Networking
For everyone

Protocol Independent Multicast

В этом модуле

- PIM Dense Mode
- PIM Sparse Mode

PIM Dense Mode

Основы PIM DM

- Протокол считает, что (по-умолчанию) заинтересованные получатели есть везде
- Маршрутизатор должен передавать многоадресный трафик через все свои интерфейсы
  - если потом оказывается, что где-то он не нужен, то эта ветка «отрезается»

Дерево кратчайшего пути

- После «обрезания» ненужных интерфейсов остаётся дерево, вдоль которого передаётся многоадресный трафик
- Это дерево называется SPT — Shortest Path Tree.
- Дерево описывает путь между источником и всеми сетями, которые нуждаются в получении трафика от этого источника
- PIM-DM создает новое SPT в случае когда источник начинает слать пакеты на новый многоадресный адрес

Reverse Path Forwarding

- Если передавать трафик во все интерфейсы, петля неизбежна
- Механизм защиты - *Reverse Path Forwarding* (RPF).
  - *при получении многоадресного пакета первым делом необходимо посмотреть на IP адрес источника. Если **одноадресный** маршрут к этому адресу пролегает через тот же интерфейс, через который пришел ммногодаресный пакет, то пакет флудится дальше. Если же нет, то данные уничтожаются.*

PIM Sparse Mode

- У PIM DM один большой недостаток – много лишнего трафика
- PIM SM использует противоположную парадигму – передавать многоадресный трафик только туда, где к нему явно проявили интерес
- Необходим механизм, позволяющий сообщить данную информацию
  - PIM Join
    - почти как IGMP Join/Report

PIM соседство

- Чтобы передавать информацию, необходимо соседство
- Поиск соседей – через PIM Hello на адрес 224.0.0.13
- Существует PIM v1 и v2
  - v2 использует свой собственный IP protocol (103)
  - v1 использует для транспорта IGMP
  - функциональность идентична

Точка рандеву

- Основная проблема
  - источник не знает адресов получателей
  - получатели не знают адрес сервера
- Идея
  - выстроить два независимых дерева
  - совместить их вместе в какой-либо точке сети
- Такая точка сети называется точкой рандеву (Randezvous Point, RP)

Построение деревьев

- Дерево кратчайшего пути
  - Shortest Path Tree (SPT)
- Общее дерево
  - Shared Tree

Построение общего дерева

- Основывается на трёх вводных данных
  - известен адрес точки рандеву
    - любым способом
  - известно местоположение точки рандеву
    - адрес присутствует в RIB
  - в сети есть хотя бы один заинтересованный получатель многоадресного трафика

Построение общего дерева

- Получатель отправляет IGMP Report
- Маршрутизатор (Last Hop Router, LHR) добавляет интерфейс, на котором был получен Report, в Outgoing Interface List (OIL)
- Формируется запись (*, G)
- LHR делает RPF проверку для адреса точки рандеву
- LHR отправляет PIM Join через интерфейс, прошедший RPF проверку
  - Join на группу (*, G)

Построение общего дерева

- Вышестоящий сосед получает PIM Join и формирует (*, G) маршрут
- Интерфейс, через который был получен PIM Join, добавляется в OIL
- PIM Join отправляется дальше в сторону точки рандеву

Построение кратчайшего дерева

- Сервер начинает рассылку пакетов
- Задача FHR (First Hop Router) – доставить пакеты до точки рандеву
- FHR инкапсулирует пакеты от сервера внутрь одноадресного PIM Register, которое отправляется к точке рандеву
- Пакет доставляется до точки рандеву посредством одноадресной маршрутизации

Построение кратчайшего дерева

- Точка рандеву обрабатывает PIM Register и создаёт маршрут (S, G)
- Полезная нагрузка из Register (многоадресный пакет) отправляется по ранее сформированному общему дереву согласно OIL
- Технически задача решена
  - однако, есть некоторые **НО …**

Построение кратчайшего дерева

- Точка рандеву отправляет PIM Join в сторону источника
  - Join (S, G)
- PIM Join проходит путь от точки рандеву до FHR
- FHR обновляет OIL и начинает передавать многоадресный трафик в чистом виде
  - Register продолжают пересылаться

Построение кратчайшего дерева

- Точка рандеву получает два одинаковых многоадресных пакета
- Создаётся сообщение PIM Register Stop и отправляется в сторону FHR
- FHR перестаёт пересылать инкапсулированный многоадресный трафик

Переключение с общего дерева на кротчайшее

- При описанном подходе трафик всегда идёт через точку рандеву, что может быть неоптимально для многих получателей
- При получении первого многоадресного пакета, LHR будет знать IP адрес источника
  - можно попробовать построить нового дерево
  - и не забыть отрезать старое

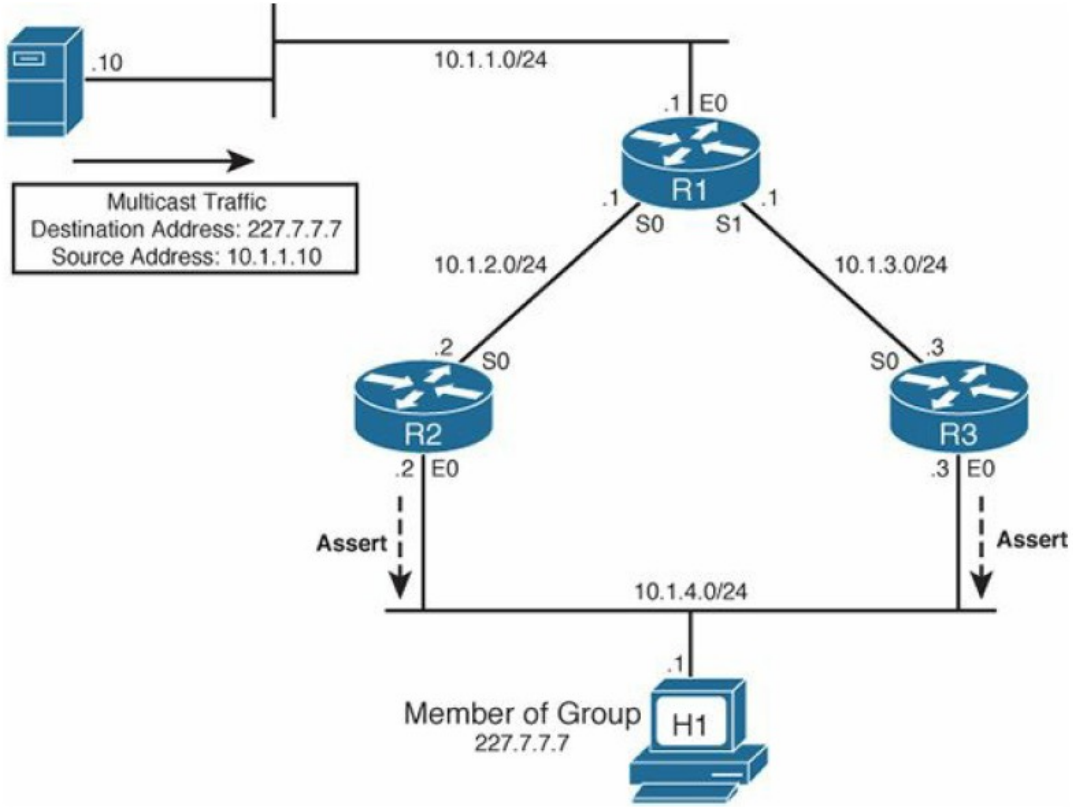
Механизмы оптимизации

Выделенный маршрутизатор

- Designated Router (DR) выбирается на каждом сегменте
- Выбор происходит посредством PIM Hello
  - поле Priority
  - если приоритеты равны, выигрывает наибольший IP адрес
- DR отвечает за отправку
  - PIM Join/Prune
  - PIM Register

Выделенный передатчик

- Designated Forwarder отвечает за передачу многоадресного трафика в LAN сегмент
- Выбор посредством PIM Assert

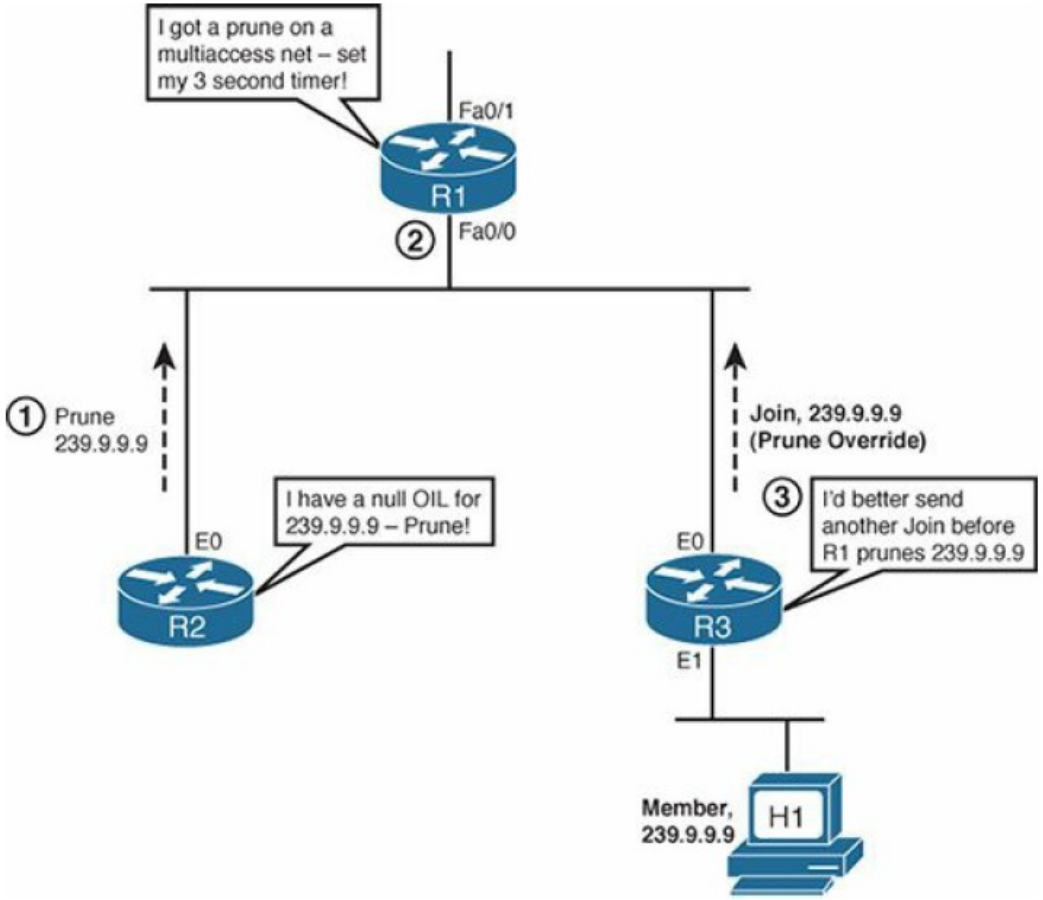


Assert

- Если два маршрутизатора обнаруживают дублирование трафика, высылается Assert
- Критерии определения победителя
  - низкая Administrative Distance
  - низкая метрика
  - высший IP адрес

Prune Override

- При получении PIM Prune выставляется таймер в 3 секунды
- Он позволяет среагировать маршрутизаторам, находящимся в этом же сегменте



Поиск точки рандеву

Методы поиска

- Статическая конфигурация
- Динамические методы поиска
  - Auto-RP
  - BootStrap Router (BSR)

Networking
For everyone

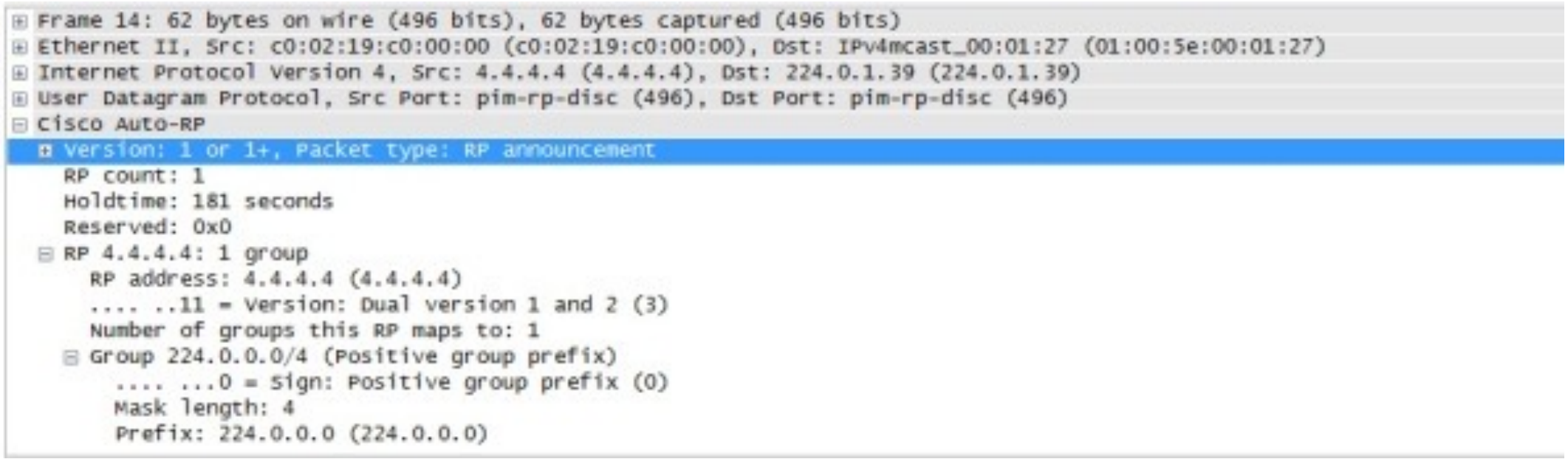
Auto-RP

Auto-RP

- Исторически первый автоматический метод поиска точки рандеву
- Каждый маршрутизатор берёт на себя одну из трёх ролей
  - кандидат на роль точки рандеву (RP Candidate)
  - агент (Mapping Agent, MA)
  - обычный маршрутизатор
- Используются зарезервированные адреса
  - 224.0.1.39 и 224.0.1.40

Кандидат на роль точки рандеву

- Роль настраивается вручную
- Кандидат отправляет сообщение RP-Announce, которое должно быть обработано агентом MA
  - 224.0.1.39



Агент

- Прослушивает сообщения RP-Announce
- Если RP-Announce приходят от нескольких кандидатов (для одного GDA), выбирается лучший из них
- Отправляет сообщение RP-Discovery
  - внутри передаётся информация о выбранном адресе точки рандеву

```
Cisco Auto-RP
 Version: 1 or 1+, Packet type: RP mapping
   0001 .... = Protocol version: 1 or 1+ (1)
   .... 0010 = Packet type: RP mapping (2)
  RP count: 1
  Holdtime: 46 seconds
  Reserved: 0x0
 RP 2.4.2.2: 1 group
   RP address: 2.4.2.2 (2.4.2.2)
   .... ..11 = Version: Dual version 1 and 2 (3)
   Number of groups this RP maps to: 1
  Group 224.0.0.0/4 (Positive group prefix)
    .... ...0 = Sign: Positive group prefix (0)
    Mask length: 4
    Prefix: 224.0.0.0 (224.0.0.0)
```

Проблема курицы и яйца

- PIM Sparse Mode требует наличия точки рандеву для своей работы
- Как быть с сообщениями RP Announce и RP Discovery?
  - для этих сообщений используется PIM Dense Mode
  - все интерфейсы настраиваются в режим sparse-dense-mode
    - или включается *autorp listener*

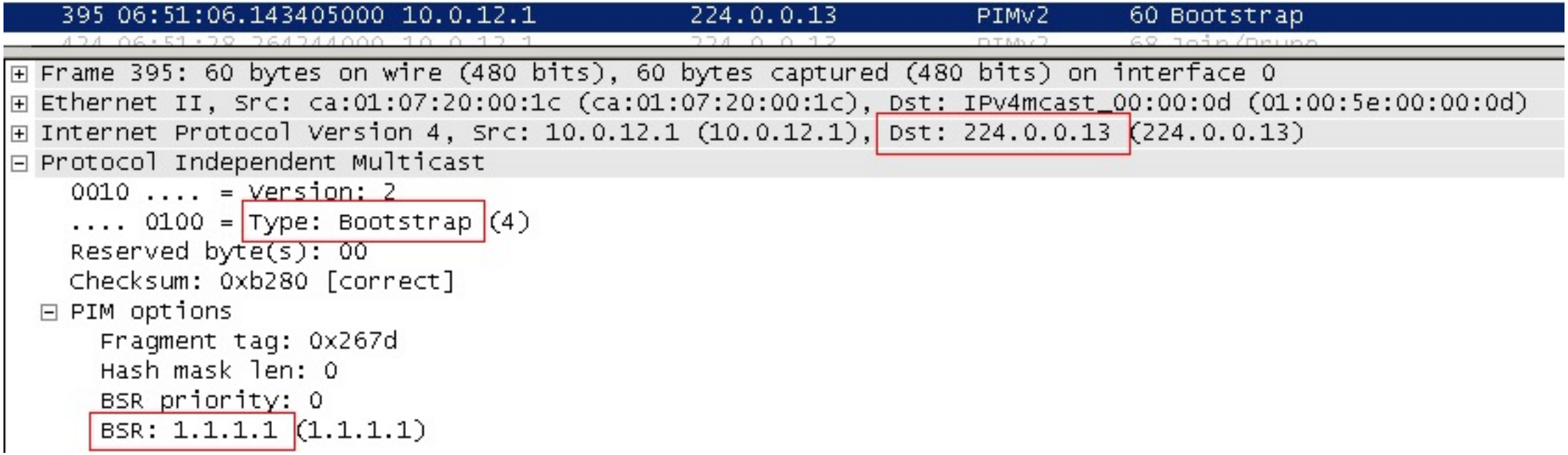
BSR

Отличия от Auto-RP

- Стандартизированный протокол
  - определён в RFC 5059
- Работает только с PIM v2
  - Auto-RP подходит для v1/v2
- Все интерфейсы могут работать в режиме *sparse-mode*
  - Auto-RP требует *sparse-dense-mode*
- Для рассылки используется адрес 224.0.0.13

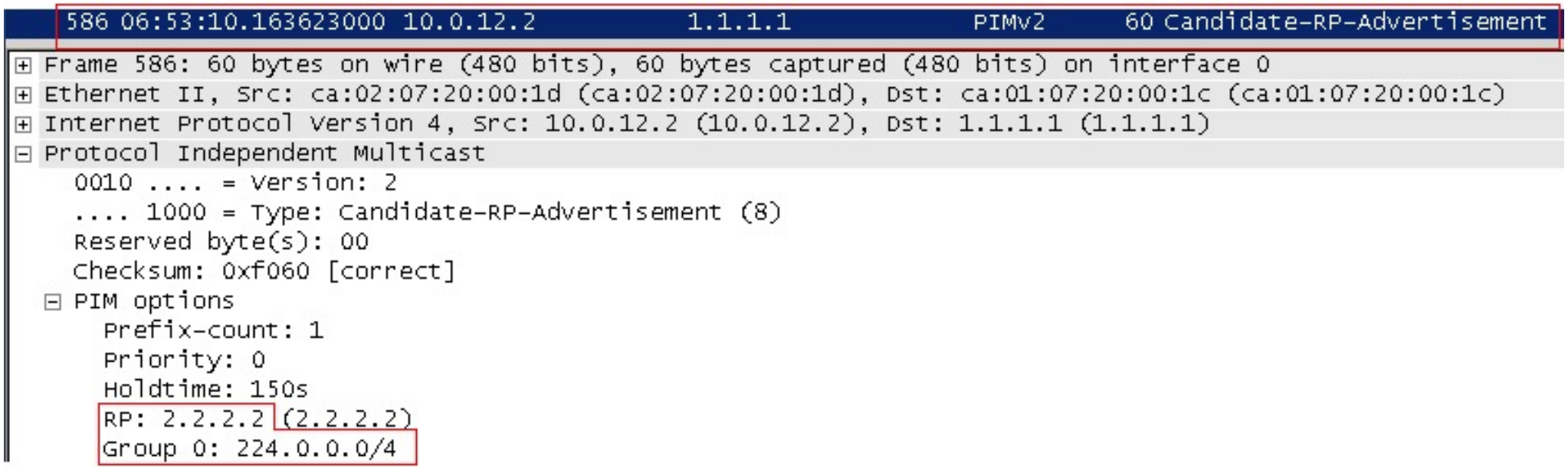
Кандидат на роль BSR

- Из всех BSR кандидатов выбирается один
- Каждый кандидат отправляет Bootstrap Message (BSM)
- Сообщения распространяются по всей сети



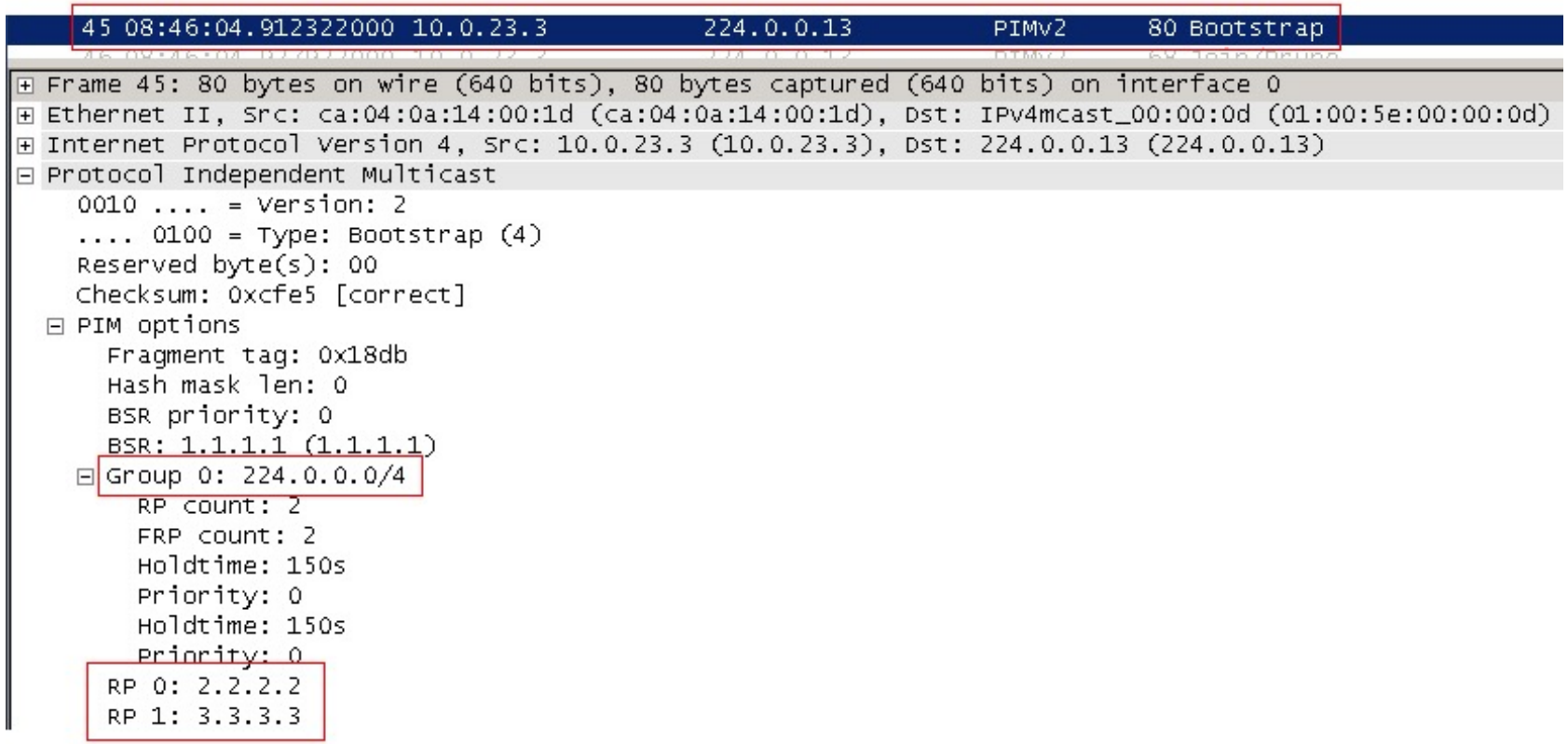
Кандидат на роль точки рандеву

- Каждый кандидат на роль точки рандеву отправляет Candidate RP Advertisement на адрес BSR



Распространение информации о точке рандеву

- Информация о всех известных точках помещается внутрь BSM и распространяется по сети
- Каждый маршрутизатор независимо решает какую точку рандеву использовать



Отказоустойчивость точки рандеву

Возможные способы

- У динамических протоколов свои методы реализации отказоустойчивости для точки рандеву
  - в Auto-RP за выбор RP отвечает MA
  - в BSR за выбор RP отвечает каждый маршрутизатор
- Есть способ, который напрямую не связан ни с Auto-RP ни с BSR
  - Anycast RP

Что такое Anycast

- Под понятием Anycast обычно понимают настройку, при которой на 2-х или более маршрутизаторах настроен одинаковый IP адрес
- Выбор маршрутизатора основывается на IGP/BGPметриках
- В зависимости от топологии это может привести к тому, что кратчайшее и общее деревья будут строиться от разных устройств
  - что приведёт к невозможности передачи многоадресного трафика
- Необходима синхронизация и/или сигнализация между Anycast точками рандеву

MSDP

- Multicast Source Discovery Protocol
- Позволяет передавать информацию об источниках между всеми Anycast точками рандеву
  - использует сообщения Source Active (SA)
- Чаще всего используется для организации многоадресной маршрутизации между разными автономными системами
  - значит никакого «растянутого» IGP
    - что делать с RPF?

Multicast BGP (MBGP)

- Дополнительная BGP таблица (AFI/SAFI)
  - AFI = 1 (IPv4) или 2 (IPv6)
    - SAFI = 2 (только многоадресный NLRI)
    - SAFI = 3 (многоадресный и одноадресный NLRI)
- Содержит в себе *одноадресные* префиксы для RPF

Приоритеты RPF

- Возможна ситуация когда IGP и MBGP указывают в разных направлениях
- Приоритетность
  - статический многоадресный маршрут
  - таблица MBGP
  - одноадресная таблица маршрутизации

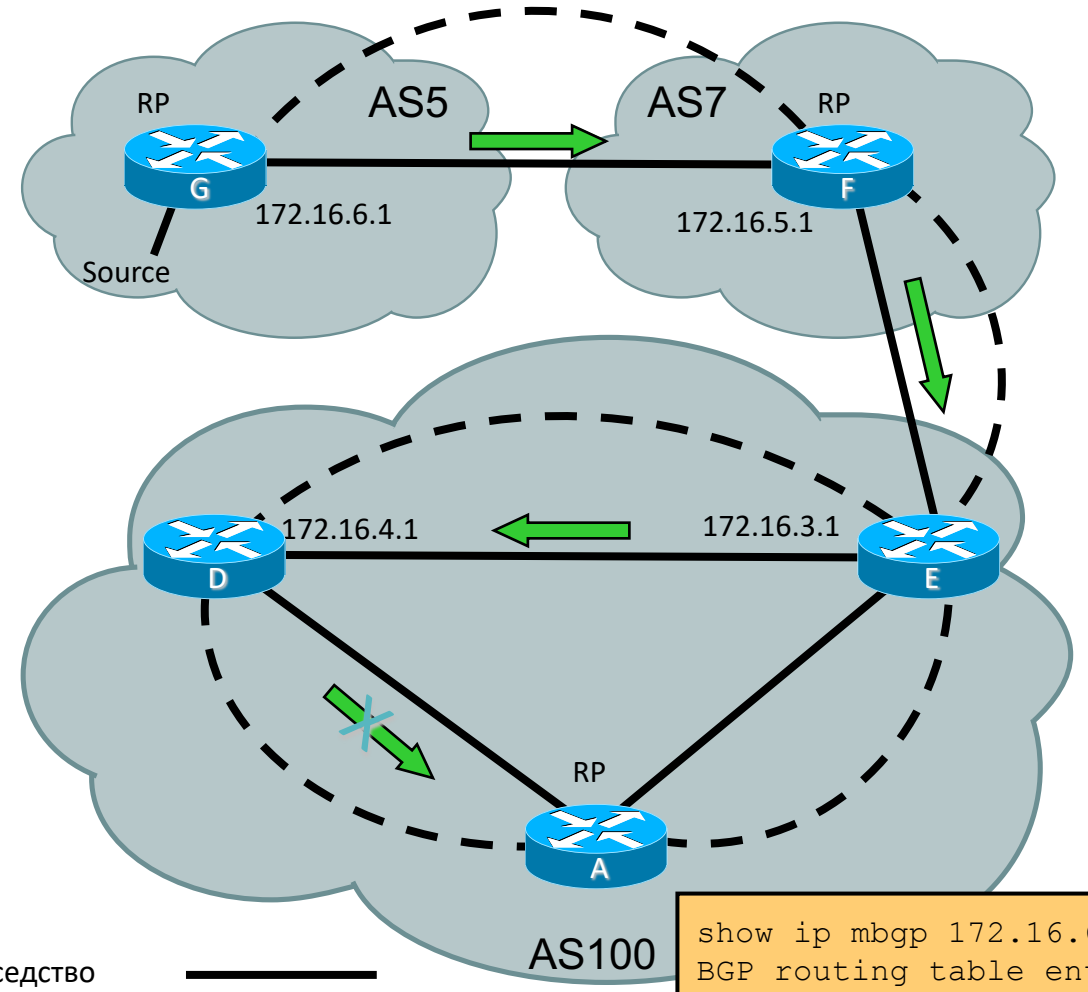
Правила MSDP

- RPF правила различаются в зависимости от типа MSDP соседа
  - MSDP сосед = i(m) BGP соседу
  - MSDP сосед = e(m) BGP соседу
  - MSDP сосед != (m) BGP соседу

MSDP сосед = i(m) BGP соседу

- Ищется лучший путь до точки рандеву
  - сначала поиск через MRIB, затем через URIB
- Запомнить BGP соседа, который отдаёт лучший путь
  - BGP сосед != BGP Next-Hop
- Адрес MSDP соседа и BGP соседа должны совпадать
  - если нет, то RPF не пройден

MSDP сосед = i(m) BGP соседу



BGP соседство

MSDP соседство

SA

i(m)BGP адрес = 172.16.3.1
(отдаёт best-path к RP)

MSDP адрес = 172.16.4.1

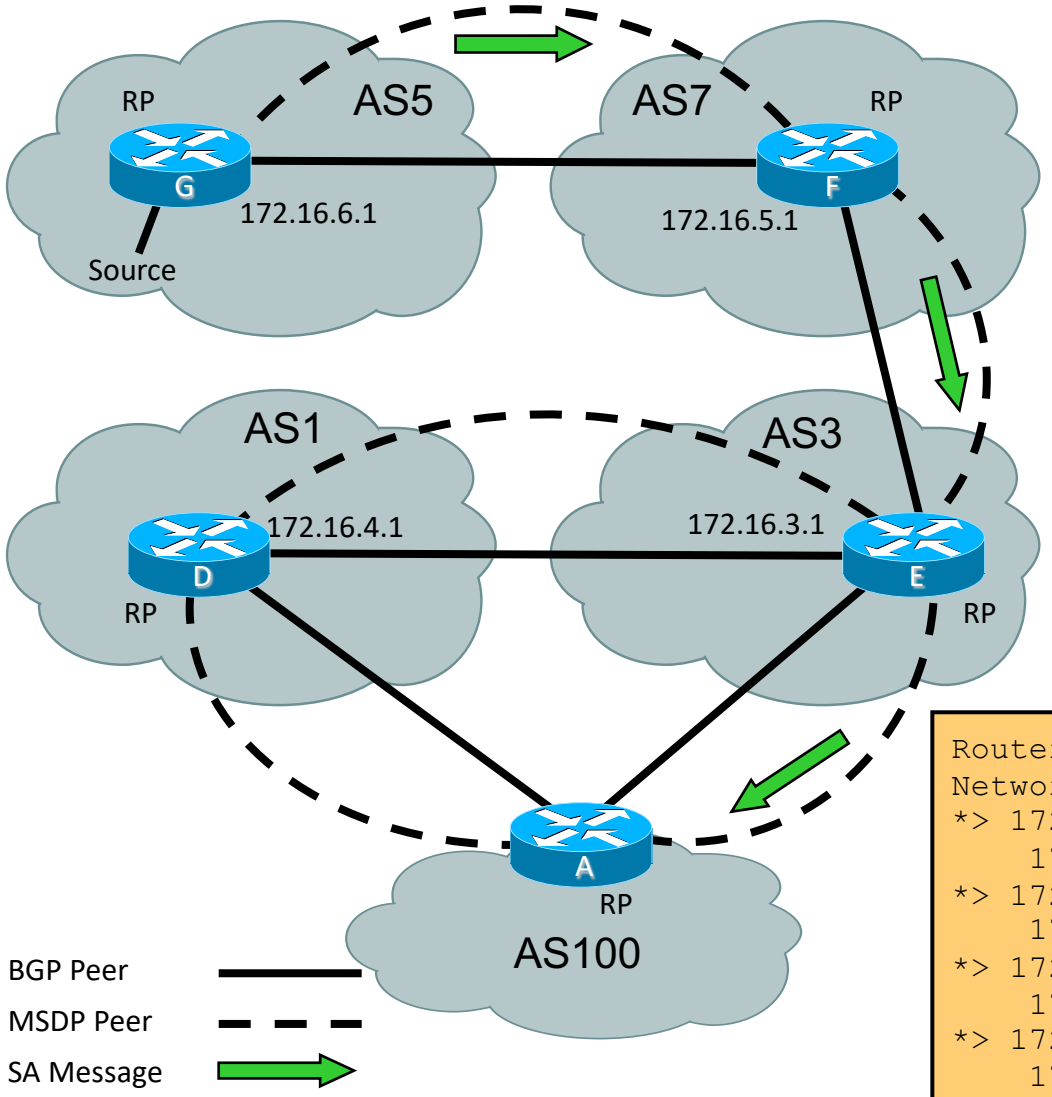
MSDP адрес != i(m)BGP адрес

SA RPF проверка не пройдена

```
show ip mbgp 172.16.6.1
BGP routing table entry for 172.16.6.0/24, version 8745118
Paths: (1 available, best #1)
7 5, (received & used)
    172.16.5.1 (metric 68096) from 172.16.3.1 (172.16.3.1)
```

MSDP сосед = e(m) BGP соседу

- Ищется лучший путь до точки рандеву
  - сначала поиск через MRIB, затем через URIB
- Запомнить номер первой AS в атрибуте AS_PATH
  - если указанный номер AS = номер AS MSDP соседа, то RPF пройдено успешно



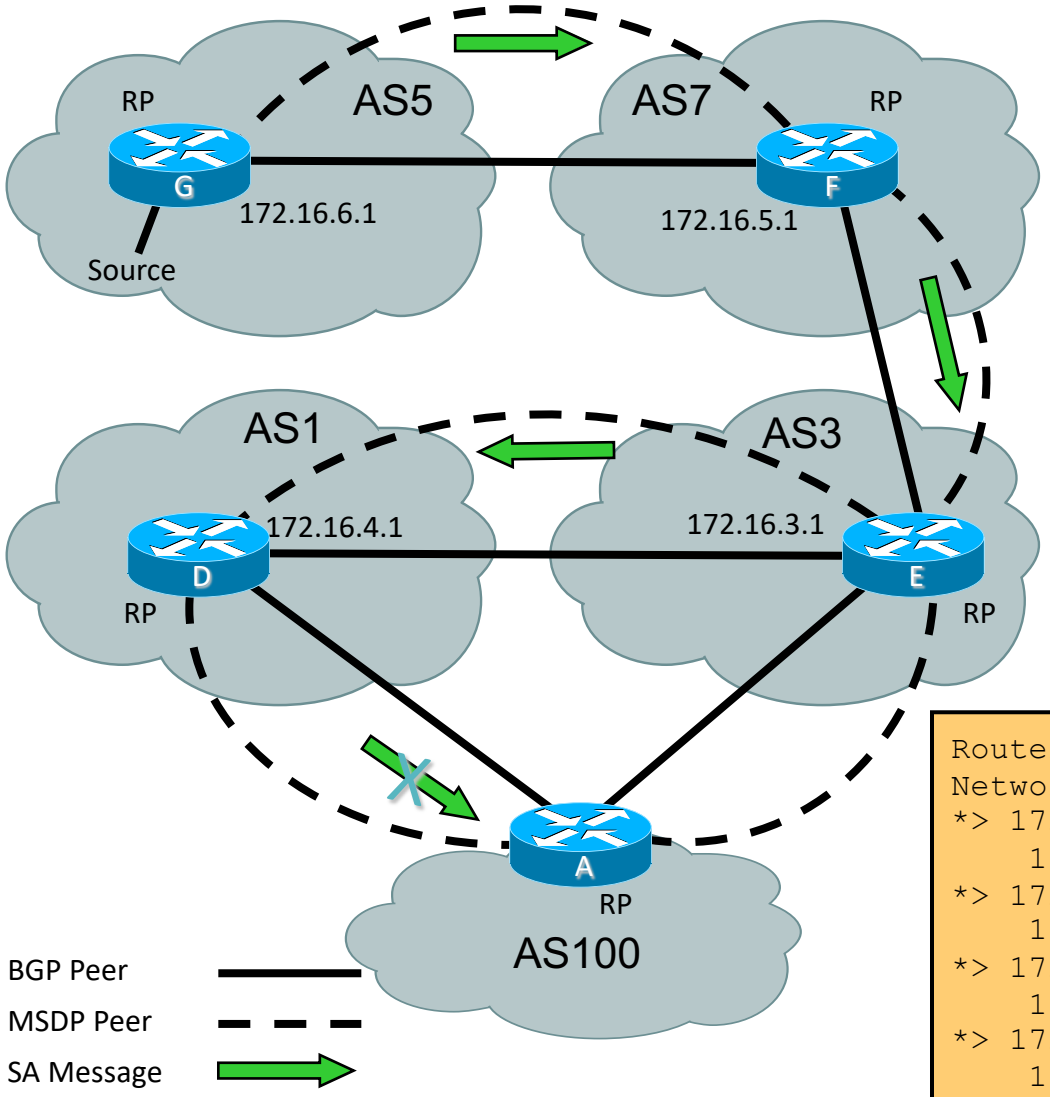
First-AS in best-path to RP = 3

AS of MSDP Peer = 3

First-AS in best-path to RP = AS of e(m)BGP Peer

SA RPF Check Succeeds

```
Router A's BGP Table
Network              Next Hop       Path
*> 172.16.3.0/24    172.16.3.1      3 i
    172.16.3.0/24    172.16.4.1     1 3 i
*> 172.16.4.0/24    172.16.4.1      1 i
    172.16.4.0/24    172.16.3.1     3 1 i
*> 172.16.5.0/24    172.16.4.1      3 7 i
    172.16.5.0/24    172.16.3.1     1 3 7 i
*> 172.16.6.0/24    172.16.3.1      3 7 5 i
    172.16.6.0/24    172.16.4.1     1 3 7 5 i
```



First-AS in best-path to RP = 3

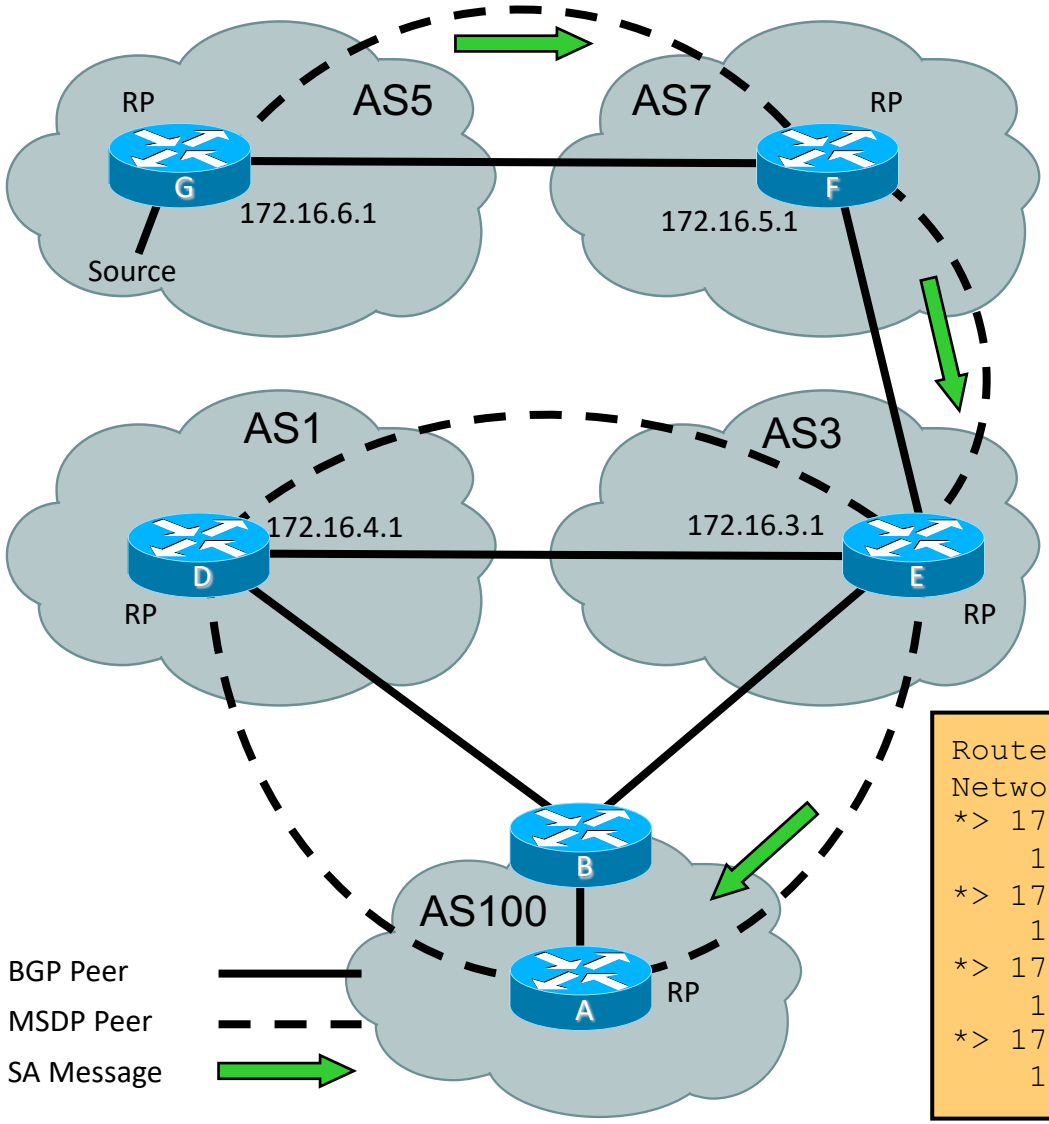
AS of MSDP Peer = 1

First-AS in best-path to RP != AS of e(m)BGP Peer

SA RPF Check Fails!

```
Router A's BGP Table
Network              Next Hop       Path
*> 172.16.3.0/24     172.16.3.1     3 i
    172.16.3.0/24     172.16.4.1     1 3 i
*> 172.16.4.0/24     172.16.4.1     1 i
    172.16.4.0/24     172.16.3.1     3 1 i
*> 172.16.5.0/24     172.16.3.1     3 7 i
    172.16.5.0/24     172.16.4.1     1 3 7 i
*> 172.16.6.0/24     172.16.3.1     3 7 5 i
    172.16.6.0/24     172.16.4.1     1 3 7 5 i
```

MSDP сосед != (m) BGP соседу

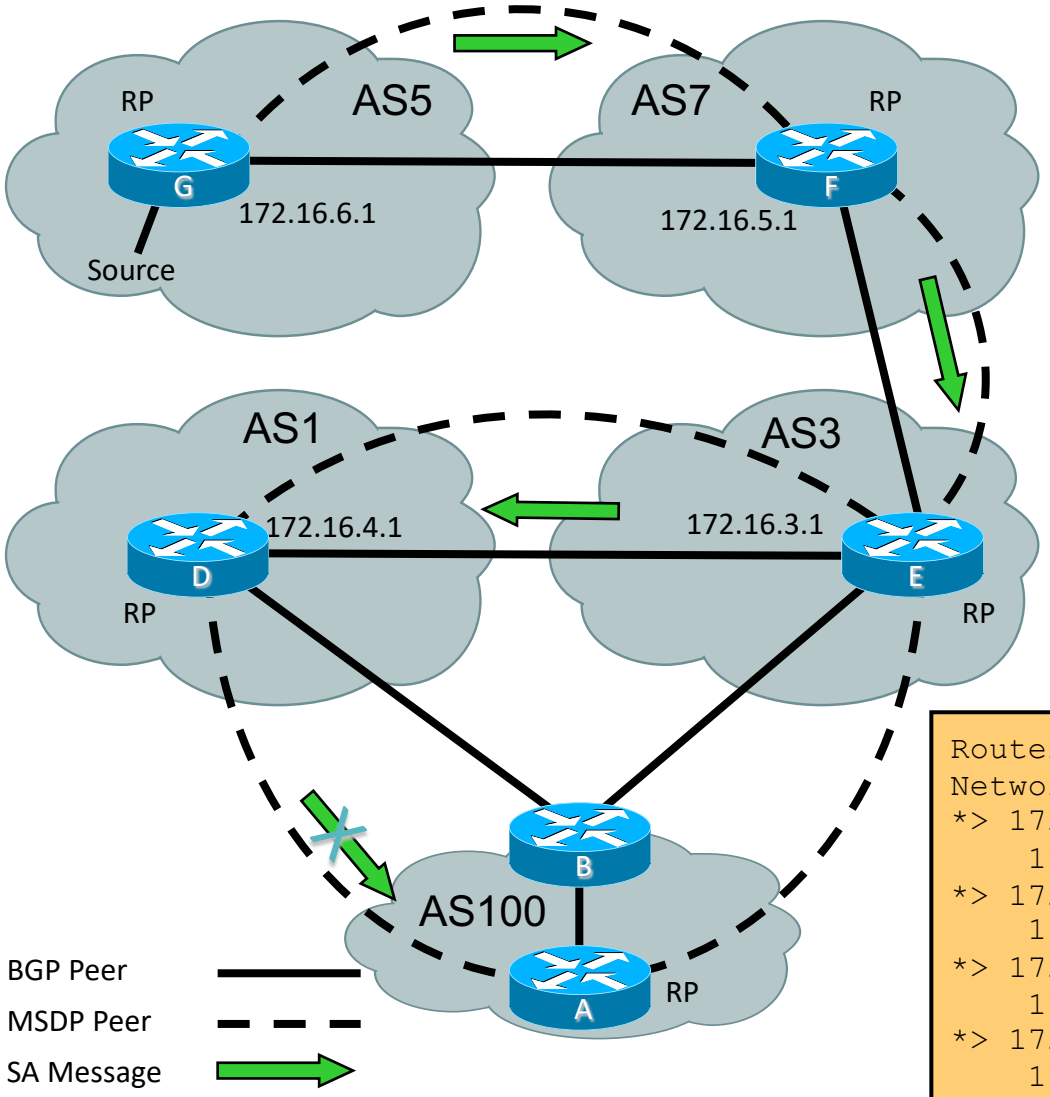


First-AS in best-path to RP = 3
AS of MSDP Peer = 3

First-AS in best-path to RP = AS of MSDP Peer

SA RPF Check Succeeds

```
Router A's BGP Table
Network              Next Hop       Path
*> 172.16.3.0/24    172.16.3.1     3 i
    172.16.3.0/24    172.16.4.1     1 3 i
*> 172.16.4.0/24    172.16.4.1     1 i
    172.16.4.0/24    172.16.3.1     3 1 i
*> 172.16.5.0/24    172.16.4.1     3 7 i
    172.16.5.0/24    172.16.3.1     1 3 7 i
*> 172.16.6.0/24    172.16.3.1     3 7 5 i
    172.16.6.0/24    172.16.4.1     1 3 7 5 i
```



First-AS in best-path to RP = 3
AS of MSDP Peer = 1

First-AS in best-path to RP != AS of MSDP Peer

SA RPF Check Fails

```
Router A's BGP Table
Network              Next Hop       Path
*> 172.16.3.0/24    172.16.3.1     3 i
    172.16.3.0/24    172.16.4.1     1 3 i
*> 172.16.4.0/24    172.16.4.1     1 i
    172.16.4.0/24    172.16.3.1     3 1 i
*> 172.16.5.0/24    172.16.4.1     3 7 i
    172.16.5.0/24    172.16.3.1     1 3 7 i
*> 172.16.6.0/24    172.16.3.1     3 7 5 i
    172.16.6.0/24    172.16.4.1     1 3 7 5 i
```

Рекомендованный SA фильтр

```
! domain-local applications
access-list 111 deny   ip any host 224.0.2.2     !
access-list 111 deny   ip any host 224.0.1.3     ! Rwhod
access-list 111 deny   ip any host 224.0.1.24    ! Microsoft-ds
access-list 111 deny   ip any host 224.0.1.22    ! SVRLOC
access-list 111 deny   ip any host 224.0.1.2     ! SGI-Dogfight
access-list 111 deny   ip any host 224.0.1.35    ! SVRLOC-DA
access-list 111 deny   ip any host 224.0.1.60    ! hp-device-disc
!-- auto-rp groups
access-list 111 deny   ip any host 224.0.1.39
access-list 111 deny   ip any host 224.0.1.40
!-- scoped groups
access-list 111 deny   ip any 239.0.0.0 0.255.255.255
!-- loopback, private addresses (RFC 1918)
access-list 111 deny   ip 10.0.0.0 0.255.255.255 any
access-list 111 deny   ip 127.0.0.0 0.255.255.255 any
access-list 111 deny   ip 172.16.0.0 0.15.255.255 any
access-list 111 deny   ip 192.168.0.0 0.0.255.255 any
access-list 111 permit ip any any
!-- Default SSM-range. Do not do MSDP in this range
access-list 111 deny   ip any 232.0.0.0 0.255.255.255
access-list 111 permit ip any any
```

Networking For everyone