.

Используемые компоненты

Настоящий документ не имеет жесткой привязки к каким-либо конкретным версиям программного обеспечения и оборудования.

Условные обозначения

Дополнительные сведения об условных обозначениях см. в документе Условные обозначения технических терминов Cisco.

Обзор зональной политики

Проверка классического межсетевого экрана с контролем состояния в Cisco IOS, которая ранее называлась контролем доступа на основе содержимого или CBAC, включала использование модели конфигурации на основе интерфейса, предполагающей применение к интерфейсу политики проверки с контролем доступа. Весь трафик, проходящий через этот интерфейс, подлежал проверке с помощью одной и той же политики. Эта модель конфигурации ограничила детализацию политик межсетевого экрана и вызвала дезорганизацию правильного применения политик межсетевого экрана (в частности, в тех сценариях, когда политики межсетевого экрана должны применяться между несколькими интерфейсами).

Межсетевой экран политики на основе зон (также известный как межсетевой экран зональной политики или ZFW) обладает измененной конфигурацией межсетевого экрана: вместо старой модели на основе интерфейса теперь применяется более гибкая и понятная зональная модель. Интерфейсы присваиваются зонам, а политика проверки — трафику, передаваемому между зонами. Межзонные политики отличаются значительной гибкостью и детализацией. Поэтому различные политики проверки можно применять к нескольким группам узлов, связанных с одним и тем же интерфейсом маршрутизатора.

Политики межсетевого экрана настроены с помощью языка политики Cisco® (CPL), в котором для определения проверки сетевых протоколов и групп узлов, подлежащих проверке, применяется иерархическая структура.

Модель конфигурации зональной политики

По сравнению с настройкой классического межсетевого экрана Cisco IOS наличие ZFW полностью меняет способ настройки, выполняемой при проверке межсетевого экрана Cisco IOS.

Первое крупное изменение конфигурации межсетевого экрана связано с появлением зональной конфигурации. Межсетевой экран Cisco IOS — это первая функция защиты от программных угроз в Cisco IOS, в которой реализована модель конфигурации зоны. Со временем зональная модель может распространиться и на другие функции. **Модель конфигурации на базе интерфейса, используемая в ходе проверки классического межсетевого экрана Cisco IOS с контролем состояния (CBAC), содержит набор команд ip inspect и применяется уже некоторое время.** Однако даже если новые функции и настраиваются с помощью классического интерфейса командной строки, их немного. При наличии ZFW проверка с контролем состояния или команды CBAC не используются. Две эти модели конфигурации могут использоваться одновременно в маршрутизаторах, но не сочетаются в интерфейсах. **Интерфейс не может быть настроен как участник зоны безопасности, а также настроен одновременно для оператора ip inspect.**

Зоны устанавливают границы безопасности вашей сети. Зона определяет границу, где трафик, переходящий в другой регион вашей сети, подвержен ограничениям политики. Политика ZFW, выбранная по умолчанию между зонами, состоит в запрете всего трафика. Если ни одна политика не задана явным образом, блокируется весь трафик, перемещающийся между зонами. Здесь наблюдается значительный отход от модели проверки с контролем состояния, где трафик неявно разрешен до тех пор, пока не заблокирован явным образом с помощью списка контроля доступа (ACL).

Второе крупное изменение относится к появлению нового языка политики конфигурации (CPL). Пользователи, знакомые с модульным интерфейсом командной строки для обеспечения качества обслуживания (QoS) в Cisco IOS (MQC), могут заметить, что этот формат похож на то, как применяются карты классов в QoS, когда нужно указать, на какой именно трафик повлияет действие, примененное в карте политик.

Правила для применения межсетевого экрана зональной политики

Участие сетевых интерфейсов маршрутизатора в зонах зависит от ряда правил, управляющих поведением интерфейса, как и трафик, которым обмениваются интерфейсы, входящие в зону:

- Зона должна быть настроена перед тем, как ей можно будет присвоить интерфейсы.
- Интерфейс можно назначить только одной зоне безопасности.
- Весь трафик на входе и на выходе интерфейса неявно блокируется, когда интерфейс присваивается зоне, за исключением трафика, обмен которым выполняется с другими интерфейсами в той же зоне, а также трафика к любому интерфейсу в маршрутизаторе.
- Поток трафика неявно разрешается по умолчанию среди интерфейсов, являющихся участниками одной и той же зоны.
- Чтобы разрешить входящий и исходящий трафик интерфейса, включенного в зону, необходимо настроить политику разрешения или проверки трафика между данной зоной и любой другой зоной.
- Для выбранной по умолчанию политики запрета всего трафика существует лишь одно исключение: собственная зона. Весь трафик к любому интерфейсу маршрутизатора разрешается до тех пор, пока он явным образом не запрещается.
- Трафик не может поступать между интерфейсом, входящим в зону, и любым интерфейсом, который в нее не входит. Такие действия как "пропустить", "проверить" и "отбросить" могут выполняться только между двумя зонами.
- Не присвоенные функции зоны интерфейсы (такие как порты классического маршрутизатора) могут тем не менее использовать классическую проверку с контролем состояния или настройку CBAC.
- Если требуется, чтобы интерфейс устройства не учитывался в политике зонирования/межсетевого экрана, возможно, все равно нужно будет поместить данный интерфейс в зону и настроить политику пропускания всего трафика (что-то вроде фиктивной политики) между этой зоной и любой другой зоной, в которую желательно направить поток трафика.
- Из вышеизложенного следует, что если трафик должен проходить по всем интерфейсам маршрутизатора, все они должны принадлежать модели зонирования (каждый интерфейс должен принадлежать той или иной зоне).

- Единственное исключение из предыдущего подхода, предполагавшего запрет по умолчанию, заключается в том, что входящий и исходящий трафик маршрутизатора разрешен по умолчанию. Явную политику можно настроить для ограничения подобного трафика.

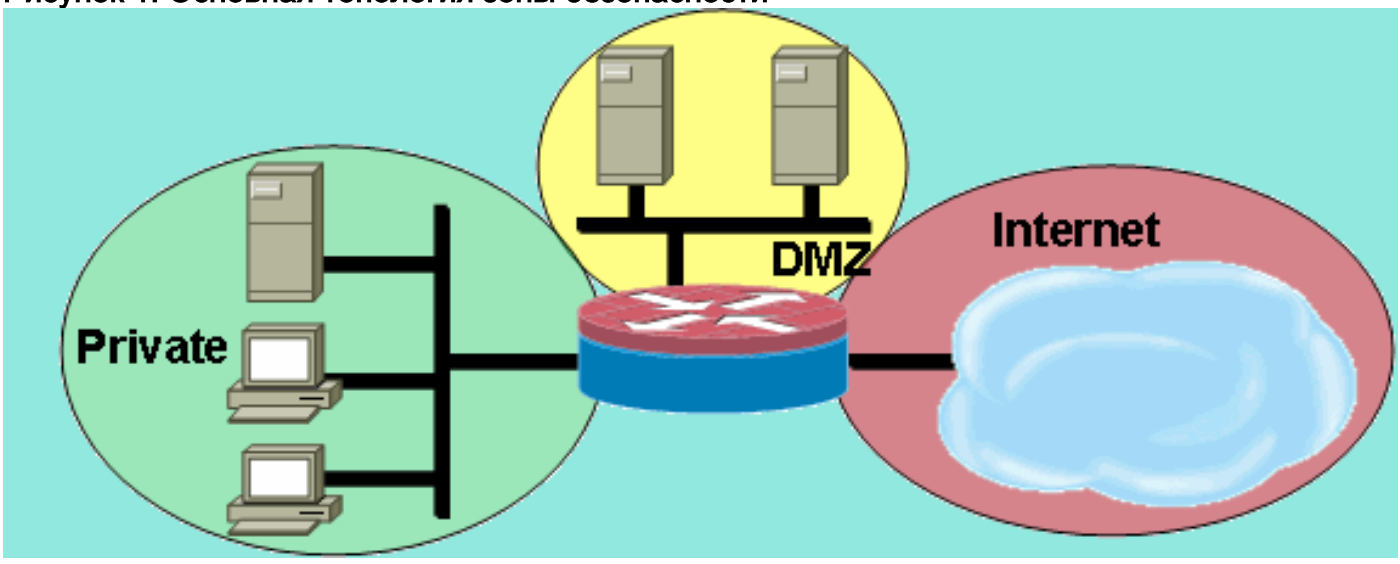
Проектирование безопасности сети для зональной политики

Зону безопасности необходимо настроить для каждого региона относительной безопасности в сети, чтобы все интерфейсы, присвоенные одной и той же зоне, были защищены на одном и том же уровне безопасности. Например, представьте маршрутизатор доступа с тремя интерфейсами:

- Один интерфейс, подключенный к публичному интернету
- Один интерфейс, подключенный к локальной частной сети, который не должен быть доступен из публичного Интернета
- Один интерфейс, подключенный к демилитаризованной зоне (DMZ) интернет-служб, где web-сервер, сервер системы доменных имен (DNS) и почтовый сервер должны быть доступны для публичного интернета

Каждый интерфейс в этой сети будет приписан своей собственной зоне, хотя вам может потребоваться различный доступ из публичного интернета к определенным узлам в DMZ и разные политики применения приложения для узлов в защищенной локальной сети. (см. рис. 1).)

Рисунок 1: Основная топология зоны безопасности



В этом примере каждая зона содержит только один интерфейс. Если в частную зону добавлен еще один интерфейс, узлы, подключенные к новому интерфейсу в зоне, могут пропустить трафик на все узлы существующего интерфейса в той же самой зоне. Кроме того, на трафик узлов к узлам в других зонах точно так же влияют существующие политики.

Как правило, пример сети будет обладать тремя основными политиками:

- Соединение частной зоны с Интернетом
- Соединение частной зоны с узлами DMZ
- Соединение зоны Интернета с узлами DMZ

Поскольку демилитаризованная зона открыта для публичного Интернета, узлы DMZ могут

стать жертвой нежелательных действий злоумышленников, которые способны нанести ущерб одному или нескольким узлам DMZ. Если для узлов DMZ не предусмотрена политика доступа, позволяющая достичь либо узлы частной зоны, либо узлы зоны Интернета, тогда лица, поставившие под угрозу узлы DMZ не могут использовать узлы DMZ для совершения дальнейших атак на узлы частной сети или Интернета. ZFW включает запретительное состояние безопасности по умолчанию. Вследствие этого, если узлам DMZ не предоставить специально доступ к другим сетям, другие сети защищены от любых подключений со стороны узлов DMZ. Аналогично, узлам Интернета не предоставляется доступ к узлам частной зоны, и узлы частной зоны защищены от нежелательного доступа со стороны узлов Интернета.

Использование IPSec VPN с межсетевым экраном зональной политики

Недавние усовершенствования IPSec VPN упрощают настройку политики межсетевого экрана для соединений VPN. Виртуальный туннельный интерфейс IPSec (VTI) и сочетание GRE+IPSec дают возможность ограничить подключения VPN типа "сеть-сеть" и подключения клиентов к определенной зоне безопасности путем размещения туннельных интерфейсов в указанной зоне безопасности. Подключения могут быть изолированы в VPN DMZ, если возможности сетевого взаимодействия должны быть ограничены той или иной политикой. Или же, если соединения VPN неявным образом считаются доверенными, их можно поместить в ту же зону безопасности, что и доверенная внутренняя сеть.

В случает применения интерфейса IPSec, который отличается от VTI, политика межсетевого экрана соединений VPN требует тщательного контроля для поддержания безопасности. В частности, зональная политика должна разрешать доступ по IP-адресу для узлов удаленных сайтов или клиентов VPN, если надежные узлы расположены в зоне, которая отличается от шифрованного подключения клиента VPN к маршрутизатору. Если политика доступа неправильно настроена, узлы, которые следует защитить, могут оказаться открыты для нежелательных и потенциально враждебных узлов. Дальнейшее обсуждение понятий и настройки см. в разделе Использование VPN с межсетевым экраном зональной политики.

Настройка языка политики Cisco (CPL)

Эта процедура может использоваться для настройки ZFW. Последовательность действий неважна, но некоторые события должны быть завершены по порядку. Например, необходимо настроить карту классов перед присвоением ее карте политик. Аналогично, до тех пор, пока не настроена политика, карту политик невозможно присвоить паре зон. При попытке настроить раздел, который зависит от других, еще не настроенных параметров конфигурации, межсетевой экран выдает сообщение об ошибке.

1. Определите зоны.
2. Определите пары зон.
3. Определите карты классов, описывающие трафик, к которому должна применяться политика при пересечении пары зон.
4. Определите карты политик, где указано действие, применяемое в отношении трафика карт классов.
5. Примените карты политик к парам зон.
6. Присвойте интерфейсы зонам.

Настройка карт классов межсетевого экрана зональной политики

Карты классов определяют трафик, выбираемый межсетевым экраном для применения политики. Карты классов уровня 4 сортируют трафик на основании перечисленных ниже критериев. **Эти критерии указываются с помощью команды match в карте классов:**

- Access-group — стандартный, расширенный или именованный список ACL может фильтровать трафик на основе IP-адреса источника и получателя, а также порта источника и получателя.
- Protocol — протоколы уровня 4 (TCP, UDP и ICMP) и службы приложения (например, HTTP, SMTP, DNS и т.д.) Можно указать любую хорошо известную или определенную пользователем службу, учитываемую при сопоставлении портов приложениям.
- Class-map — подчиненная карта классов, предоставляющая дополнительные критерии соответствия, которые вложены в другую карту классов.
- *Not — критерий not указывает на то, что для данной карты классов будет использоваться любой трафик, служба (протокол), группа доступа или подчиненная карта классов которого отличаются от заданных.*

Сочетание критериев соответствия: “Match-Any” и “Match-All”

Чтобы определить порядок применения критериев соответствия в картах классов могут применяться операторы match-any или match-all. Если выбран вариант "match-any", трафик должен соответствовать только одному из критериев соответствия в карте классов. Если выбран вариант "match-all", трафик должен совпадать со всеми критериями карты классов, чтобы принадлежать данному классу.

Сначала должны применяться более конкретные критерии соответствия, а затем — менее конкретные, если трафик отвечает сразу нескольким критериям. Например, рассмотрим следующую карту классов:

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

Трафик HTTP сначала должен быть сравнен с протоколом соответствия http, чтобы убедиться, что трафик обрабатывается специфическими для службы средствами проверки HTTP. Если строки соответствия указаны в обратном порядке, перед сравнением трафика с протоколом соответствия http он проверяется с помощью оператора соответствия протоколу tcp. При этом трафик просто классифицируется как трафик TCP и проверяется, исходя из возможностей компонента межсетевого экрана для проверки TCP. Это представляет проблему для таких служб, как, например, FTP, TFTP, а также для ряда мультимедийных служб и служб голосовой сигнализации (например, H.323, SIP, Skinny, RTSP и т.д.). Эти службы требуют наличия дополнительных возможностей проверки, позволяющих выявить более сложные действия этих служб.

Применение списка ACL в качестве источника критериев соответствия

В картах классов список ACL может применяться в качестве одного из критериев соответствия для применения политики. Если список ACL является единственным критерием соответствия карты классов, и последняя связана с картой политик, применяющей действие проверки, маршрутизатор выполняет основную проверку TCP или UDP для всего трафика, разрешенного списком ACL, за исключением того, для которого в

ZFW предусмотрена проверка с учетом приложения. Во время этой проверки, в том числе, проверяется следующее: FTP, SIP, Skinny (SCCP), H.323, Sun RPC и TFTP. Если доступна проверка с учетом приложения, а список ACL разрешает первичный или контрольный канал, любой второстепенный канал или медиаканал, связанный с основным или контрольным каналом, разрешается независимо от того, разрешен ли трафик в списке ACL.

Если в карте класса в качестве критерия соответствия применен только ACL 101, ACL 101 выглядит следующим образом:

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

Весь трафик разрешен в направлении служебной политики, примененной к заданной паре зон. Соответствующий обратный трафик разрешен в обратном направлении. Следовательно, список ACL должен применить ограничение, лимитировав трафик и оставив только определенные желательные типы трафика. Учтите, что список PAM включает такие службы приложений как HTTP, NetBIOS, H.323 и DNS. Однако несмотря на знание PAM об использовании порта определенным приложением, межсетевой экран только применяет достаточную возможность с учетом приложения, чтобы пойти навстречу хорошо известным требованиям трафика приложения. **Таким образом, простой трафик приложения (например, telnet, SSH и других одноканальных приложений) проверяется как TCP, и статистика комбинируется при выполнении команды show.** Если приложению требуется учитывать сетевую активность, необходимо настроить проверку служб по названию приложения (настроить протокол соответствия http, протокол соответствия telnet, и т.д.).).

Сравните статистику, получаемую при выполнении команды show policy-map type inspect zone-pair в данной конфигурации с более явной политикой межсетевого экрана, которая обсуждается ниже на этой странице. Эта конфигурация применяется для проверки трафика с Cisco IP Phone, а также с нескольких рабочих станций, применяющих разный трафик (в том числе http, ftp, netbios, ssh и dns:):

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

Хотя эту конфигурацию просто определить и учесть весь трафик, возникший в частной зоне (до тех пор пока он направляется через стандартные порты назначения, распознаваемые PAM), она дает ограниченное представление об активности службы и не обеспечивает возможности применения ограничений в отношение полосы пропускания ZFW и сеансов для определенных видов трафика. **Результат выполнения команды show policy-map type inspect zone-pair priv-pub появился из-за предыдущей простой настройки, использующей только**

разрешенные IP-адреса [подсеть] ###для любого списка ACL между парами зон. Как видите, большая часть трафика рабочей станции подсчитано с использованием основной статистики TCP или UDP:

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

В отличие от этой конфигурации похожая конфигурация, добавляющая классы приложения, позволяет получить более подробную статистику о приложении и более полный контроль. При этом она обеспечивает предоставление того же спектра служб, который был продемонстрирован в первом примере, путем определения конечной карты классов, которая соответствует только списку ACL как конечному элементу в карте политик:

```
class-map type inspect match-all all-private
 match access-group 101
class-map type inspect match-all private-ftp
 match protocol ftp
 match access-group 101
class-map type inspect match-any netbios
 match protocol msrpc
 match protocol netbios-dgm
 match protocol netbios-ns
 match protocol netbios-ssn
class-map type inspect match-all private-netbios
 match class-map netbios
 match access-group 101
class-map type inspect match-all private-ssh
 match protocol ssh
 match access-group 101
class-map type inspect match-all private-http
 match protocol http
 match access-group 101
!
policy-map type inspect priv-pub-pmap
 class type inspect private-http
  inspect
 class type inspect private-ftp
  inspect
 class type inspect private-ssh
  inspect
 class type inspect private-netbios
  inspect
 class type inspect all-private
  inspect
 class class-default!
zone security private
zone security public
zone-pair security priv-pub source private destination public
 service-policy type inspect priv-pub-pmap
!
interface FastEthernet4
 ip address 172.16.108.44 255.255.255.0
 zone-member security public
!
interface Vlan1
 ip address 192.168.108.1 255.255.255.0
```

```
 zone-member security private
!
access-list 101 permit ip 192.168.108.0 0.0.0.255 any
```

Уточненная конфигурация дает возможность получить следующие существенные и подробные данные при выполнении команды show policy-map type inspect zone-pair priv-pub:

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

Дополнительное преимущество использования более детальной карты классов и карты политик, как ранее упоминалось, состоит в возможности применения ограничений для класса по отношению к значениям сеанса и скорости, а в особенности к регулировке параметров проверки путем применения карты параметров для коррекции поведения при проверке каждого класса.

Настройка карт политик межсетевого экрана зональной политики

Карта политик применяет действия политики межсетевого экрана к одной или нескольким картам классов, чтобы определить политику службы, которая будет применена к паре зон безопасности. При создании карты политик проверочного типа к концу класса применяется выбранный по умолчанию класс, который называется "class-default". Действие по умолчанию для политики класса class-default — "отбросить", но его можно заменить действием "пропустить". К действию отбрасывания можно добавить параметр журнала. Проверку невозможно применять к классу "class-default".

Действия межсетевого экрана зональной политики

ZFW предусматривает три действия для трафика, переходящего из одной зоны в другую:

- **Отбросить — это действие, выполняемое по отношению ко всему трафику по умолчанию и применяемое классом class-default, указанным в конце каждой карты политик проверочного типа.** Другие карты классов, входящие в данную карту политик, могут быть также настроены на отбрасывание нежелательного трафика. Трафик, обрабатываемый с помощью действия отбрасывания "бесшумно" отбрасывается ZFW (то есть, соответствующий конечный узел не получает уведомление об отбрасывании). Это отличается от поведения при использовании списка ACL, когда узлу, отправившему отклоненный трафик, по протоколу ICMP направляется сообщение “узел недоступен”. На данный момент изменить поведение "бесшумного отбрасывания" невозможно. Отбрасывание может быть дополнено параметром журнала, позволяющим уведомить системный журнал о том, что трафик был отклонен межсетевым экраном.
- **Пропустить — это действие позволяет маршрутизатору переадресовать трафик из одной зоны в другую.** Действие пропускания не сопровождается отслеживанием состояния подключений или сеансов в рамках трафика. Пропускание трафика действует только в одном направлении. Необходимо применить соответствующую политику, чтобы разрешить ответному трафику поступать в противоположном направлении. Действие пропускания полезно для таких протоколов как IPSec ESP, IPSec AH, ISAKMP и других безопасных по своей сути протоколов с предсказуемым поведением. Однако большая часть трафика приложения лучше обрабатывается в ZFW с помощью действия проверки.
- **Проверка — действие проверки позволяет контролировать трафик, исходя из состояния.** Например, если проверяется трафик, поступающий в предыдущем примере из частной зоны в зону Интернета, маршрутизатор сохраняет информацию о подключении или сеансе для трафика, использующего протокол TCP или UDP. Следовательно, маршрутизатор разрешает обратный трафик, отправляемый с узлов зоны Интернета в ответ на запросы соединения из частной зоны. Кроме того, проверка может предполагать проверку приложения и управление определенными протоколами службы, используемыми при передаче уязвимого или особо важного трафика приложения. Можно применить след аудита с картой параметров, чтобы записать время начала подключения или сеанса, время его окончания, продолжительность, перенесенный объем данных, а также адреса источника и назначения.

Действия связаны с картами классов в картах политик:

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

Карты параметров дают возможность изменять параметры подключения для заданной политики проверки в карте классов.

Настройка карт параметров межсетевого экрана зональной политики

В картах параметров указывается поведение проверки для ZFW, относящееся к таким параметрам, как защита от DoS-атак, таймеры подключений TCP или сеансов UDP, а также настройки ведения журнала при создании следа аудита. Карты параметров также применяются к классу уровня 7 и картам политик для определения поведения приложения (например, объекты HTTP, требования проверки подлинности POP3 и IMAP и другая информация, относящаяся к приложению).

Карты параметров проверки для ZFW настроены как type inspect подобно другим объектам классов и политик ZFW:

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

В определенных типах карт параметров указаны параметры, применяемые согласно политикам проверки приложения на уровне 7. Карты параметров типа regex определяют регулярное выражение, используемое вместе с проверкой приложения HTTP, фильтрующей трафик с помощью регулярного выражения:

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

В картах параметров типа protocol-info определяются названия серверов, используемые при проверке приложения для обмена мгновенными сообщениями:

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

Полностью сведения о конфигурации проверки HTTP и приложения для обмена мгновенными сообщениями приведены в соответствующих разделах этого документа, посвященных проверке приложений.

Настройка защиты от DoS-атак рассматривается в одном из последующих разделов документа.

Настройка проверки приложения рассматривается в одном из приведенных ниже разделов.

Применение ведения журнала для политик межсетевого экрана зональной политики

ZFW позволяет вести журнал для трафика, отброшенного или проверенного по умолчанию, а также согласно настроенным действиям политики межсетевого экрана. Запись в журнал следа аудита доступно для проверяемого ZFW трафика. След аудита применяется путем определения следа аудита в карте параметров и применения карты параметров с действием проверки в карте политик:

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

Запись в журнал доступна для отбрасываемого ZFW трафика. Она настраивается путем добавления в карту политик журнала с действием отбрасывания:

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

Изменение карт классов и карт политик межсетевого экрана зональной политики

Сейчас в ZFW нет редактора, позволяющего изменять различные компоненты ZFW (например, карты политик, карты классов и карты параметров). Чтобы перенести операторы

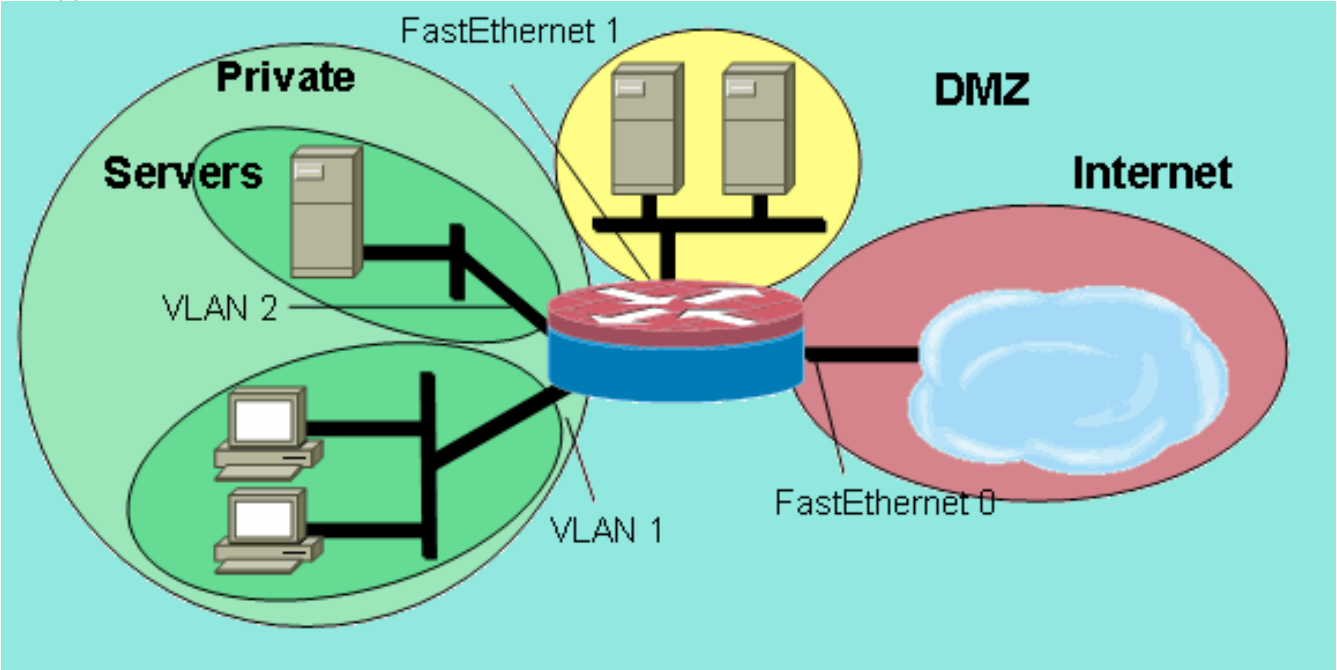
соответствия из карты классов или выполняющего действие приложения в различные карты классов, содержащиеся в карте политик, необходимо выполнить следующие действия:

1. Скопируйте существующий компонент в текстовый редактор (например, в блокнот Microsoft Windows или в редактор vi на платформе Linux/Unix).
2. Удалите существующий компонент из конфигурации маршрутизатора.
3. Измените компонент в своем текстовом редакторе.
4. Скопируйте его обратно в интерфейс командной строки маршрутизатора.

Примеры конфигураций

В этом примере конфигурации используется маршрутизатор Cisco 1811 Integrated Services Router. Базовая конфигурация с возможностью IP-подключения, конфигурацией VLAN и прозрачным мостовым соединением между двумя частными сегментами локальной сети на базе Ethernet приведена в Приложении A. Маршрутизатор делится на пять зон:

- Публичный Интернет подключен к FastEthernet 0 (зона Интернета)
- Два web-сервера подключены к FastEthernet 1 (зона DMZ)
- Коммутатор Ethernet настроен с помощью двух интерфейсов виртуальной локальной сети VLAN:Рабочие станции подключены к VLAN1 (клиентская зона).Серверы подключены к VLAN2 (серверная зона).И клиентская, и серверная зоны принадлежат одной и той же подсети. Между зонами будет располагаться прозрачный межсетевой экран, и межзонные политики этих интерфейсов будут влиять только на трафик между клиентской и серверной зонами.
- Интерфейсы VLAN1 и VLAN2 связываются с другими сетями с помощью виртуального интерфейса моста (BVI1). Этот интерфейс присвоен частной зоне. (см. рис. 2).)**Рис. 2: Сведения о топологии зоны**

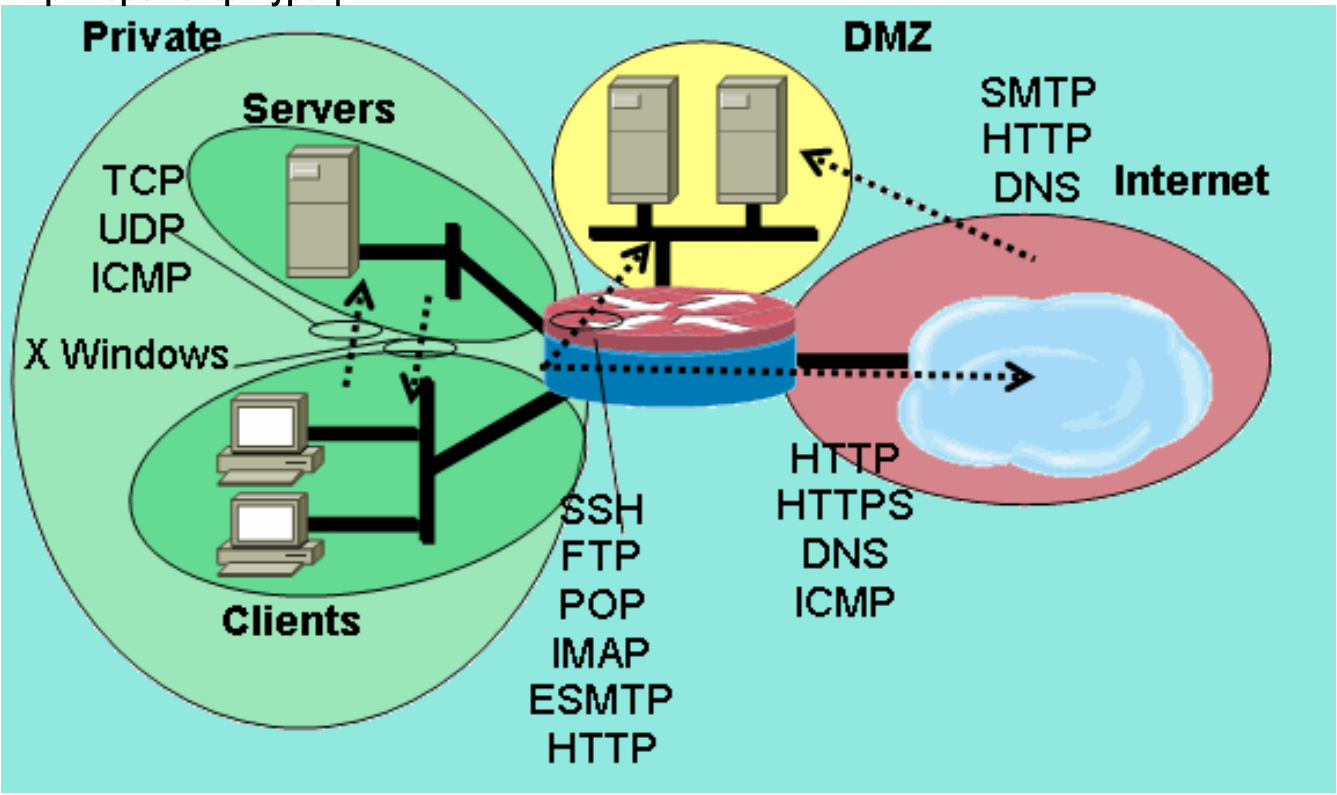


Эти политики применяются с помощью ранее определенных сетевых зон:

- Узлы в зоне Интернета могут иметь доступ к службам DNS, SMTP и SSH на одном сервере в DMZ. Другой сервер будет предлагать службы SMTP, HTTP и HTTPS. Политика межсетевого экрана ограничит доступ к определенным службам, которые

доступны на каждом узле.

- Узлы DMZ не могут подключаться к узлам в любой другой зоне.
- Узлы в клиентской зоне могут подключаться к узлам в серверной зоне во всех службах TCP, UDP и ICMP.
- Узлы в серверной зоне не могут подключаться к узлам в клиентской зоне кроме серверам приложений на базе UNIX, который способен открывать клиентские сеансы X Windows для серверов X Windows на настольных компьютерах в клиентской зоне (порты с 6900 по 6910).
- Все узлы в частной зоне (сочетание клиентов и серверов) могут обращаться к узлам в DMZ по поводу служб SSH, FTP, POP, IMAP, ESMTP и HTTP. Что касается зоны Интернета, в ней доступны службы HTTP, HTTPS, DNS и ICMP. Более того, проверка приложения будет применена к подключениям HTTP из частной зоны к зоне Интернета, чтобы поддерживаемые приложения для обмена мгновенными сообщениями и P2P не использовали порт 80 (см. рис. 3.).)**Рис. 3: Разрешения службы пары зон, применяемые в примере конфигурации**



Эти политики межсетевого экрана настраиваются в порядке сложности:

1. Проверка клиентов и серверов (TCP, UDP и ICMP)
2. Проверка частной зоны DMZ (SSH, FTP, POP, IMAP, ESMTP и HTTP)
3. Проверка зоны Интернета и зоны DMZ (SMTP, HTTP и DNS), ограниченная адресом узла
4. Проверка серверов и клиентов X Windows со службой, заданной при сопоставлении портов приложениям (PAM)
5. Частный Интернет (HTTP/HTTPS/DNS/ICMP) с проверкой приложения HTTP

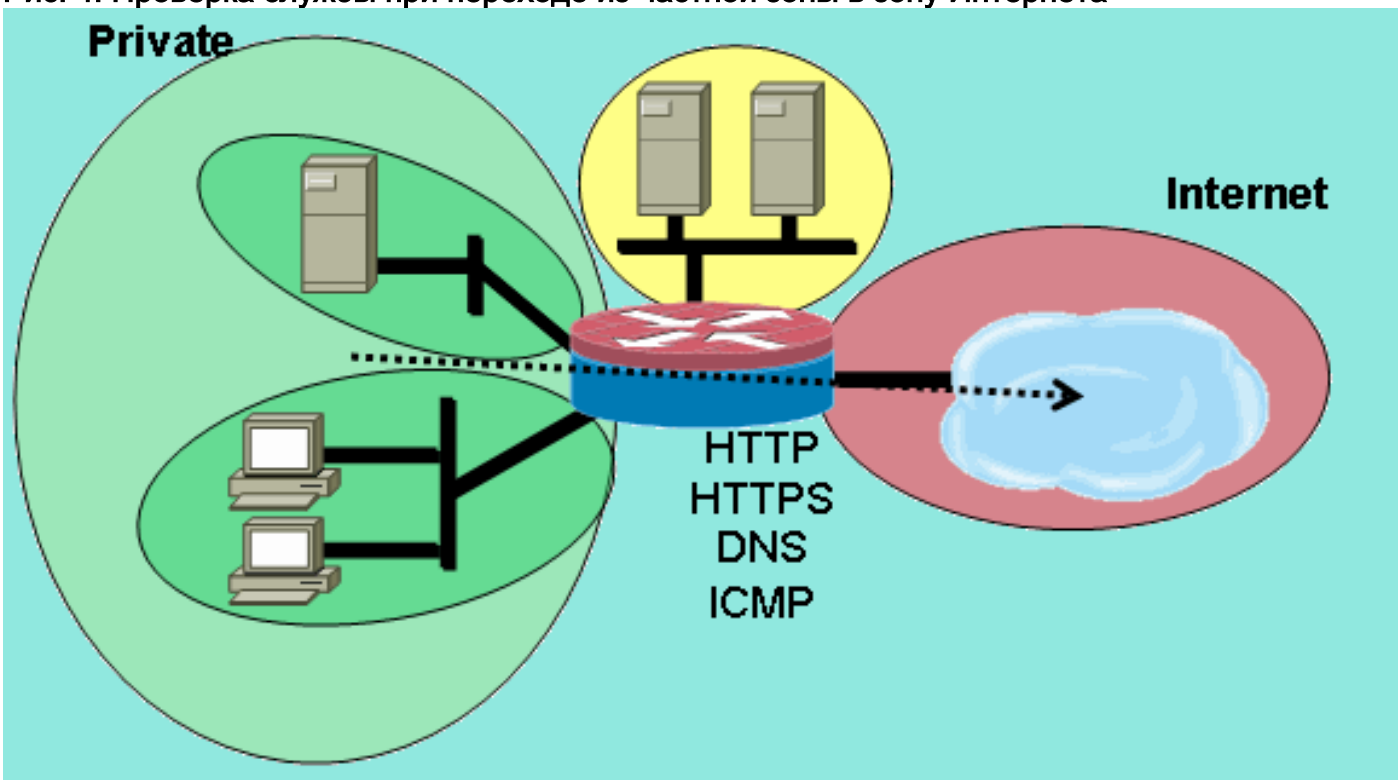
Поскольку вы будете применять компоненты конфигурации к различным сетевым сегментам в разное время, важно помнить, что сетевой сегмент потеряет возможность подключения к другим сегментам при помещении в зону. Например, когда частная зона настроена, узлы в частной зоне потеряют возможность подключения к зонам DMZ и Интернета вплоть до определения соответствующих политик.

Межсетевой экран маршрутизации проверки с отслеживанием состояния соединения

Настройка политики в отношении частной зоны и зоны Интернета

На рис. 4 приведена конфигурация политики частного Интернета.

Рис. 4: Проверка службы при переходе из частной зоны в зону Интернета



Политика частной зоны и Интернета применяет проверку уровня 4 к проверке HTTP, HTTPS, DNS и проверку уровня 4 для ICMP при переходе из частной зоны в зону Интернета. Она разрешает устанавливать подключения из частной зоны в зону Интернета и обратный трафик. Проверка на уровне 7 имеет такие преимущества как более тесное управление приложениями, повышенная безопасность и поддержка приложений, требующих привязки. Однако проверка на уровне 7, как упоминалось ранее, требует лучшего понимания сетевой активности, поскольку между зонами будут запрещены протоколы уровня 7, для которых не настроена проверка.

1. Определите карты классов, описывающие трафик, который требуется разрешить между зонами согласно ранее описанным политикам:`conf t`
```
 class-map type inspect match-any internet-traffic-class
  match protocol http
  match protocol https
  match protocol dns
  match protocol icmp
```
2. Настройте карту политик для проверки трафика в только что определенных картах классов:`conf t`
```
 policy-map type inspect private-internet-policy
  class type inspect internet-traffic-class
   inspect
```
3. Настройте частную зону и зону Интернета и присвойте соответствующим зонам интерфейсы маршрутизатора:`conf t`
```
zone security private
```

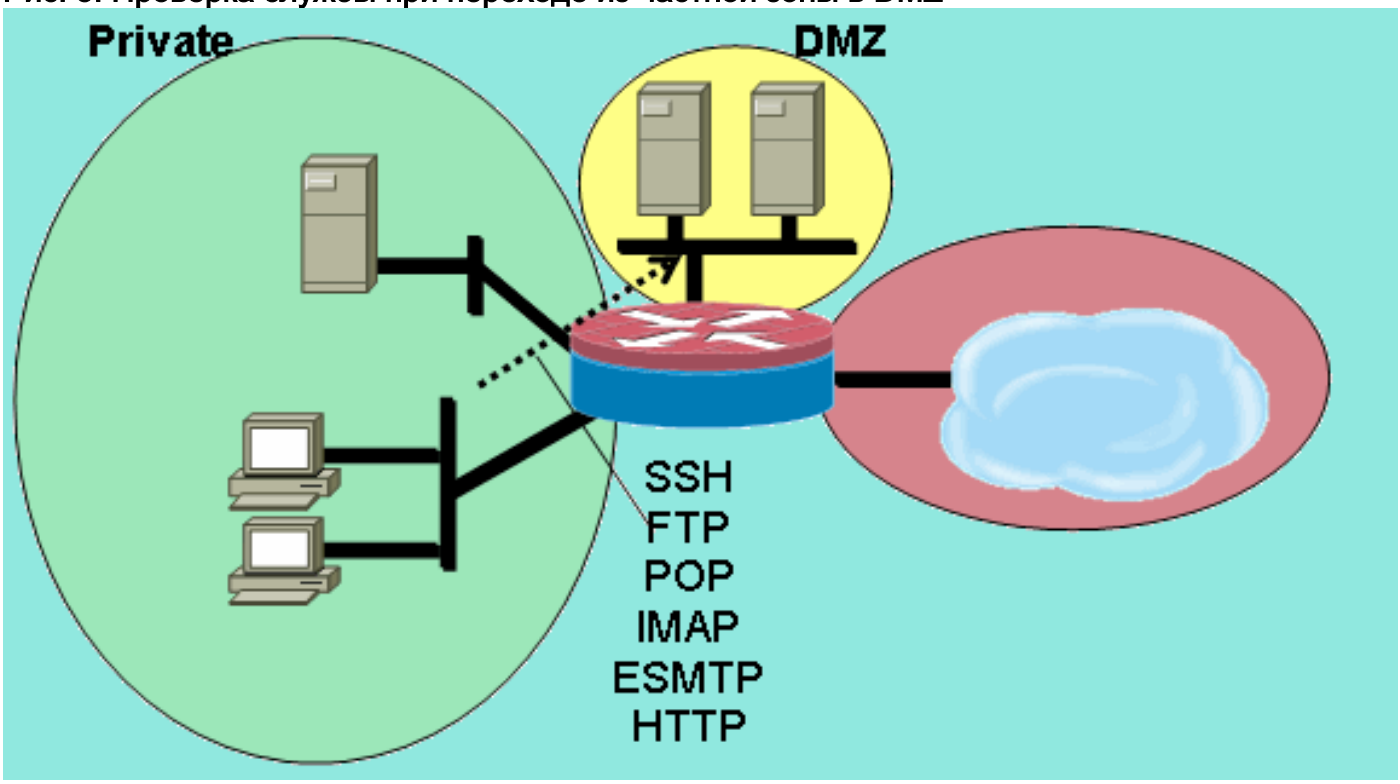
```
zone security internet
int bvi1
zone-member security private
int fastethernet 0
zone-member security internet
```

4. Настройте пару зон и примените соответствующую карту политик.**Примечание:** Только необходимо настроить пару зоны закрытого Интернета в настоящее время для осмотра соединений, полученных в частной зоне, перемещающейся в интернет-зону:`conf t`
 `zone-pair security private-internet source private destination internet`
 `service-policy type inspect private-internet-policy`На этом настройка политики проверки на уровне 7 в паре "частная зона-зона Интернета", разрешающая подключения по протоколам HTTP, HTTPS, DNS и ICMP из клиентской зоны в серверную зону и применение проверки приложений к трафику HTTP с целью убедиться, что нежелательному трафику запрещено проходить через порт TCP 80, относящийся к службе HTTP.

Настройка политики в отношении частной зоны и DMZ

На рис. 5 приведена конфигурация политики, относящейся к частной зоне и зоне Интернета.

Рис. 5: Проверка службы при переходе из частной зоны в DMZ



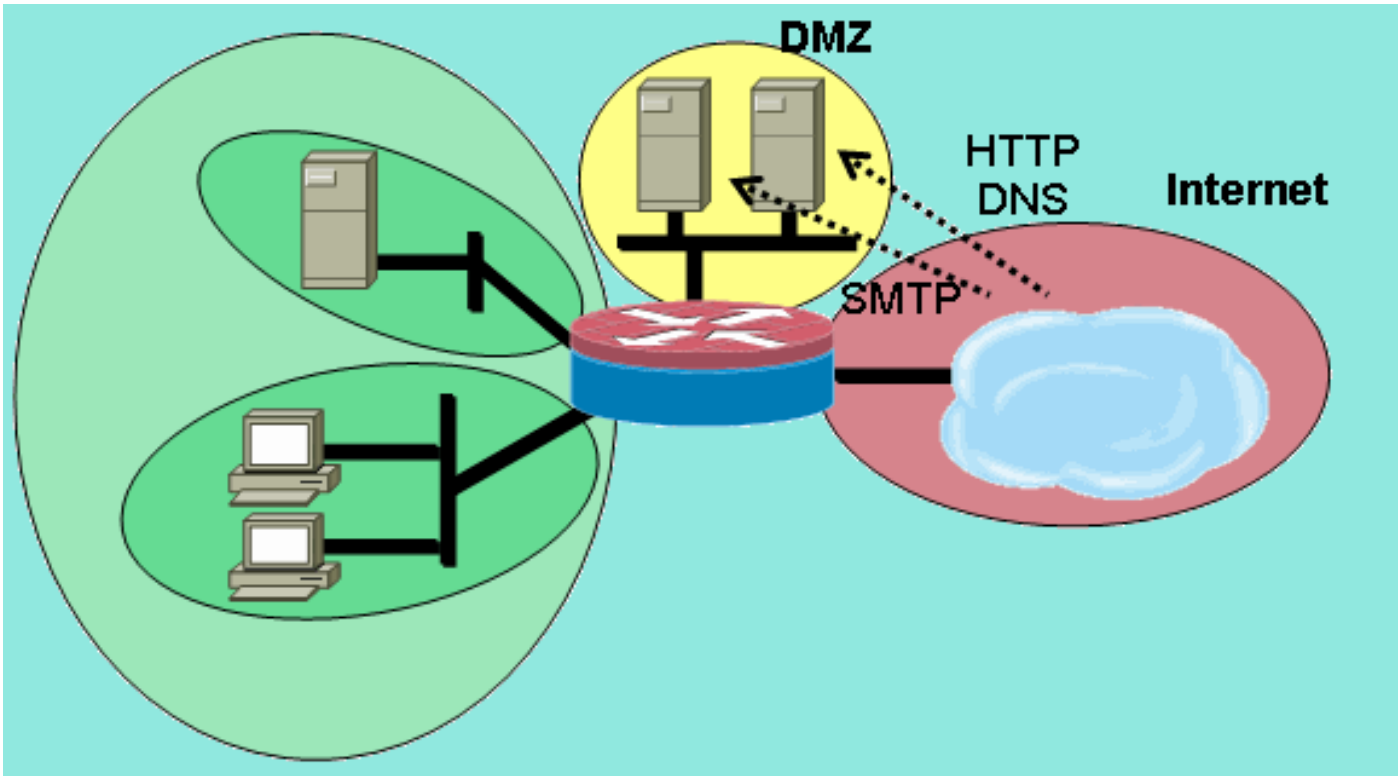
Политика проверки частной зоны и DMZ осложняет ситуацию, поскольку она требует более четкого понимания маршрутов сетевого трафика между зонами. Эта политика применяет проверку на уровне 7 при переходе из частной зоны в DMZ. Она разрешает устанавливать подключения из частной зоны в DMZ и обратный трафик. Проверка на уровне 7 имеет такие преимущества как более тесное управление приложениями, повышенная безопасность и поддержка приложений, требующих привязки. Однако проверка на уровне 7, как упоминалось ранее, требует лучшего понимания сетевой активности, поскольку между зонами будут запрещены протоколы уровня 7, для которых не настроена проверка.

1. Определите карты классов, описывающие трафик, который требуется разрешить между зонами согласно ранее описанным политикам:`conf t`
```
class-map type inspect match-any L7-inspect-class
 match protocol ssh
 match protocol ftp
 match protocol pop
 match protocol imap
 match protocol esmtp
 match protocol http
```
2. Настройте карты политик для проверки трафика в только что определенных картах классов:`conf t`
```
policy-map type inspect private-dmz-policy
 class type inspect L7-inspect-class
  inspect
```
3. Настройте частную зону и DMZ, а затем присвойте соответствующим зонам интерфейсы маршрутизатора:`conf t`
```
zone security private
zone security dmz
int bvi1
zone-member security private
int fastethernet 1
zone-member security dmz
```
4. Настройте пару зон и примените соответствующую карту политик.**Примечание:** Только необходимо настроить частный DMZ, зонально-парный в настоящее время для осмотра соединений, полученных в частной зоне, перемещающейся в DMZ:`conf t`
```
zone-pair security private-dmz source private destination dmz
 service-policy type inspect private-dmz-policy
```
На этом настройка политики проверки DMZ на уровне 7, разрешающей все подключения по протоколу TCP, UDP и ICMP из клиентской зоны в серверную зону, завершена. В этой политике не применяется привязка для подчиненных каналов. Она лишь служит примером простой политики, подходящей для большинства подключений приложений.

Настройка политики в отношении зоны Интернета и DMZ

На рис. 6 приведена конфигурация политики, относящейся к зоне Интернета и DMZ.

Рис. 6: Проверка службы при переходе из зоны Интернета в DMZ



Эта политика применяет проверку на уровне 7 при переходе из зоны Интернета в DMZ. Она разрешает подключения из зоны Интернета к DMZ, а также обратный трафик от узлов DMZ к узлам Интернета, инициировавшим подключение. Политика в отношении зоны Интернета и DMZ сочетает проверку на уровне 7 с группами адресов, определенными списками ACL в целях ограничения доступа к определенным службам на определенных узлах, в группах узлов или подсетях. Это достигается путем вложения карты классов, указывающей на службы, в другую карту классов, ссылающуюся на список ACL для указания IP-адресов.

1. Определите карты классов и списки ACL, описывающие трафик, который требуется разрешить между зонами согласно ранее описанным политикам.Необходимо использовать несколько карт классов для служб, поскольку при доступе к двум различным серверам будут применяться разные политики. Интернет-узлам разрешены подключения по протоколам DNS и HTTP по адресу 172.16.2.2, а также SMTP-подключения по адресу 172.16.2.3. Обратите внимание на разницу в картах классов. **Ключевое слово match-any применяется в картах классов, указывающих на услуги, с целью разрешить любые из перечисленных служб. Ключевое слово match-all применяется в картах классов, связывающих списки ACL с картами классов служб. Оно означает, что разрешение трафика требует соблюдения обоих условий в карте классов:**
```
conf t
 access-list 110 permit ip any host 172.16.2.2
 access-list 111 permit ip any host 172.16.2.3
 class-map type inspect match-any dns-http-class
  match protocol dns
  match protocol http
 class-map type inspect match-any smtp-class
  match protocol smtp
 class-map type inspect match-all dns-http-acl-class
  match access-group 110
  match class-map dns-http-class
 class-map type inspect match-all smtp-acl-class
  match access-group 111
  match class-map smtp-class
```
2. Настройте карты политик для проверки трафика в только что определенных картах

классов:
```
conf t
 policy-map type inspect internet-dmz-policy
  class type inspect dns-http-acl-class
   inspect
  class type inspect smtp-acl-class
   inspect
```

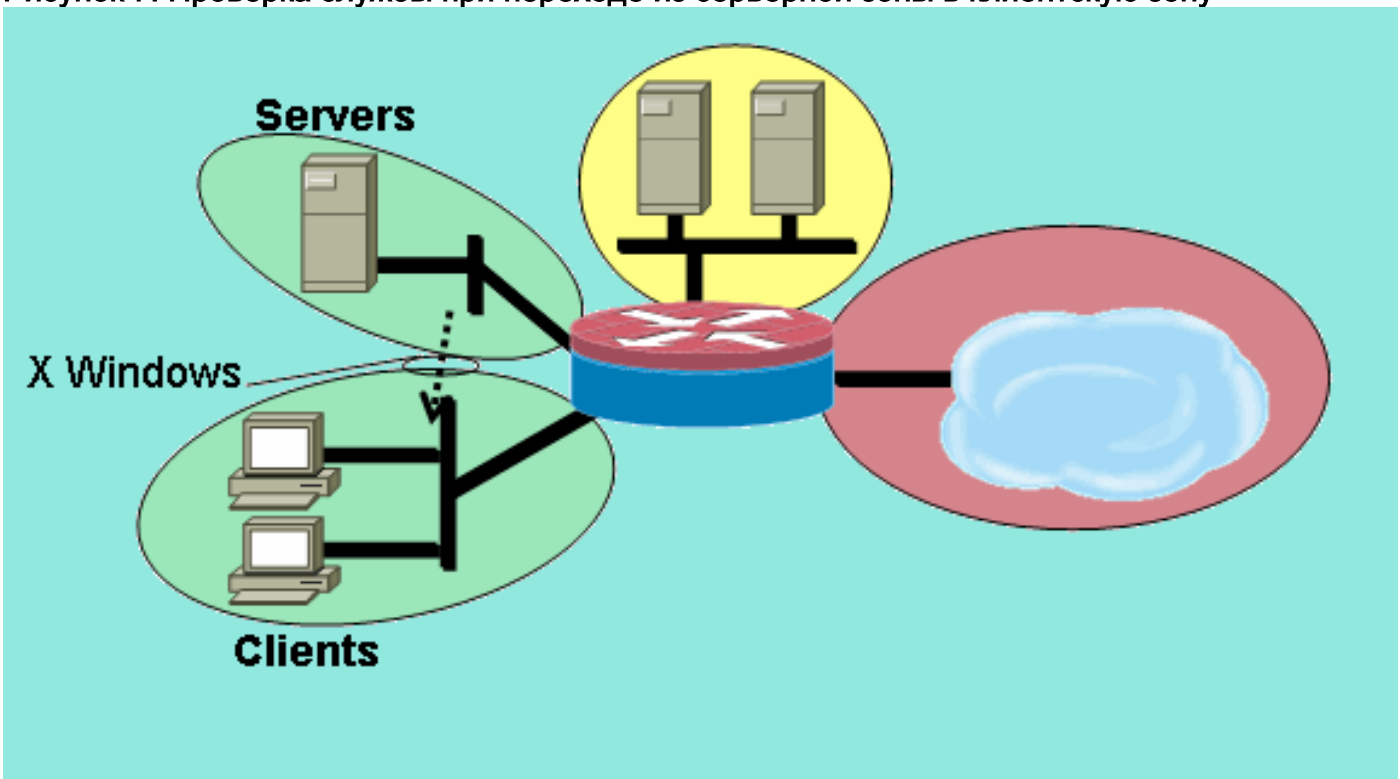
3. Настройте частную зону и DMZ, а затем присвойте соответствующим зонам интерфейсы маршрутизатора. Пропустите настройку DMZ, если она уже настроена при ознакомлении с предыдущим разделом:
```
conf t
zone security internet
zone security dmz
int fastethernet 0
 zone-member security internet
int fastethernet 1
 zone-member security dmz
```
4. Настройте пару зон и примените соответствующую карту политик.**Примечание:** Только необходимо настроить интернет-пару зоны DMZ в настоящее время, для осмотра соединений, полученных в интернет-зоне, перемещающейся в зону DMZ:
```
conf t
 zone-pair security internet-dmz source internet destination dmz
  service-policy type inspect internet-dmz-policy
```
На этом настройка политики проверки по адресу на уровне 7 для пары "зона Интернета-DMZ" завершена.

Прозрачный межсетевой экран проверки с отслеживанием состояния соединения

Настройка политики "серверы-клиенты"

На рис. 7 приведена конфигурация политики сервер-клиент.

Рисунок 7: Проверка службы при переходе из серверной зоны в клиентскую зону



Политика в отношение серверной и клиентской зоны предполагает применение проверки с помощью службы, определенной пользователем. Проверка на уровне 7 применяется при

переходе из серверной зоны в клиентскую зону. Она позволяет устанавливать подключения X Windows к определенному диапазону портов из серверной зоны в клиентскую зону и разрешает обратный трафик. X Windows не является исходно поддерживаемым протоколом в PAM. Поэтому в PAM необходимо определить настраиваемую пользователем службу, чтобы дать возможность ZFW распознать и проверить соответствующий трафик.

В группе мостов IEEE настраиваются два и более интерфейса маршрутизатора, позволяющие выполнять интегрированную маршрутизацию и соединение с помощью мостов (IRB). Это дает возможность создавать мосты между интерфейсами в группе мостов и организовывать маршрутизацию в другие подсети с помощью виртуального интерфейса моста (BVI). Политика прозрачного межсетевого экрана предложит применить проверку межсетевого экрана к “пересекающему мост” трафику, но не к трафику, который покидает группу моста через BVI. Политика проверки применяется только к трафику, пересекающему эту группу моста. Следовательно, в этом сценарии проверка будет применяться только к трафику, перемещающемуся между клиентской и серверной зонами, которые вложены в частную зону. Политика, применяемая между частной зоной, публичной зоной и DMZ, вступает в силу лишь тогда, когда трафик покидает группу моста через BVI. Когда трафик поступает сквозь BVI из клиентской или серверной зоны, политика прозрачного межсетевого экрана не вызывается.

1. Настройте PAM с помощью определяемой пользователем записи для X Windows.Клиенты X Windows (где расположены приложения) открывают подключения для предъявления информации клиентам (где работает пользователь) в диапазоне, начинающемся с порта 6900.Каждое дополнительное подключение использует последовательные порты. Если указано, что клиент имеет 10 различных сеансов на одном узле, сервер использует порты 6900-6909. Поэтому при проверке диапазона портов с 6900 по 6909 подключения, которые открываются для портов с номером, превышающим 6909, не удастся установить:`conf t`
```
 ip port-map user-Xwindows port tcp from 6900 to 6910
```
2. Просмотрите документы PAM, чтобы найти ответы на другие вопросы в связи с PAM. Или обратитесь к подробной документации по проверке, где приведены сведения о взаимодействии между PAM и проверкой межсетевого экрана с контролем состояния соединения в Cisco IOS.
3. Определите карты классов, описывающие трафик, который требуется разрешить между зонами согласно ранее описанным политикам:`conf t`
```
 class-map type inspect match-any Xwindows-class
  match protocol user-Xwindows
```
4. Настройте карты политик для проверки трафика в только что определенных картах классов:`conf t`
```
 policy-map type inspect servers-clients-policy
  class type inspect Xwindows-class
   inspect
```
5. Настройте клиентскую и серверную зоны, а затем присвойте соответствующим зонам интерфейсы маршрутизатора.Если настройка этих зон и назначение интерфейсов уже выполнены во время ознакомления с разделом "Конфигурация политики клиентских и серверных зон", можно перейти к определению пары зон. Для полноты картины предоставляется конфигурация создания моста IRB:`conf t`
```
bridge irb
bridge 1 protocol ieee
bridge 1 route ip
zone security clients
zone security servers
 int vlan 1
```

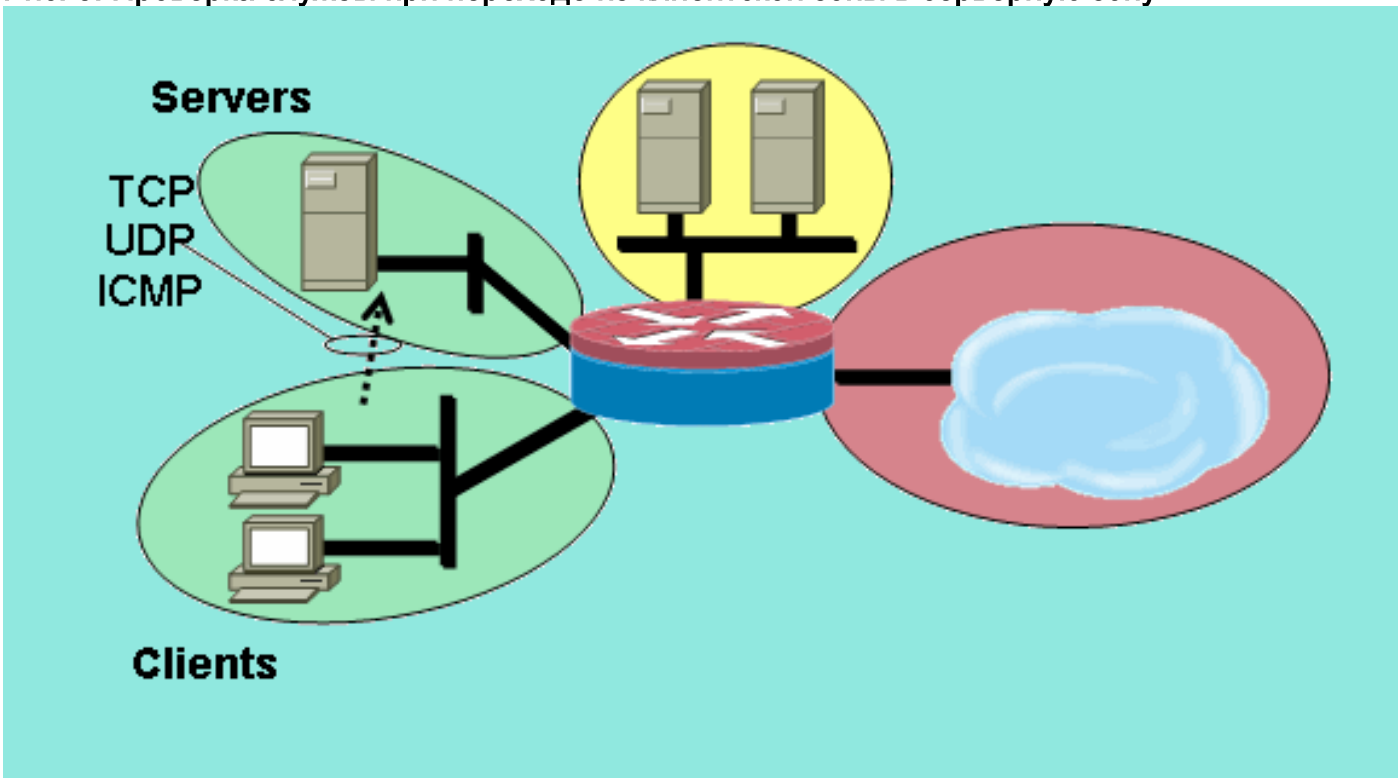
```
 bridge-group 1
 zone-member security clients
int vlan 2
 bridge-group 1
 zone-member security servers
```

6. Настройте пару зон и примените соответствующую карту политик.**Примечание:** Только необходимо настроить пару зоны клиентов серверов в настоящее время для осмотра соединений, полученных в зоне серверов, перемещающейся в зону клиентов:`conf t`
 `zone-pair security servers-clients source servers destination clients`
 `service-policy type inspect servers-clients-policy`Настройка политики проверки, определяемой пользователем, в паре "серверная зона-клиентская зона", разрешающая подключения X Windows из серверной зоны к клиентской зоне, на этом завершается.

Настройка политики "клиенты-серверы"

На рис. 8 приведена конфигурация политика "клиент-сервер".

Рис. 8: Проверка службы при переходе из клиентской зоны в серверную зону



Политика "клиент-серверы" менее сложна, чем другие. Проверка на уровне 4 применяется при переходе из клиентской зоны в серверную зону. Она разрешает устанавливать подключения из клиентской зоны в серверную зону и обратный трафик. Проверка на уровне 4 отличается простотой конфигурации межсетевого экрана. Преимущество состоит в том, что для разрешения большей части трафика приложений требуется лишь несколько правил. Однако проверка на уровне 4 сопряжена с двумя крупными недостатками:

- Такие приложения как FTP или службы потоковой передачи мультимедиа часто согласуют открытие дополнительного подчиненного канала от сервера к клиенту. Эта функциональная возможность обычно реализуется за счет привязки службы, следящей за диалогом по контрольному каналу и разрешающей наличие подчиненного канала. Эта возможность недоступна при проведении проверки на уровне 4.
- Проверка на уровне 4 разрешает почти весь трафик уровня приложений. Если

использование сети должно контролироваться так, чтобы через межсетевой экран допускались только несколько приложений, необходимо настроить список ACL для исходящего трафика, чтобы ограничить круг служб, которым разрешено проникновение сквозь межсетевой экран .

Оба интерфейса маршрутизатора настроены в виде группы моста IEEE. Поэтому данная политика межсетевого экрана приведет к применению проверки прозрачного межсетевого экрана. Эта политика применяется по отношению к двум интерфейсам в группе моста IEEE IP. Политика проверки применяется только к трафику, пересекающему эту группу моста. Это объясняет причину того, что клиентская и серверная зоны вложены в частную зону.

1. Определите карты классов, описывающие трафик, который требуется разрешить между зонами согласно ранее описанным политикам:`conf t`
```
class-map type inspect match-any L4-inspect-class
match protocol tcp
match protocol udp
match protocol icmp
```
2. Настройте карты политик для проверки трафика в только что определенных картах классов:`conf t`
```
policy-map type inspect clients-servers-policy
class type inspect L4-inspect-class
 inspect
```
3. Настройте клиентскую и серверную зоны и присвойте соответствующим зонам интерфейсы маршрутизатора:`conf t`
```
zone security clients
zone security servers
int vlan 1
zone-member security clients
int vlan 2
zone-member security servers
```
4. Настройте пару зон и примените соответствующую карту политик.**Примечание:** Только необходимо настроить серверы клиентов, зонально-парные в настоящее время, для осмотра соединений, полученных в зоне клиентов, перемещающейся в зону серверов:`conf t`
```
zone-pair security clients-servers source clients destination servers
 service-policy type inspect clients-servers-policy
```
На этом настройка политики проверки DMZ на уровне 4, разрешающей все подключения по протоколу TCP, UDP и ICMP из клиентской зоны в серверную зону, завершена. В этой политике не применяется привязка для подчиненных каналов. Она лишь служит примером простой политики, подходящей для большинства подключений приложений.

Регулировка скорости для межсетевого экрана зональной политики

Сети передачи данных часто выигрывают от наличия возможности ограничить скорость передачи определенных типов сетевого трафика, а также ограничить влияние низкоприоритетного трафика на более важный для бизнеса трафик. В программном обеспечении Cisco IOS эта возможность реализована с помощью политик ограничения скорости трафика, которые ограничивают номинальную скорость и размер пакета. Поддержка политик ограничения скорости трафика обеспечивается, начиная с версии 12.1(5)T Cisco IOS.

Cisco IOS версии 12.4(9)T расширяет возможности ZFW за счет поддержки ограничения скорости, добавляя функцию регулировки трафика, совпадающего с определениями из конкретной карты классов и пересекающего межсетевой экран при переходе из одной зоны

безопасности в другую. Удобство состоит в том, что предлагается описывать трафик, применять политику межсетевого экрана и регулировать расход полосы пропускания для данного трафика с помощью одной точки конфигурации. Контроль с помощью ZFW отличается от регулировки на базе интерфейса тем, что в ходе него только предоставляются действия (передача в случае соответствия политике и отбрасывание при нарушении политики). Такой контроль не позволяет маркировать трафик для DSCP.

С помощью ZFW можно только указать использование полосы пропускания в байтах за секунду и пакетах за секунду. Выбрать процент использования полосы пропускания не удастся. Регулировка посредством ZWF может применяться как наряду с регулировкой на основе интерфейса, так и по отдельности. Поэтому если необходимы дополнительные возможности регулировки, эти функции можно применять при регулировке на базе интерфейса. Если регулировка на основе интерфейса используется в сочетании с регулировкой с помощью межсетевого экрана, убедитесь в том, что соответствующие политики не конфликтуют.

Настройка регулировки с помощью ZFW

Регулировка с помощью ZFW ограничивает трафик в карте классов карты политик до значения скорости, определяемого пользователем, которое находится в пределах от 8 000 до 2 000 000 000 бит/с, и присваивает настраиваемое значение размера пакета в диапазоне от 1 000 до 512 000 000 байт.

Настройка регулировки осуществляется дополнительной строкой конфигурации в карте политик, которая применяется после действия политики:

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

Управление сеансами

Во время регулировки с помощью ZFW выполняется также управление сеансами, позволяющее ограничивать трафик сеанса в карте политик, соответствующей карте классов. Это — дополнение к существующей возможности применять защиту от DoS-атак для отдельной карты классов. По сути это позволяет более точно управлять количеством сеансов, совпадающих с любой заданной картой классов, пересекающей пару зон. Если одна и та же карта классов используется в нескольких картах классов или парах зон, к различным картам классов можно применять разные ограничения сеанса.

Управление сеансами применяется путем настройки карты параметров, содержащей объем данных сеанса, и последующего присоединения карты параметров к действию проверки, применяемому к карте классов в соответствие с картой политик:

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

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

Карты параметров могут применяться только к действию проверки; они не доступны при выполнении таких действий как пропуск или отбрасывание.

Управление сеансами ZFW и действия регулирования видны при применении команды

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

Контроль трафика на прикладном уровне

Проверка приложения — это дополнительная возможность ZFW. Политики проверки приложений применяются на уровне 7 модели OSI, где пользовательские приложения отправляют и получают сообщения, позволяющие приложениям предлагать полезные возможности. Ряд приложений может предлагать нежелательные или уязвимые возможности. Поэтому связанные с этими возможностями сообщения должны фильтроваться, чтобы ограничить действия служб приложений.

ZFW в Cisco IOS позволяет проводить проверку приложений и управлять следующими службами приложений:

- HTTP
- SMTP
- POP3
- IMAP
- Sun RPC
- Трафик приложения P2P
- Приложения для обмена мгновенными сообщениями

Возможности проверки приложения и управления (AIC) различны в разных службах. Проверка HTTP обеспечивает точную фильтрацию нескольких типов активности приложений, давая возможность ограничить объем переноса, длину web-адреса и активность браузера, чтобы обеспечить соответствие со стандартами поведения приложений и ограничить типы содержания, переносимого с помощью данной службы. С помощью AIC для SMTP можно ограничить длину содержимого и обеспечить соответствие протоколу. Проверка POP3 и IMAP помогает убедиться, что пользователи прибегают к надежным механизмам аутентификации, которые не ставят под удар учетные данные пользователей.

Проверка приложения настроена как дополнительный набор карт классов и карт политик для конкретного приложения, которые затем применяются к существующим картам классов и картам политик проверки путем определения политики службы приложений в карте политик проверки.

Проверка приложения HTTP

Проверка приложения может быть применена к трафику HTTP, чтобы проконтролировать нежелательное использование порта службы HTTP для других приложений (например, программ для обмена мгновенными сообщениями, приложений для однорангового общего доступа к файлам и приложений для туннелирования, которые могут перенаправлять данные приложений через порт TCP 80, которые в противном случае были бы блокированы межсетевым экраном).

Настройте карту классов для проверки приложений, чтобы описать трафик, который не является разрешенным трафиком HTTP:

```
! configure the actions that are not permitted
```

```
class-map type inspect http match-any http-aic-cmap
 match request port-misuse any
 match req-resp protocol-violation
! define actions to be applied to unwanted traffic
policy-map type inspect http http-aic-pmap
 class type insp http http-aic-cmap
  reset
  log
! define class-map for stateful http inspection
class-map type inspect match-any http-cmap
 match protocol http
! define class-map for stateful inspection for other traffic
class-map type inspect match-any other-traffic-cmap
 match protocol smtp
 match protocol dns
 match protocol ftp
! define policy-map, associate class-maps and actions
policy-map type inspect priv-pub-pmap
 class type inspect http-cmap
  inspect
  service-policy http http-aic-pmap
 class type inspect other-traffic-cmap
  inspect
```

Усовершенствования при проверке приложения HTTP

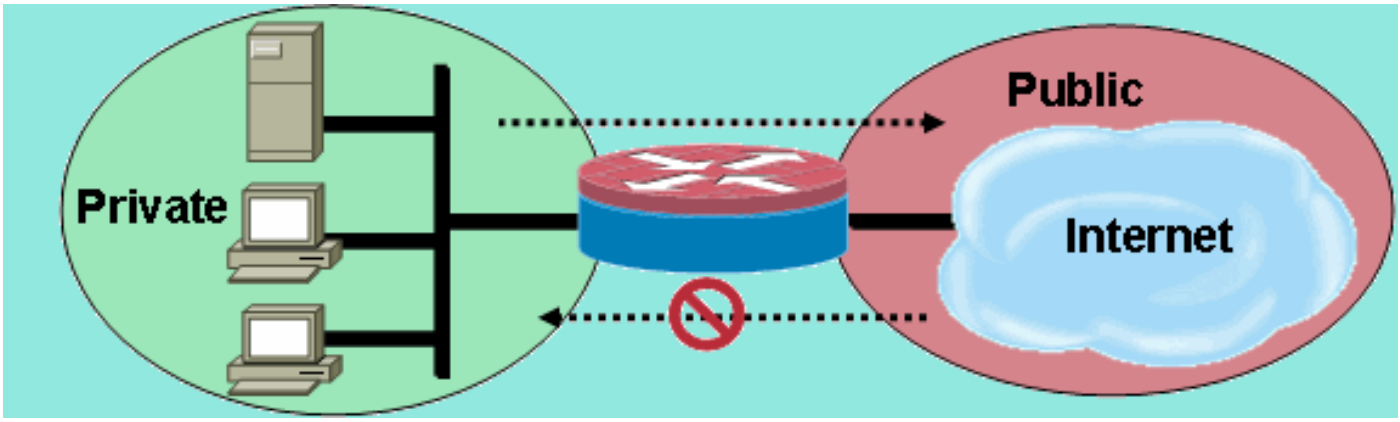
В Cisco IOS версии 12.4(9)T расширены возможности проверки HTTP с помощью ZFW. В межсетевом экране Cisco IOS версии 12.3(14)T реализована проверка приложений HTTP. В версии Cisco IOS 12.4(9)T существующие возможности расширяются за счет добавления следующих функций:

- Появляется возможность разрешать, запрещать и отслеживать запросы и ответы, исходя из названия и значений заголовка. Это позволяет блокировать запросы и ответы, содержащие уязвимые поля заголовка.
- Теперь можно ограничивать размеры различных элементов в заголовках запроса HTTP и ответа (например, максимальную длину URL-адресов, максимальную длину заголовка, максимальное количество заголовков, максимальную длину строки заголовка и т.д). Это позволяет предотвращать переполнение буфера.
- Дается возможность блокировать запросы и ответы с несколькими однотипными заголовками; например, запрос с двумя заголовками длины содержимого.
- Существует возможность блокировать запросы и ответы, содержащие заголовки в кодировке, отличной от ASCII. Благодаря этому можно предотвращать различные атаки, при которых для передачи "червей" и другого вредоносного содержимого на web-серверы применяются двоичные символы и символы в других кодировках.
- Теперь можно группировать методы HTTP по категориям, указанным пользователем, а также гибко блокировать, разрешать и отслеживать каждую из предлагаемых групп. В RFC, посвященном HTTP, разрешается ограниченный набор методов HTTP. Некоторые стандартные методы считаются небезопасными, поскольку их можно использовать для атак через уязвимые места web-сервера. Многие нестандартные методы, как выяснилось, также не обеспечивают безопасность.
- Появился метод блокировки определенных URI, основанный на регулярном выражении, настраиваемом пользователем. Эта функция дает пользователю возможность блокировать пользовательские URI и запросы.
- Можно выполнять спуфинг типов заголовков (особенно типа заголовка сервера) с

помощью строк, которые настраиваются пользователем. Это полезно, если совершающее атаку лицо анализирует ответы web-сервера, узнает как можно больше информации, а затем совершает атаку, использующую слабые места данного web-сервера.

- Возникла возможность блокировать соединение HTTP или предупреждать пользователя о том, что одно или несколько значений параметров HTTP совпадает со значениями, введенными пользователем в качестве регулярного выражения. В том числе возможны следующие контексты значения HTTP: заголовок, тело, имя пользователя, пароль, агент пользователя, строка запроса, строка статуса и декодированные переменные CGI.

Примеры конфигурации при использовании усовершенствований в сфере проверки приложений HTTP предполагают наличие простой сети:



Межсетевой экран группирует трафик по двум классам:

- Трафик HTTP
- Все остальные виды одноканального трафика TCP, UDP и ICMP

Трафик HTTP выделен, что позволяет выполнять специальную проверку web-трафика. Благодаря этому можно настроить регулировку, обратившись к первому разделу данного документа. Настройка проверки приложений HTTP выполняется согласно второму разделу. Конкретные карты классов и карты политик настраиваются для трафика одноранговых сетей и программ для обмена мгновенными сообщениями в третьем разделе данного документа. Разрешено подключение из частной зоны в публичную зону. Возможность подключения из публичной зоны в частную зону не предусмотрена.

Полная конфигурация, реализующая исходную политику, приведена в документе Приложение B, Основная конфигурация межсетевого экрана зональной политики для двух зон.

Настройка усовершенствованных средств проверки приложений HTTP

Проверка приложений HTTP (а также другие политики проверки приложений) требует наличия более сложной конфигурации, чем базовая конфигурация уровня 4. Необходимо настроить классификацию трафика на уровне 7 и политику, чтобы распознавать конкретный трафик, которым требуется управлять, и применять выбранное действие к нужному и нежелательному трафику.

Проверка приложений HTTP (похожая на другие типы проверок приложений) может выполняться только по отношению к трафику HTTP. Таким образом, необходимо определить карты классов и карты политик на уровне 7 для трафика HTTP, затем

определить карту классов на уровне 4 специально для HTTP и применить политику уровня 7 как таковую к проверке HTTP в карте политик уровня 4:

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

Все эти характеристики проверки приложений HTTP определены в карте классов уровня 7:

- **Header inspection — команда, дающая возможность разрешать, запрещать и отслеживать запросы или ответы, заголовок которых совпадает с настроенным регулярным выражением.** К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-6-HTTP_HDR_REGEX_MATCHED`*Использование команды:*`match {request|response|req-resp} header regex <parameter-map-name>` *Пример использования*Настройте политику в отношение трафика http и appfw, чтобы блокировать запрос или ответ, заголовок которого содержит символы в кодировке, отличной от ASCII.
```
parameter-map type regex non_ascii_regex
   pattern “[^\x00-\x80]”
class-map type inspect http non_ascii_cm
   match req-resp header regex non_ascii_regex
policy-map type inspect http non_ascii_pm
   class type inspect http non_ascii_cm
      reset
```
- **Header length inspection — команда, проверяющая длину заголовка запроса или ответа и применяющая действие, если длина превышает выбранное при настройке предельное значение.** Можно выбрать действие "разрешить" или "сбросить". Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-4-HTTP_HEADER_LENGTH`*Использование команды:*`match {request|response|req-resp} header length gt <bytes>` *Пример использования*Настройте политику в отношении трафика http и appfw, чтобы блокировать запросы и ответы, длина заголовка которых превышает 4096 байт.
```
class-map type inspect http hdr_len_cm
   match req-resp header length gt 4096

policy-map type inspect http hdr_len_pm
   class type inspect http hdr_len_cm
      reset
```

- **Header count inspection — команда, проверяющая число строк заголовка (полей) в запросе или ответе и применяющая действие, когда это число превышает выбранное при настройке пороговое значение.** Можно выбрать действие "разрешить" или "сбросить". Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-6- HTTP_HEADER_COUNT`.*Использование команды:*`match {request|response|req-resp} header count gt <number>` *Пример использования*Настройте политику в отношение трафика http и appfw, чтобы блокировать запрос, у которого в заголовке более 16 полей.
```
class-map type inspect http hdr_cnt_cm
   match request header count gt 16

policy-map type inspect http hdr_cnt_pm
   class type inspect http hdr_cnt_cm
      reset
```
- **Проверка поля заголовка — команда, дающая возможность разрешать, запрещать или отслеживать запросы и ответы, содержащие определенное поле и значение заголовка HTTP.** К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-6-HTTP_HDR_FIELD_REGEX_MATCHED`*Использование команды:*`match {request|response|req-resp} header <header-name>` *Пример использования*Настройте политику проверки приложений http, чтобы блокировать шпионское и рекламное ПО:
```
parameter-map type regex ref_regex
   pattern “\.delfinproject\.com”
   pattern “\.looksmart\.com”

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

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

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

policy-map type inspect http spy_adwr_pm
   class type inspect http spy_adwr_cm
      reset
```
- **Header field length inspection — команда, дающая возможность ограничить длину строки в поле заголовка.** К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-6-HTTP_HDR_FIELD_LENGTH`.*Использование команды:*`match {request|response|req-resp} header <header-name> length gt <bytes>` *Пример использования*Настройте политику в отношение трафика http и appfw, чтобы блокировать запрос, в котором длина полей для файла cookie и агента пользователя превышает 256 и 128 символов соответственно.
```
class-map type inspect http hdrline_len_cm
   match request header cookie length gt 256
   match request header user-agnet length gt 128

policy-map type inspect http hdrline_len_pm
   class type inspect http hdrline_len_cm
      reset
```
- **Inspection of header field repetition — команда, проверяющая наличие в запросе или**

ответе повторяющихся полей заголовка. К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Включенное действие журнала вызывает появление следующего сообщения системного журнала:`APPFW-6- HTTP_REPEATED_HDR_FIELDS`.*Использование команды:*`match {request|response|req-resp} header <header-name>` *Пример использования*Настройте политику в отношение трафика http и appfw, чтобы блокировать запрос или ответ, в котором имеется несколько строк заголовка, связанных с длиной содержимого. Это одна из наиболее полезных функций, используемая для предотвращения несанкционированных сеансов.

```
class-map type inspect http multi_occrns_cm
   match req-resp header content-length count gt 1

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

- **Method inspection — HTTP RFC, разрешающая ограниченный набор методов HTTP.** Однако даже некоторые стандартные методы считаются небезопасными, поскольку их можно использовать для атак через уязвимые места web-сервера. Многие нестандартные методы часто применяются для совершения вредоносных действий. Это объясняет необходимость группировки методов по разным категориям и предоставление пользователю права выбрать действие для каждой категории. Эта команда позволяет пользователю гибко сгруппировать методы по различным категориям (например, безопасные методы, небезопасные методы, методы webdav, методы rfc и расширенные методы. К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-6-HTTP_METHOD`.*Использование команды:*`match request method <method>` *Пример использования*Настройте политику в отношение трафика http и appfw, которая группирует методы HTTP по трем категориям: безопасные, небезопасные и webdav. Это показано в таблице. Настройте следующие действия:все безопасные методы разрешены без журналавсе небезопасные методы разрешены с журналомвсе методы webdav блокированы с журналом.`http policy:`

```
class-map type inspect http safe_methods_cm
   match request method get
   match request method head
   match request method option

class-map type inspect http unsafe_methods_cm
   match request method post
   match request method put
   match request method connect
   match request method trace

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

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

- **URI inspection — команда, дающая возможность разрешать, запрещать и отслеживать запросы или ответы, у которых URI совпадает с выбранным при настройке регулярным выражением.** Эта функция дает пользователю возможность блокировать пользовательские URL-адреса и запросы. К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-6- HTTP_URI_REGEX_MATCHED`*Использование команды:*`match request uri regex <parameter-map-name>` *Пример использования*Настройте политику в отношение трафика http и appfw, чтобы блокировать запрос, URI которого совпадает с любым из следующих регулярных выражений:.*cmd.exe.*sex.*gambling`parameter-map type regex uri_regex_cm
   pattern “.*cmd.exe”
   pattern “.*sex”
   pattern “.*gambling”

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

policy-map type inspect http uri_check_pm
   class type inspect http uri_check_cm
      reset`
- **URI length inspection — команда, проверяющая длину URI, отправляемого в запросе, и применяющая выбранное при настройке действие, когда длина превышает настроенное пороговое значение.** К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-6-HTTP_URI_LENGTH`.*Использование команды:*`match request uri length gt <bytes>` *Пример использования*Настройте политику в отношение трафика http и appfw, при которой создается оповещение каждый раз, когда длина URI запроса превышает 3076 байт.`class-map type inspect http uri_len_cm
   match request uri length gt 3076

policy-map type inspect http uri_len_pm
   class type inspect http uri_len_cm
      log`
- **Argument inspection — команда, дающая возможность разрешать, запрещать и отслеживать запрос, у которого аргументы (параметры) совпадают с выбранным при настройке обычной проверки значением###.** К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-6- HTTP_ARG_REGEX_MATCHED`*Использование команды:*`match request arg regex <parameter-map-name>` *Пример использования*Настройте политику в отношение трафика http и appfw, чтобы блокировать запрос, аргументы которого совпадают с любым из следующих регулярных выражений:.*codered.*attack`parameter-map type regex arg_regex_cm
   pattern “.*codered”
   pattern “.*attack”

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

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

- **Argument length inspection — команда, проверяющая длину аргументов, отправляемых в запросе, и применяет выбранное при настройке действие, когда длина превышает настроенное пороговое значение.** К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-6- HTTP_ARG_LENGTH`.*Использование команды:*`match request arg length gt <bytes>` *Пример использования*Настройте политику в отношение трафика http и appfw, при которой создается оповещение каждый раз, когда длина аргумента запроса превышает 512 байт.
```
class-map type inspect http arg_len_cm
   match request arg length gt 512

policy-map type inspect http arg_len_pm
   class type inspect http arg_len_cm
      log
```
- **Body inspection — интерфейс командной строки (CLI), который дает пользователю возможность указать список регулярных выражений, которые должны сравниваться с телом запроса или ответа.** К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-6- HTTP_BODY_REGEX_MATCHED`*Использование команды:*`match {request|response|reg-resp} body regex <parameter-map-name>` *Пример использования*
```
http
appfw ,   ,      *[Aa][Tt][Tt][Aa][Cc][Kk]parameter-map type regex body_regex
   pattern “.*[Aa][Tt][Tt][Aa][Cc][Kk]”

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

policy-map type inspect http body_match_pm
   class type inspect http body_match_cm
      reset
```
- **Body (Content) length inspection — команда, проверяющая размер сообщения, отправляемого с помощью запроса или ответа.** К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-4- HTTP_CONTENT_LENGTH`*Использование команды:*`match {request|response|req-resp} body length lt <bytes> gt <bytes>` *Пример использования*Настройте политику http и appfw так, чтобы она блокировала сеанс http, содержащий сообщение размером более 10 килобайт в запросе или ответе.
```
class-map type inspect http cont_len_cm
   match req-resp header content-length gt 10240

policy-map type inspect http cont_len_pm
   class type inspect http cont_len_cm
      reset
```
- **Status line inspection — команда, дающая пользователю возможность указать список регулярных выражений, которые должны сравниваться со строкой статуса ответа.** К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-6- HTTP_STLINE_REGEX_MATCHED`.*Использование команды:*`match response status-line regex <class-map-name>` *Пример использования*Настройте http appfw так, чтобы при каждой попытке доступа к запрещенной странице в журнал записывалось оповещение.
```
403,      :
HTTP/1.0 403 page forbidden\r\n.parameter-map type regex status_line_regex
```

```
   pattern “[Hh][Tt][Tt][Pp][/][0-9][.][0-9][ \t]+403”

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

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

- **Content-type inspection — команда, проверяющая, находится ли в списке поддерживаемых типов содержимого заголовок сообщения.** Кроме того, она позволяет выяснить, совпадает ли тип содержимого заголовка с содержимым данных сообщения или с фрагментом, содержащим тело объекта. **Если настроено ключевое слово mismatch, эта команда сверяет тип содержимого с принятым значением поля в сообщении запроса.** К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление соответствующего сообщения системного журнала:`APPFW-4- HTTP_CONT_TYPE_VIOLATION, APPFW-4- HTTP_CONT_TYPE_MISMATCH, APPFW-4- HTTP_CONT_TYPE_UNKNOWN`*Использование команды:*`match {request|response|req-resp} header content-type [mismatch|unknown|violation]`*Пример использования*Настройте политику http appfw так, чтобы она блокировала сеанс http, в котором есть запросы и ответы с неизвестным типом содержимого.
```
class-map type inspect http cont_type_cm
   match req-resp header content-type unknown

policy-map type inspect http cont_type_pm
   class type inspect http cont_type_cm
      reset
```
- **Port-misuse inspection — команда, используемая для предотвращения неправильного использования порта 80 (http) другими приложениями (например, программами для обмена мгновенными сообщениями, туннелирования и т.д. К запросу или ответу, соответствующему критериям карты классов, можно применить действие разрешения или сброса.** Добавление действия журнала вызывает появление соответствующего сообщения системного журнала:`APPFW-4- HTTP_PORT_MISUSE_TYPE_IM APPFW-4-HTTP_PORT_MISUSE_TYPE_P2P APPFW-4-HTTP_PORT_MISUSE_TYPE_TUNNEL`*Использование команды:*`match request port-misuse {im|p2p|tunneling|any}`*Пример использования*Настройте политику http appfw так, чтобы сеанс http, неправильно используемый для работы с приложением для обмена мгновенными сообщениями, блокировался.
```
class-map type inspect http port_misuse_cm
   match request port-misuse im

policy-map type inspect http port_misuse_pm
   class type inspect http port_misuse_cm
      reset
```
- **Strict-http inspection — команда, включающая строгую проверку соответствия протоколу, включающую сравнение с запросами HTTP и ответами.** К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-4- HTTP_PROTOCOL_VIOLATION`*Использование команды:*`match req-resp protocol-violation`*Пример использования*Настройте политику http appfw так, чтобы запросы или ответы, нарушающие RFC 2616, блокировались:
```
class-map type inspect http proto-viol_cm
   match req-resp protocol-violation

policy-map type inspect http proto-viol_pm
```

```
   class type inspect http proto-viol_cm
      reset
```

- **Transfer-Encoding inspection — команда, дающая возможность разрешать, запрещать или отслеживать запрос или ответ, тип шифрования передачи которого совпадает с типом, выбранным во время настройки.** К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-6- HTTP_TRANSFER_ENCODING`*Использование команды:*`match {request|response|req-resp} header transfer-encoding {regex <parameter-map-name> |gzip|deflate|chunked|identity|all}` *Пример использования*Настройте политику http appfw, чтобы блокировать запрос или ответ, тип кодировки которого предполагает сжатие данных###.
```
class-map type inspect http trans_encoding_cm
   match req-resp header transfer-encoding type compress

policy-map type inspect http trans_encoding_pm
   class type inspect http trans_encoding_cm
      reset
```

- **Java Applet inspection — команда, проверяющая наличие в ответе сетевого приложения Java и применяющая выбранное при настройке действие после обнаружения сетевого приложения.** К запросу или ответу, совпадающему с критериями карты классов, могут применяться такие действия как разрешение или сброс. Добавление действия журнала вызывает появление следующего сообщения системного журнала:`APPFW-4-HTTP_JAVA_APPLET` *Использование команды:*`match response body java-applet`*Пример использования*Настройте политику http appfw, чтобы блокировать сетевые приложения Java.
```
class-map type inspect http java_applet_cm
   match response body java-applet

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

Поддержка ZFW управления приложениями для обмена мгновенными сообщениями и работы в одноранговой сети

В Cisco IOS версии 12.4(9)T появилась поддержка ZFW приложений для обмена мгновенными сообщениями и работы в одноранговой сети.

Впервые поддержка управления приложениями для обмена мгновенными сообщениями была реализована в Cisco IOS версии 12.4(4)T. Интерфейс первой версии ZFW не поддерживал приложение для обмена мгновенными сообщениями в интерфейсе ZFW. При необходимости управления приложениями для обмена мгновенными сообщениями пользователям не удавалось провести миграцию на интерфейс конфигурации ZFW. ZFW в Cisco IOS версии 12.4(9)T поддерживает проверку программ для обмена мгновенными сообщениями (Yahoo! Messenger (YM), MSN Messenger (MSN) и AOL Instant Messenger (AIM)).

Cisco IOS версии 12.4(9)T — это первая версия Cisco IOS, обладающая встроенной поддержкой межсетевого экрана IOS при работе с приложениями для обмена файлами в одноранговой сети.

Как при проверке программ для обмена файлами в одноранговой сети, так и при проверке программ для обмена мгновенными сообщениями предлагаются политики уровня 4 и

уровня 7 для трафика приложений. Это означает, что с помощью ZFW можно выполнять основную проверку с контролем состояния, чтобы разрешать или запрещать трафик, а также точно управлять действиями при использовании различных протоколов на уровне 7, чтобы разрешать одни действия приложений и запрещать другие.

Проверки и управление приложениями P2P

В SDM 2.2 реализовано управление приложениями для обмена данными в одноранговой сети в разделе "Конфигурация межсетевого экрана". В SDM применяется сетевое распознавание приложений (NBAR) и политика QoS, позволяющие распознавать и регулировать активность приложения для обмена данными в одноранговой сети, доводя линейную скорость до нуля и блокируя весь трафик в одноранговой сети. Это приводило к проблема в связи с тем, что пользователям интерфейса командной строки (CLI), ожидающим поддержки одноранговой сети в CLI межсетевого экрана IOS, не удавалось настроить блокировку трафика одноранговой сети в CLI, если они не знали о том, что необходимо настраивать NBAR/QoS. В Cisco IOS версии 12.4(9)T имеется интерфейс командной строки ZFW со встроенным управлением одноранговой сетью, где для выявления активности приложений для обмена данными в одноранговой сети применяется NBAR. Эта версия ПО поддерживает ряд протоколов приложений для обмена данными в одноранговой сети:

- BitTorrent
- eDonkey
- FastTrack
- Gnutella
- KaZaA / KaZaA2
- WinMX

Приложения для обмена данными в одноранговой сети особенно сложно обнаружить из-за смены портов и других уловок для избегания обнаружения, а также из-за проблем, вызванных быстрыми изменениями и обновлениями таких приложений, которые модифицируют поведение протоколов. ZFW сочетает встроенную проверку межсетевого экрана с контролем состояния с возможностями распознавания трафика NBAR, обеспечивая управление приложениями для обмена данными в одноранговой сети в интерфейсе конфигурации CPL для ZFW. NBAR обладает двумя отличными преимуществами:

- Дополнительный механизм распознавания приложений на базе эвристик, который выявляет приложения несмотря на сложное и трудноуловимое поведение
- Обширная инфраструктура, обладающая механизмом обновления, который позволяет быть в курсе обновлений и изменений протоколов

Настройка проверки одноранговой сети

Как упоминалось ранее, проверка одноранговой сети и управление ею позволяют выполнить как проверку с контролем состояния на уровне 4, так и проверку приложений на уровне 7.

Проверка на уровне 4 настраивается так же, как и службы приложений, если проверка исходных портов служб приложений будет достаточно:

```
class-map type inspect match-any my-p2p-class
```

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

Обратите внимание на такой дополнительный параметр как подпись в выражении match protocol [имя службы]. При добавлении параметра подписи в конец оператора match protocol к трафику применяются эвристики NBAR для поиска в нем указаний на ту или иную активность приложений для обмена данными в одноранговой сети. В том числе выявляется смена портов и другие изменения поведения приложения, внесенные во избежание распознавания. Этот уровень проверки трафика дается ценой повышенной загрузки процессора и сниженной пропускной способности сети. Если подпись не применяется, для выявления смены портов не будет применяться эвристический анализ на основе NBAR, и это не вызовет загрузки процессора на том же уровне.

Недостаток встроенной проверки службы состоит в том, что не удается сохранять контроль над приложениями для обмена данными в одноранговой сети, когда приложение скачкообразно переходит к применению нестандартных порта источника и порта назначения или когда в результате обновления приложения оно начинает работу с порта, имеющего нераспознанный номер:

Приложение	Собственные порты (распознанные по списку PAM в версии 12.4(15)T)
bittorrent	TCP 6881-6889
edonkey	TCP 4662
fasttrack	TCP 1214
gnutella	TCP 6346-6349 TCP 6355,5634 UDP 6346-6348
kazaa2	Зависит от PAM
winmx	TCP 6699

Если требуется разрешить (проверить) трафик при обмене данными в одноранговой сети, возможно, потребуется дополнительная настройка. Некоторые приложения могут использовать несколько одноранговых сетей или обладать определенным поведением, которое, возможно, нужно будет учитывать, чтобы разрешить работу приложения:

- Клиенты BitTorrent обычно обмениваются данными с так называемыми “трекерами” (серверами одноранговых каталогов) с помощью протокола http, использующего какой-либо нестандартный порт. Обычно это порт TCP 6969, но, возможно, потребуется проверить порт, специально предназначенный для трекера. **Если необходимо разрешить BitTorrent, учесть наличие дополнительного порта лучше всего следующим образом: настроить HTTP как один из протоколов соответствия### и добавить TCP 6969 к HTTP с помощью команды ip port-map:**
  `ip port-map http port tcp 6969` При этом потребуется определить http и bittorrent как критерии соответствия, применяемые в карте классов.
- eDonkey по всей видимости инициирует подключения, которые распознаются как имеющие отношение к eDonkey и Gnutella.
- Проверка KaZaA всецело зависит от обнаружения подписи NBAR.

Проверка приложений на уровне 7 дополняет проверку на уровне 4 возможностью распознавать и применять действия для конкретных служб (например, избирательно

блокировать трафик или разрешать возможности поиска файлов, передачи файлов и текстового чата). Возможности служб различны у разных служб.

Проверка приложений для обмена данными в одноранговой сети выполняется аналогично проверке приложений HTTP:

```
!configure the layer-7 traffic characteristics:
class-map type inspect [p2p protocol] match-any p2p-l7-cmap
 match action
!
!configure the action to be applied to the traffic
!matching the specific characteristics:
policy-map type inspect [p2p protocol] p2p-l7-pmap
 class type inspect p2p p2p-l7-cmap
  [ reset | allow ]
  log
!
!define the layer-4 inspection policy
class-map type inspect match-all p2p-l4-cmap
 match protocol [p2p protocol]
!
!associate layer-4 class and layer-7 policy-map
!in the layer-4 policy-map:
policy-map type inspect private-allowed-policy
 class type inspect p2p-l4-cmap
  [ inspect | drop | pass ]
  service-policy p2p p2p-l7-pmap
```

Проверка приложений для обмена данными в одноранговой сети обладает возможностями приложения, предназначенными для набора приложений, который можно проверить на уровне 4:

- edonkey
- fasttrack
- gnutella
- kazaa2

Каждое из этих приложений обладает различными параметрами критериев соответствия:

edonkey

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

fasttrack

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

gnutella

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

kazaa2

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

Новые определения протоколов одноранговой сети или обновления для существующих протоколов можно загрузить с помощью функции динамического обновления pdlm, относящейся к NBAR. Вот команда настройки, используемая для загрузки нового PDLM:

ip nbar pdlm *<file-location>*

Новый протокол доступен в командах match protocol для класса type inspect. Если в новом протоколе одноранговой сети имеются службы (подпротоколы), становятся доступны новые типы карт классов проверки на уровне 7, а также критерии соответствия уровня 7.

Проверка и управление приложениями для обмена мгновенными сообщениями

В Cisco IOS версии 12.4(4)T появилась функция проверки и управления приложениями для обмена мгновенными сообщениями. Поддержка обмена мгновенными сообщениями не предусматривалась в ZFW версии 12.4(6)T, и пользователи не могли применить управление обменом мгновенными сообщениями и ZFW в одной и той же политике межсетевого экрана, так как ZFW устаревшие функции межсетевого экрана не могут сосуществовать в данном интерфейсе.

Cisco IOS версии 12.4(9)T поддерживает проверку с контролем состояния и управлением приложениями для следующих служб обмена сообщениями:

- AOL Instant Messenger
- MSN Messenger
- Yahoo! Messenger

Проверка обмена мгновенными сообщениями слегка отличается от проверки большинства служб, поскольку она зависит от управления доступом к определенной группе узлов для каждой службы. Службы для обмена мгновенными сообщениями обычно зависят от относительно постоянной группы серверов каталогов, клиенты которых должны быть способны связываться друг с другом, чтобы иметь доступ к службе обмена мгновенными сообщениями. Приложения для обмена сообщениями обычно очень сложно контролировать с точки зрения протокола или службы. Наиболее эффективный способ управлять этими приложениями состоит в ограничении доступа к фиксированным серверам обмена мгновенными сообщениями.

Настройка проверки обмена мгновенными сообщениями

Благодаря проверке программ для обмена мгновенными сообщениями и управлению ими можно выполнить как проверку с контролем состояния на уровне 4, так и управление приложениями на уровне 7.

Проверка на уровне 4 настраивается аналогично другим службам приложений:

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

Приложения для обмена мгновенными сообщениями способны связываться со своими серверами через несколько портов, чтобы поддерживать работоспособность. Если нужно разрешить ту или иную службу обмена мгновенными сообщениями, применив действие проверки, для определения разрешенного доступа к серверам службы обмена мгновенными сообщениями может не потребоваться список серверов. Однако настройка карты классов, в которой указывается заданная служба обмена мгновенными сообщениями (например, AOL Instant Messenger), и применение действия отбрасывания в связанной с ней карте политик

может заставить клиент этой службы попытаться найти другой порт, через который разрешено подключаться к Интернету. Если службе не нужно разрешать подключение или если требуется ограничить способность службы обмена мгновенными сообщениями к текстовому чату, необходимо определить список серверов, чтобы дать ZFW возможность идентифицировать трафик, связанный с приложением для обмена мгновенными сообщениями:

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

Например, список серверов обмена мгновенными сообщениями Yahoo определяется следующим образом:

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

Нужно будет применить список серверов к определению протокола:

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

Необходимо настроить команды ip domain lookup и ip name-server ip.ad.re.ss, чтобы включить разрешение имен.

Имена серверов, используемые при обмене мгновенными сообщениями, достаточно динамичны. Нужно будет периодически удостоверяться в том, что списки серверов обмена мгновенными сообщениями приведены полностью и правильны.

Проверка приложений на уровне 7 дополняет проверку на уровне 4 возможностью распознавать и применять действия для конкретных служб (например, избирательно блокировать трафик или разрешать возможности текстового чата, продолжая запрещать возможности других служб).

На данный момент проверка приложений для обмена мгновенными сообщениями позволяет разделять активность текстового чата и все остальные службы приложений. Чтобы ограничить активность при обмене мгновенными сообщениями, оставив только текстовый чат, настройте политику уровня 7:

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

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

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

Примените политику уровня 7 к ранее настроенной политике Yahoo! Messenger:

```
class-map type inspect match-any my-im-class
```

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

Фильтрация URL-адресов

ZFW обладает возможностью фильтрации по URL-адресам с целью ограничения доступа к содержимому в соответствии с "белым" и "черным" списками, которые определены для маршрутизатора. Кроме того, для этого может применяться перенаправление доменных имен на сервер фильтрации по URL-адресам, чтобы проверить доступ к определенным доменам. Фильтрация по URL-адресам в ZFW, входящем в состав Cisco IOS версий 12.4(6)T-12.4(15)T, применяется в качестве дополнительного действия политики, как и проверка приложений.

В случае серверной фильтрации по URL-адресам необходимо определить карту параметров, описывающую конфигурацию сервера urlfilter:

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

Если предпочитаются статические белые или черные списки, можно определить список разрешенных либо запрещенных доменов или поддоменов, тогда как противоположное действие применяется к трафику, который не указан в списке:

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

Если "черный" список URL-адресов определен с помощью параметров запрета в определениях исключенных доменов, будут разрешены все остальные домены. Если составлены любые определения действия “разрешить”, должны быть явно указаны все домены, которые будут разрешены, аналогично тому, как это делается с помощью списков управления доступом IP.

Настройте карту классов, которая будет совпадать с трафиком HTTP:

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

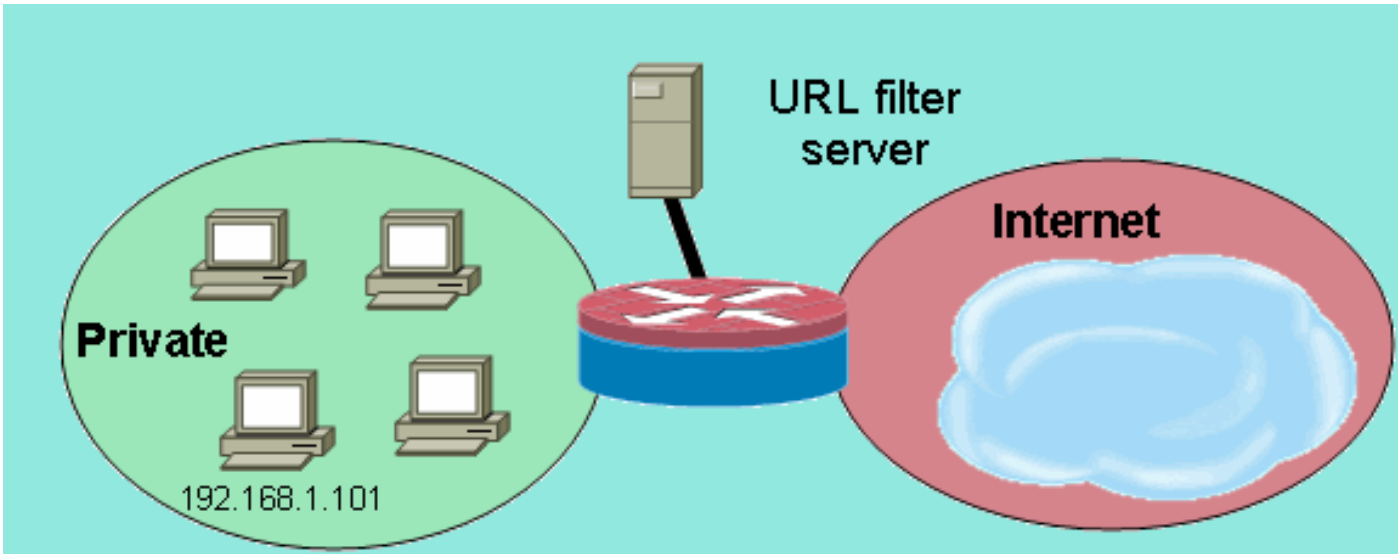
Определите карту политик, связывающую ее карту классов с действиями inspect и urlfilter:

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

Это минимально необходимая настройка, которая нужна для связи с сервером фильтрации URL-адресов. Чтобы определить дополнительное поведение при фильтрации URL-адресов, имеется несколько возможностей.

В некоторых развернутых сетях может потребоваться применение фильтрации по URL-адресам для ряда узлов и подсетей, в то время как для других узлов она может не выполняться. Например, сервер фильтрации по URL-адресам должен проверять трафик HTTP всех узлов в частной зоне на рисунке 9 за исключением узла 192.168.1.101.

Рис. 9: Топология примера фильтрации URL-адресов



Этого можно достичь, определив две разные карты классов:

- Одну карту классов, соответствующую трафику HTTP для большей группы узлов, подлежащей фильтрации по URL-адресам.
- Еще одну карту классов для меньшей группы узлов, для которой не выполняется фильтрация URL-адресов. Вторая карта классов будет соответствовать трафику HTTP, а также списку узлов, исключаемых из политики фильтрации по URL-адресам.

Обе карты классов настроены в карте политик, но действие urlfilter будет применено только к одной из них:

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

Управление доступом к маршрутизатору

Большинство специалистов по сетевой безопасности с неохотой открывают доступ к интерфейсам управления маршрутизатора (например, SSH, Telnet, HTTP, HTTPS, SNMP и т.д.) со стороны публичного Интернета. При определенных обстоятельствах необходимо также управлять доступом к маршрутизатору со стороны локальной сети. Cisco IOS располагает рядом возможностей по ограничению доступа к различным интерфейсам, включая семейство функций Network Foundation Protection (NFP), различные механизмы управления доступом к интерфейсам управления и собственной зоне ZFW. Необходимо проверить другие функции (например, управление доступом к VTY), защиту панели управления и управление доступом SNMP, чтобы определить, какое сочетание функций управления маршрутизатором лучше всего подходит для конкретного приложения.

Обычно семейство функций NFP лучше всего подходит для управления трафиком, поступающим в сам маршрутизатор. Информацию по защите маршрутизатора с помощью

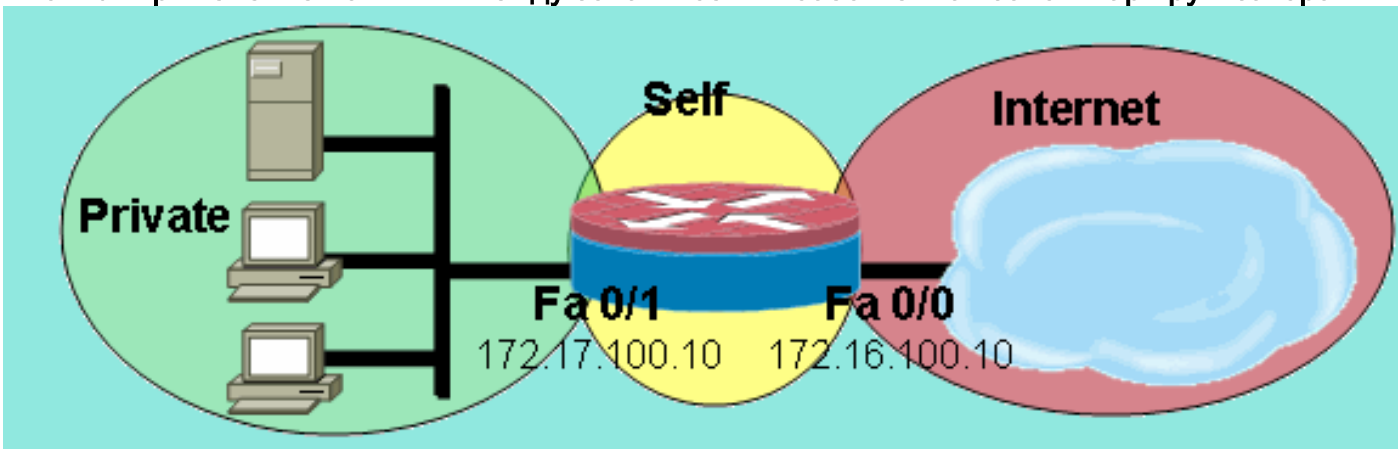
функций NFP см. в документе Обзор безопасности панели управления в Cisco IOS.

Если вы захотите применить ZFW для управления трафиком, поступающим на IP-адреса самого маршрутизатора или отправляемым с них, необходимо понять, что политика межсетевого экрана, выбранная по умолчанию, и его возможности отличаются от политики и возможностей, применяемых в отношение транзитного трафика. Транзитный трафик определяется как сетевой трафик, для которого IP-адреса источника и назначения не совпадают с IP-адресами, примененными к любому из интерфейсов маршрутизатора. Этот трафик не вызывает отправку маршрутизатором, например, сообщений управления сетью (таких как сообщения о завершении времени жизни (TTL) для ICMP или сообщения о недоступности сети или узла).

ZFW применяет политику запрета всего трафика, которая применяется к трафику, перемещающемуся между зонами, за исключением трафика в любую зону, который направляется напрямую на адреса интерфейсов маршрутизатора и который неявно разрешен. Упоминание об этом содержится в общих правилах. Это гарантирует сохранение подключения к интерфейсам управления маршрутизатора после применения конфигурации зонального межсетевого экрана к маршрутизатору. Если та же политика запрета всего трафика повлияла на подключение непосредственно к маршрутизатору, необходимо применить полную конфигурацию политики управления перед настройкой зон в маршрутизаторе. Скорее всего это нарушит подключение управления, если политика была неправильно реализована или применена в неверном порядке.

Когда интерфейс настроен в качестве участника зоны, в эту зону включаются узлы, подключенные к этому интерфейсу. Однако входящий и исходящий трафик для IP-адресов интерфейсов маршрутизатора не регулируется зональными политиками (за исключением обстоятельств, описанных в примечании, которое приводится вслед за рисунком 10). Вместо этого все IP-интерфейсы в маршрутизаторе автоматически причисляются к собственной зоне во время настройки ZFW. Чтобы управлять перемещением трафика IP к интерфейсам маршрутизатора из различных зон маршрутизатора, необходимо применять политики блокировки или разрешения/проверки трафика и обратного трафика между зоной и собственной зоной маршрутизатора. (См. рис. 10.).)

Рис. 10: Применение политики между зонами сети и собственной зоной маршрутизатора



Хотя по умолчанию в маршрутизаторе выбрана политика разрешения трафика между всеми зонами и собственной зоной, если настроена политика для трафика из любой зоны в собственную зону и не настроена политика для трафика из собственной зоны в настраиваемые пользователем и подключенные к интерфейсам зоны маршрутизатора, любой трафик маршрутизатора сталкивается с политикой для трафика из подключенной зоны к собственной зоне на обратном пути из маршрутизатора и блокируется. Таким

образом, трафик, исходящий из маршрутизатора, необходимо проверять, чтобы разрешить его возврат в собственную зону.

Примечание: Программное обеспечение Cisco IOS всегда использует IP-адрес, привязанный к интерфейсу “самые близкие” адресаты для трафика, такие как системный журнал, tftp, telnet и другие сервисы уровня управления, и подвергает этот трафик самозональной политике межсетевого экрана. Однако, если сервис определяет определенный интерфейс как команды использования source-interface, которые включают, но не ограниченные **logging source-interface [номер типа]**, **ip tftp source-interface [номер типа]** и **ip telnet source-interface [номер типа]**, трафик подвергнут самозоне.

Примечание: Некоторые сервисы (особенно сервисы передачи голоса по IP маршрутизаторов) используют эфемерные или неизменяемые интерфейсы, которые не могут быть назначены на зоны безопасности. Эти службы могут неправильно работать, если их трафик невозможно связать с настроенной зоной безопасности.

Ограничения политики собственной зоны

Политика собственной зоны имеет ограниченные функциональные возможности по сравнению с политиками, которые доступны для пар зон транзитного трафика:

- Как и в случае классической проверки с контролем состояния, созданный маршрутизатором трафик может быть связан только с протоколами TCP, UDP, ICMP. При работе с H.323. применяется ###комплексная проверка протоколов.
- Проверка приложений недоступна для политик собственной зоны.
- Ограничение сеанса и скорости невозможно настроить в политиках собственной зоны.

Настройка политики собственной зоны

Почти при любых обстоятельствах указанные ниже политики доступа рекомендуются для служб управления маршрутизатором:

- Запретите все подключения Telnet, поскольку нешифрованный протокол Telnet позволяет легко узнать учетные данные пользователя и другую конфиденциальную информацию.
- Разрешите подключения SSH, устанавливаемые любым пользователем в любой зоне. SSH шифрует учетные данные пользователей и данные сеанса. Это дает возможность защититься от злоумышленников, применяющих средства захвата пакетов, чтобы следить за действиями пользователей, похищая учетные данные или конфиденциальную информацию (например, конфигурацию маршрутизатора). В SSH версии 2 предоставляется усиленная защита и исправлены уязвимости, присущие SSH версии 1.
- Разрешите подключение HTTP к маршрутизатору из частных зон, если она заслуживает доверия. В противном случае если частная зона обладает потенциалом, который позволяет злоумышленникам похищать информацию, в HTTP не применяется шифрование для защиты трафика управления. Это может привести к раскрытию учетных данных пользователей или конфигурации.
- Разрешите подключения HTTPS из любой зоны. HTTPS шифрует данные о сеансе и учетные данные пользователя как и SSH.

- Ограничьте доступ SNMP к определенному узлу или подсети. SNMP можно использовать для изменения конфигурации маршрутизатора и узнавания сведений о конфигурации. SNMP необходимо настроить с помощью управления доступом в различных сообществах.
- Заблокируйте запросы ICMP из публичного Интернета на адрес частной зоны (предполагается, что адрес частной зоны является маршрутизируемым). Один или несколько публичных адресов по необходимости может быть открыт для трафика ICMP при выяснении неисправностей сети. Могут быть предприняты несколько атак ICMP, нацеленных на переполнение ресурсов маршрутизатора и позволяющих разведать топологию и архитектуру сети.

Маршрутизатор может применить этот тип политики с добавлением двух пар зон для каждой зоны, которой необходимо управлять. Каждая пара зон для исходящего или входящего трафика собственной зоны маршрутизатора должна обладать соответствующей политикой в отношение обратного трафика, если последний не будет создаваться для отправки в противоположном направлении. Можно применить по одной карте политик для пары зон входящего и исходящего трафика, которая описывает весь трафик, либо применить конкретные карты политик для пары зон. Конфигурация определенных пар зон для карты политик обеспечивает детализацию при просмотре активности, соответствующей каждой из карт политик.

Предположим, что в примере сети управляющая станция SNMP расположена по адресу 172.17.100.11, а сервер TFTP — по адресу 172.17.100.17. Вот выходные данные, которые служат примером всей политики доступа к интерфейсу управления:

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

```
zone-pair security priv-self source private destination self
 service-policy type inspect to-self-pmap
zone-pair security net-self source internet destination self
 service-policy type inspect to-self-pmap
zone-pair security self-priv source self destination private
 service-policy type inspect from-self-pmap
zone-pair security self-net source self destination internet
 service-policy type inspect from-self-pmap

!
interface FastEthernet 0/0
 ip address 172.16.100.10
 zone-member security internet
!
interface FastEthernet 0/1
 ip address 172.17.100.10
 zone-member security private
!
access-list 120 permit icmp 172.17.100.0 0.0.0.255 any
access-list 120 permit icmp any host 172.17.100.10 echo
access-list 120 deny icmp any any
access-list 120 permit tcp 172.17.100.0 0.0.0.255 host 172.17.100.10 eq www
access-list 120 permit tcp any any eq 443
access-list 120 permit tcp any any eq 22
access-list 120 permit udp any host 172.17.100.10 eq snmp
access-list 121 permit udp host 172.17.100.17 host 172.17.100.10
access-list 122 permit udp host 172.17.100.10 host 172.17.100.17
```

К сожалению, политика собственной зоны не дает возможности проверить передачу данных по протоколу TFTP. Таким образом, межсетевой экран должен пропускать весь трафик, направленный на сервер TFTP или отправляемый с него, если межсетевому экрану требуется пропускать трафик TFTP.

Если маршрутизатор будет прерывать подключения IPSec VPN, необходимо также определить политику, чтобы пропускать трафик IPSec ESP, IPSec AH, ISAKMP и NAT-T IPSec (UDP 4500). Это зависит от того, что необходимо для служб, которые планируется использовать. Помимо указанной выше политики можно применить следующую политику. Обратите внимание на изменение в картах политик, где карта политик для трафика VPN была вставлена с действием пропускания. Обычно шифрованному трафику можно доверять, если в выбранной политике безопасности указано, что требуется разрешить исходящий и входящий шифрованный трафик для определенных конечных точек.

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

```
access-list 123 permit udp any any eq 4500
access-list 123 permit ah any any
access-list 123 permit udp any any eq 500
```

Зональный межсетевой экран и Wide-Area Application Services

Сведения о приложении с примерами конфигурации и указаниями по применению см. в документе Примечание к выпуску Cisco Wide Area Application Services (версии 4.0.13) — новые функции в версии 4.0.13

Наблюдение за межсетевым экраном зональной политики с помощью команд show и debug

В ZFW появились новые команды, позволяющие просматривать конфигурацию политики и следить за активностью межсетевого экрана.

Ознакомьтесь с описанием зоны и интерфейсами, которые содержатся в указанной зоне:

show zone security *[<zone-name>]*
Если название зоны не включено, команда отображает информацию по всем настроенным зонам.

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

Просмотрите зону источника, зону назначения и политику, закрепленную за парой зон:

show zone-pair security [source *<source-zone-name>***] [destination** *<destination-zone-name>***]**
Когда не указан ни источник, ни назначение, отображаются все пары зон с источником, назначением и соответствующей политикой. Когда упомянута только зона источника или назначения, отображаются все пары зон, содержащие эту зону в роли источника или назначения.

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

Отображает указанную карту политик:

show policy-map type inspect [*<policy-map-name>* **[class** *<class-map-name>***]]**
Если название карты политик не указано, отображаются все карты политик проверочного типа (включая карты политик уровня 7, содержащие подтип).

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

Отображается динамическая статистика по карте политик проверочного типа, имеющаяся для указанной пары зон.

show policy-map type inspect zone-pair [*zone-pair-name***] [sessions]**
Если упомянут параметр no zone-pair name, отображаются карты политик для всех пар зон.

Параметр sessions вызывает показ сеансов проверки, созданных приложением карты

политик в указанной паре зон.

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

Ключевое слово urlfilter приводит к показу статистики, связанной с действием urlfilter и относящейся к указанной карте политик (или к картам политик для всех целей, если не задано название пары зон):

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

Если наряду с ключевым словом cache указано ключевое слово urlfilter, на экране отображается кэш urlfilter (IP-адресов).

Сводка по команде show policy-map для проверочных карт политик:

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

Настройка защиты против DoS-атак межсетевого экрана зональной политики

ZFW обеспечивает защиту от DoS-атак, оповещая сетевых инженеров о значительных изменениях сетевой активности и сокращая нежелательную активность, чтобы снизить воздействие изменений сетевой активности. ZFW располагает отдельными счетчиками для каждого класса политик карты политик. Поэтому если одна и та же карта классов применяется для двух различных пар зон, будут применяться два разных набора счетчиков для защиты от DoS-атак.

ZFW по умолчанию обеспечивает подавление DoS-атак в версиях Cisco IOS, выпущенных до версии 12.4(11)T. Поведение по умолчанию при защите от DoS-атак изменилось с выходом версии 12.4(11)T. Дополнительное обсуждение и процедуру настройки защиты от DoS-атак с помощью ZFW см. в документе Настройка защиты против DoS-атак межсетевого экрана Cisco IOS.

Дополнительные сведения о DoS-атаках через пакеты TCP SYN см. в документе Определение стратегий защиты от атак типа "отказ в обслуживании" TCP SYN.

Приложение

Приложение A: Основная конфигурация

```
ip subnet-zero
ip cef
!
bridge irb
!
interface FastEthernet0
 ip address 172.16.1.88 255.255.255.0
```

```
 duplex auto
 speed auto
!
interface FastEthernet1
 ip address 172.16.2.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet2
 switchport access vlan 2
!
interface FastEthernet3
 switchport access vlan 2
!
interface FastEthernet4
 switchport access vlan 1
!
interface FastEthernet5
 switchport access vlan 1
!
interface FastEthernet6
 switchport access vlan 1
!
interface FastEthernet7
 switchport access vlan 1
!
interface Vlan1
 no ip address
 bridge-group 1
!
interface Vlan2
 no ip address
 bridge-group 1
!
interface BVI1
 ip address 192.168.1.254 255.255.255.0
 ip route-cache flow
!
ip classless
ip route 0.0.0.0 0.0.0.0 172.16.1.1
!
bridge 1 protocol ieee
bridge 1 route ip
!
end
```

Приложение Б: Финальная (полная) конфигурация

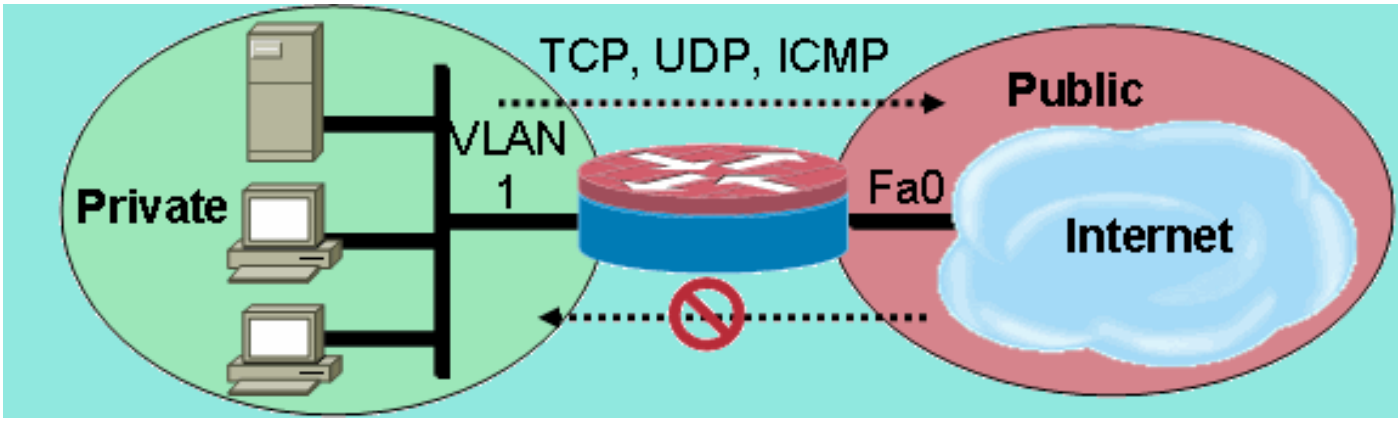
```
ip subnet-zero
ip cef
!
ip port-map user-Xwindows port tcp from 6900 to 6910
!
class-map type inspect match-any L4-inspect-class
 match protocol tcp
 match protocol udp
 match protocol icmp
class-map type inspect match-any L7-inspect-class
  match protocol ssh
  match protocol ftp
  match protocol pop
  match protocol imap
  match protocol esmtp
  match protocol http
```

```
class-map type inspect match-any dns-http-class
 match protocol dns
 match protocol http
class-map type inspect match-any smtp-class
 match protocol smtp
class-map type inspect match-all dns-http-acl-class
 match access-group 110
 match class-map dns-http-class
class-map type inspect match-all smtp-acl-class
 match access-group 111
 match class-map smtp-class
class-map type inspect match-any Xwindows-class
 match protocol user-Xwindows
class-map type inspect match-any internet-traffic-class
 match protocol http
 match protocol https
 match protocol dns
 match protocol icmp
class-map type inspect http match-any bad-http-class
 match  port-misuse all
 match  strict-http
!
policy-map type inspect clients-servers-policy
 class type inspect L4-inspect-class
  inspect
policy-map type inspect private-dmz-policy
  class type inspect L7-inspect-class
   inspect
policy-map type inspect internet-dmz-policy
 class type inspect dns-http-acl-class
  inspect
 class type inspect smtp-acl-class
  inspect
policy-map type inspect servers-clients-policy
 class type inspect Xwindows-class
  inspect
policy-map type inspect private-internet-policy
  class type inspect internet-traffic-class
   inspect
  class type inspect bad-http-class
   drop
!
zone security clients
zone security servers
zone security private
zone security internet
zone security dmz
zone-pair security private-internet source private destination internet
  service-policy type inspect private-internet-policy
zone-pair security servers-clients source servers destination clients
  service-policy type inspect servers-clients-policy
zone-pair security clients-servers source clients destination servers
  service-policy type inspect clients-servers-policy
zone-pair security private-dmz source private destination dmz
  service-policy type inspect private-dmz-policy
zone-pair security internet-dmz source internet destination dmz
  service-policy type inspect internet-dmz-policy
!
bridge irb
!
interface FastEthernet0
 ip address 172.16.1.88 255.255.255.0
 zone-member internet
!
```

```
interface FastEthernet1
 ip address 172.16.2.1 255.255.255.0
 zone-member dmz
!
interface FastEthernet2
 switchport access vlan 2
!
interface FastEthernet3
 switchport access vlan 2
!
interface FastEthernet4
 switchport access vlan 1
!
interface FastEthernet5
 switchport access vlan 1
!
interface FastEthernet6
 switchport access vlan 1
!
interface FastEthernet7
 switchport access vlan 1
!
interface Vlan1
 no ip address
 zone-member clients
 bridge-group 1
!
interface Vlan2
 no ip address
 zone-member servers
 bridge-group 1
!
interface BVI1
 ip address 192.168.1.254 255.255.255.0
 zone-member private
!
ip classless
ip route 0.0.0.0 0.0.0.0 172.16.1.1
!
access-list 110 permit ip any host 172.16.2.2 access-list 111 permit ip any host 172.16.2.3 !
bridge 1 protocol ieee bridge 1 route ip ! End
```

Приложение B: Основная конфигурация межсетевого экрана зональной политики для двух зон

В этом примере приведена простая конфигурация в качестве основы для тестирования функций, относящихся к разряду усовершенствований ZFW в Cisco IOS. Она представляет собой модель для двух зон, настроенных с помощью маршрутизатора 1811. Частная зона соответствует фиксированным портам маршрутизатора, чтобы подключить все узлы портов коммутатора к VLAN 1. Публичной зоне сопоставлен FastEthernet 0.




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

Дополнительные сведения

- Cisco Systems – техническая поддержка и документация