Segment Routing

Матвей Александров
системный инженер

matvey@juniper.net

Содержание

- Концепция. Типы сегментов.
- Пример: миграция из LDP на SR
- Пример: SR-TE из NorthStar
- Пример: Anycast-SID
- Пример: (R)LFA через SR
- Пример: сопряжение SR и LDP
- Разное

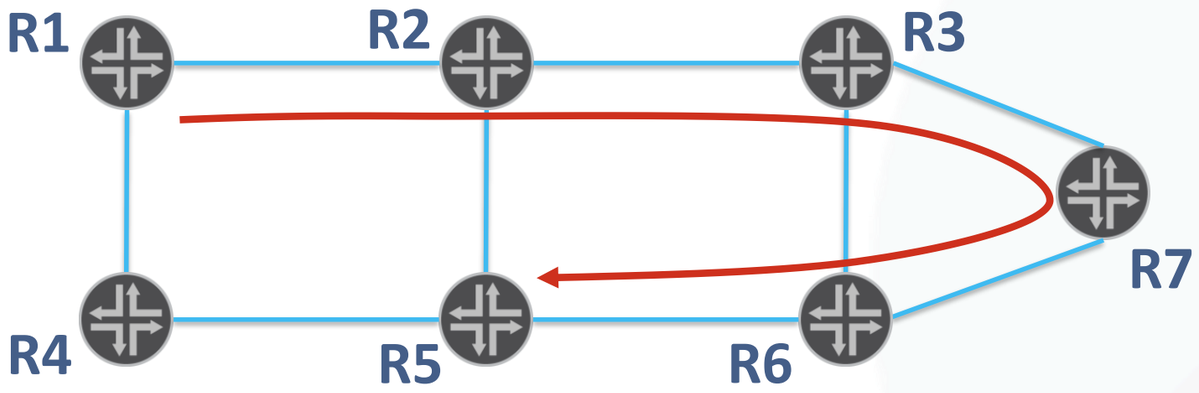
Концепция. Типы сегментов.

SR / SPRING

- Source Packet Routing in Networking
- Segment Routing (SR)
- Новая технология для построения MPLS транспорта
  - LDP
  - RSVP
  - BGP-LU
- Метки распространяются через IGP протокол
  - новые *TLV* для IS-IS
  - новые *Opaque LSA* для OSPF
  - уменьшаем количество сущностей (=протоколов) в сети

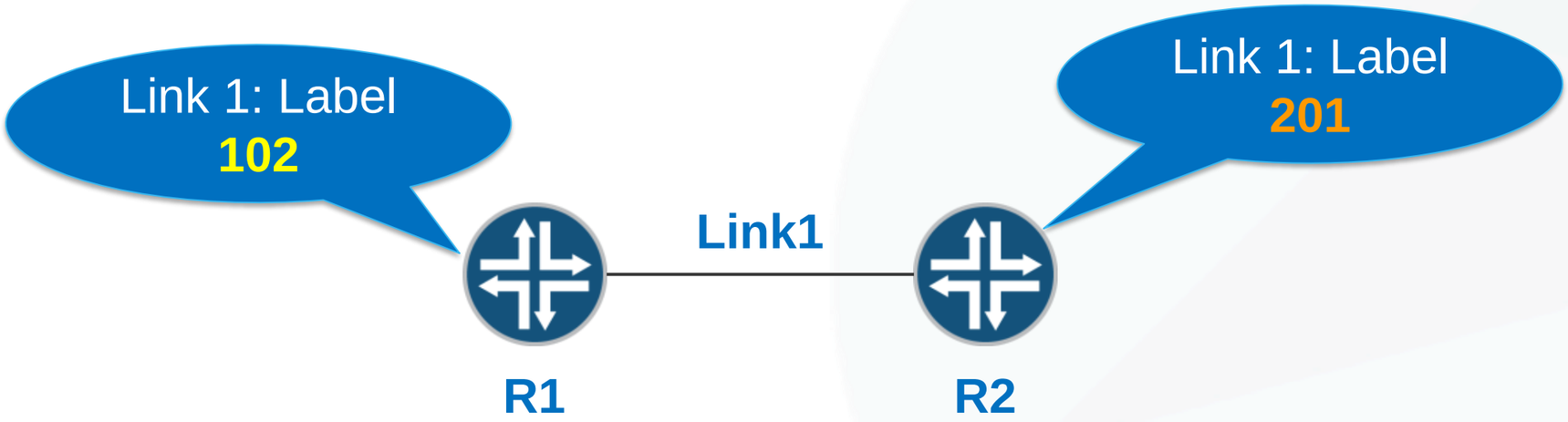
Segment (Сегмент)

- сегмент = инструкция
- существуют разные типы сегментов (=инструкций) – см. далее
- ingress маршрутизатор (=source) программирует путь пакета в сети
- два подхода к организации форвардинга на текущий момент:
  - с помощью стека MPLS меток (SR-MPLS) // в данной презентации только об этом
  - с помощью стека IPv6 адресов (SRv6)



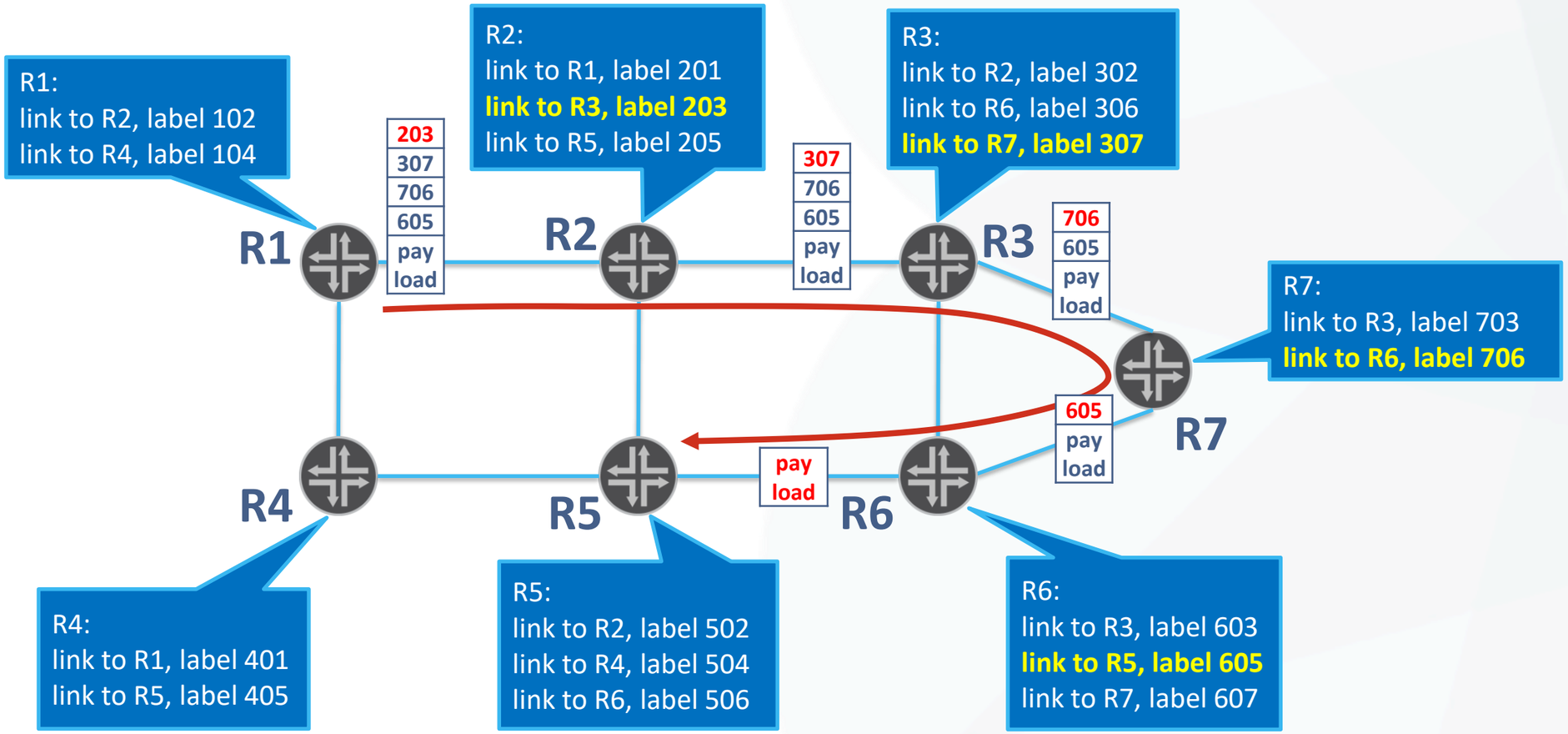
Adj-SID – анонсирование (*Control Plane*)

- Adjacency Segment (Adjacency = Соседство)
- Для каждого линка, который R1 анонсирует в IGP, анонсируется также соответствующая MPLS метка
- Значение этой метки: если R1 получает MPLS пакет с (верхней) меткой X, он снимает эту метку (pop) и пересылает через Link1
- Контекст метки: локальный (свой собственный для каждого Rx)



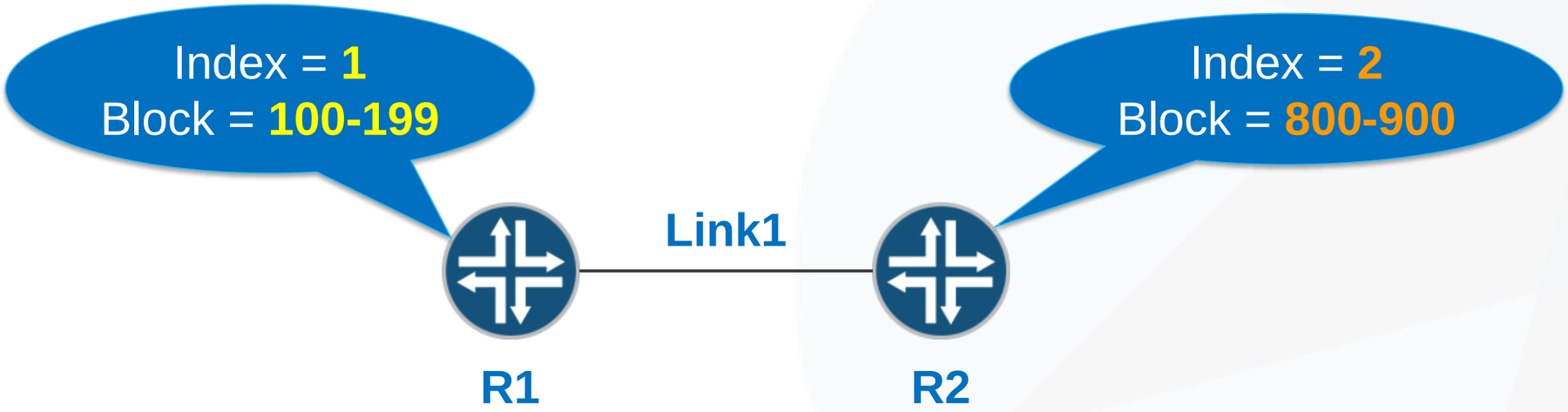
Adj-SID – форвардинг (*Data Plane*)

- Чтобы отправить пакет до R5 по пути (R2,R3,R7,R6), R1 отправляет пакет на R2 со <u>стеком</u> меток (203,307,706,605). *ERO*



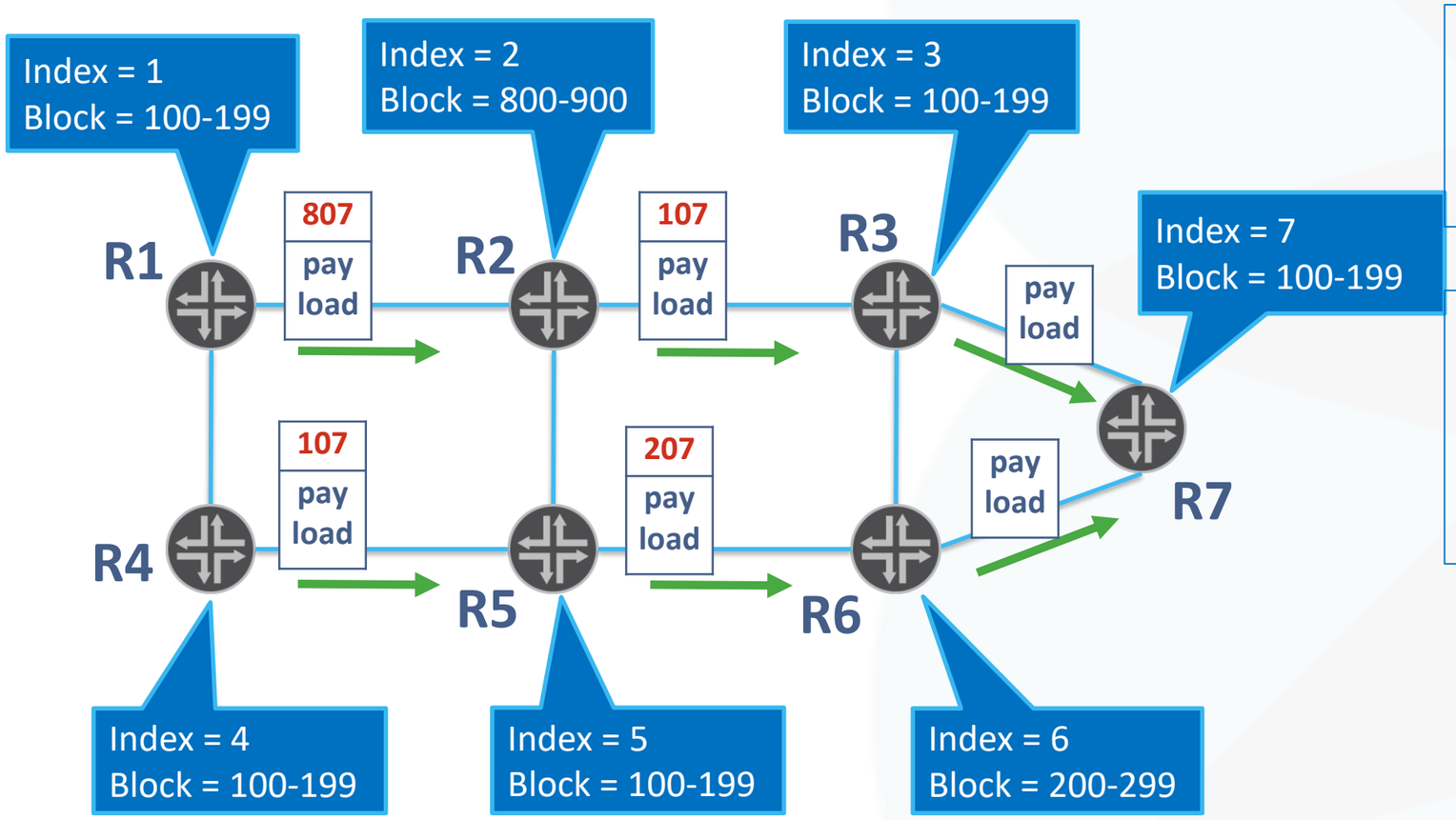
Node-SID – анонсирование (*Control Plane*)

- Node Segment (Node = Узел)
- Маршрутизатор анонсирует индекс + границы блока меток (SRGB)
- Разные блоки меток у разных вендоров
- Контекст для индекса: глобальный (=уникальный в пределах сети)



Node-SID – форвардинг (*Data Plane*)

- Форвардинг происходит по кратчайшему IGP маршруту. *LDP*

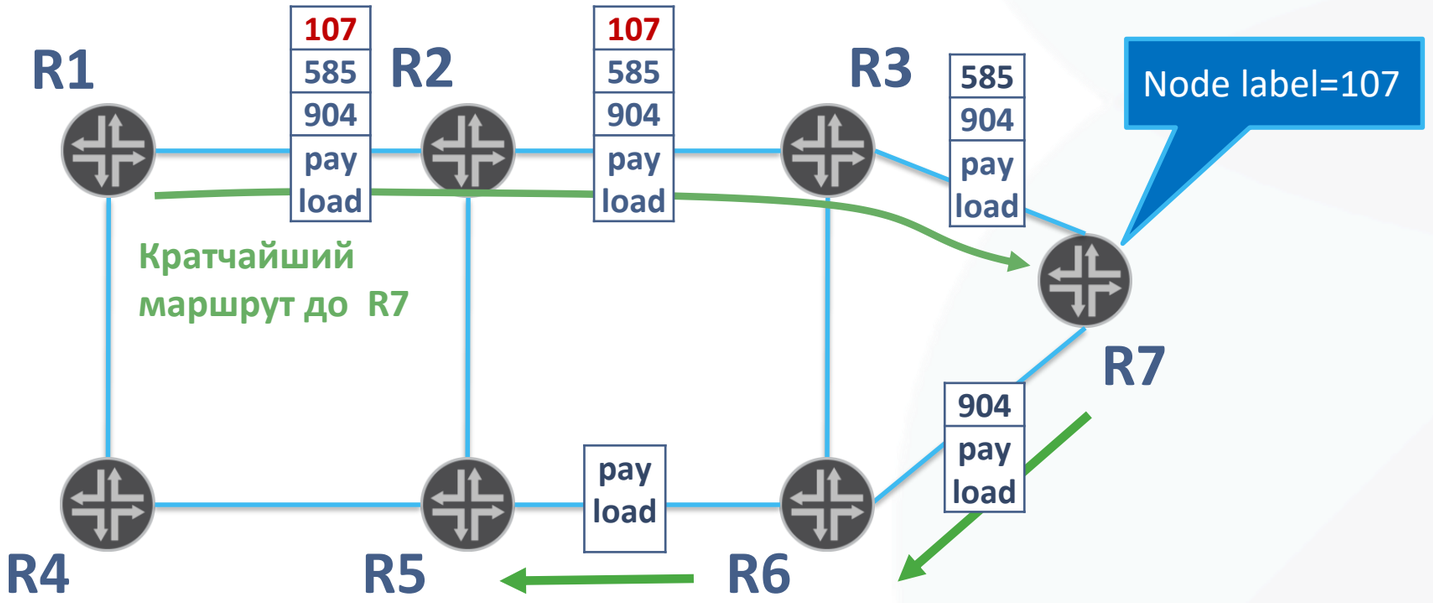


R4:
destination = R7 (index = 7)
next-hop = R5
Tx_label = (R5_block + index) = 100 + 7 = 107

R5:
Index = Rx_label – R5_block = 107 – 100 = 7
Destination = R7
Next-hop = R6
Tx_label = (R6_block + index) = 200 + 7 = 207

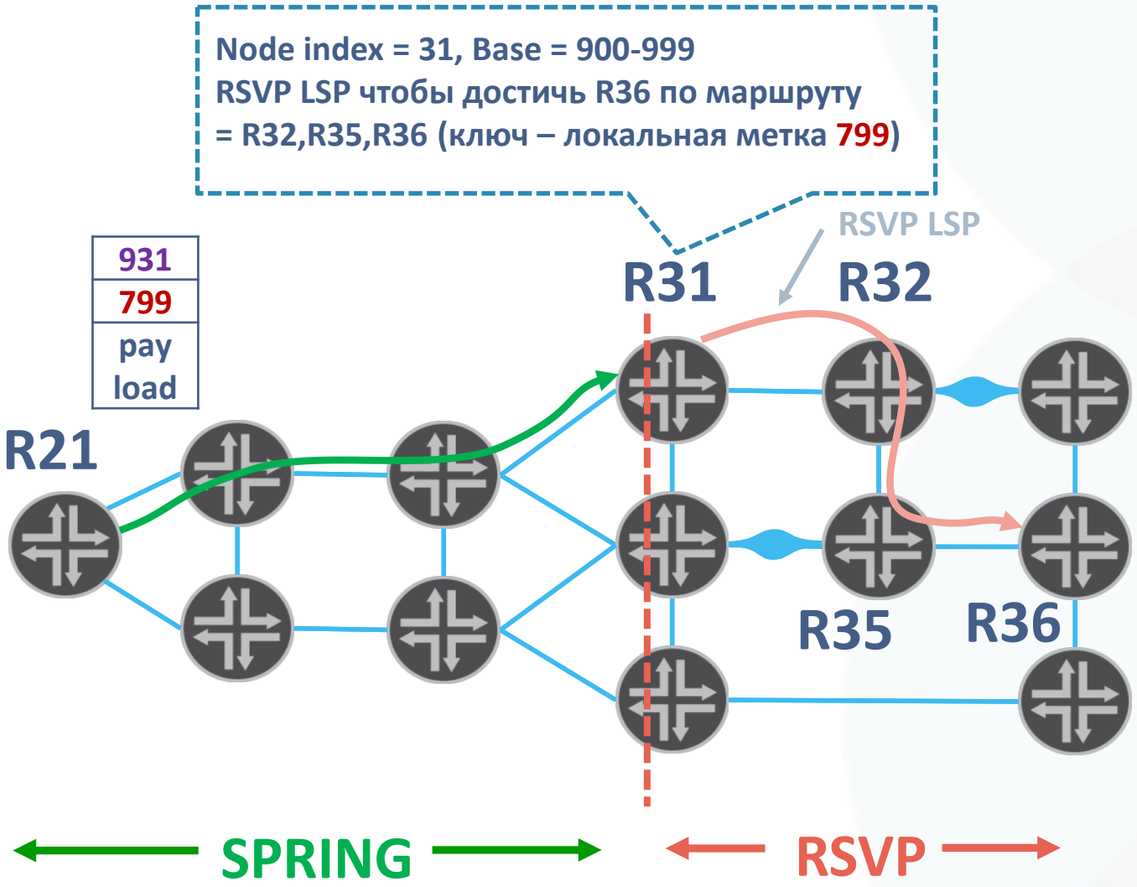
Adj-SID + Node-SID

- Могут использоваться совместно (!)
- Node-SID “107” – инструкция добраться до R7
- Adj-SID “585,904” – после R7 следовать по конкретным линкам



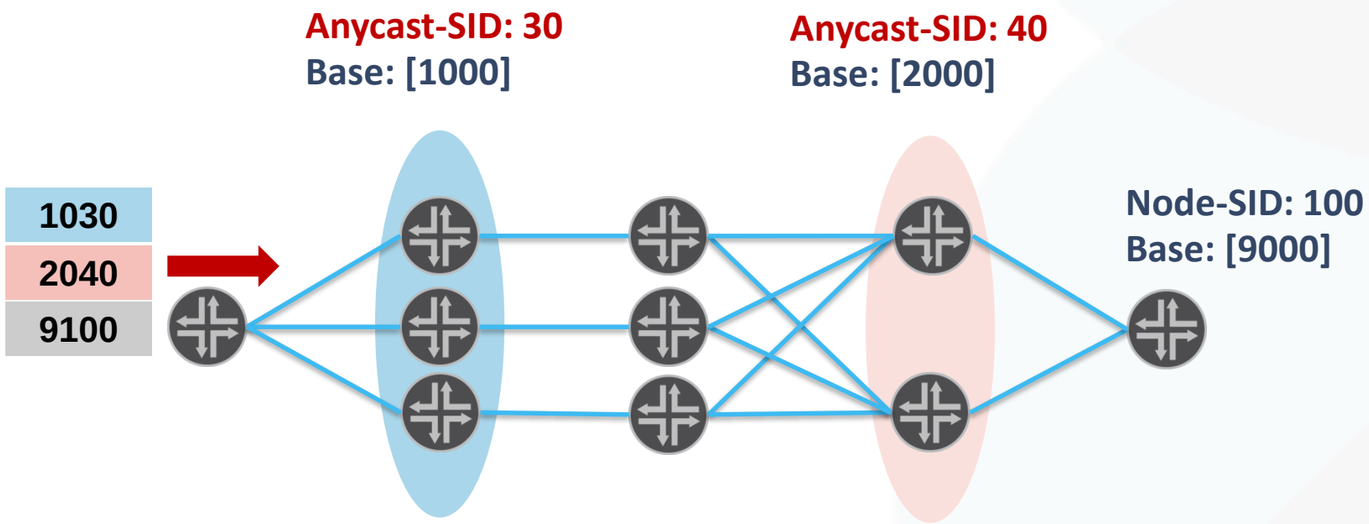
Binding-SID

- Binding Segment (Binding = Привязка)
- например: Binding-SID “799”: связать с конкретным RSVP LSP



Anycast-SID (Any = Любой)

- Определяет набор маршрутизаторов (*~Anycast IPv6 адрес*)
- Форвардинг по кратчайшему маршруту
- Отказоустойчивость, балансировка нагрузки



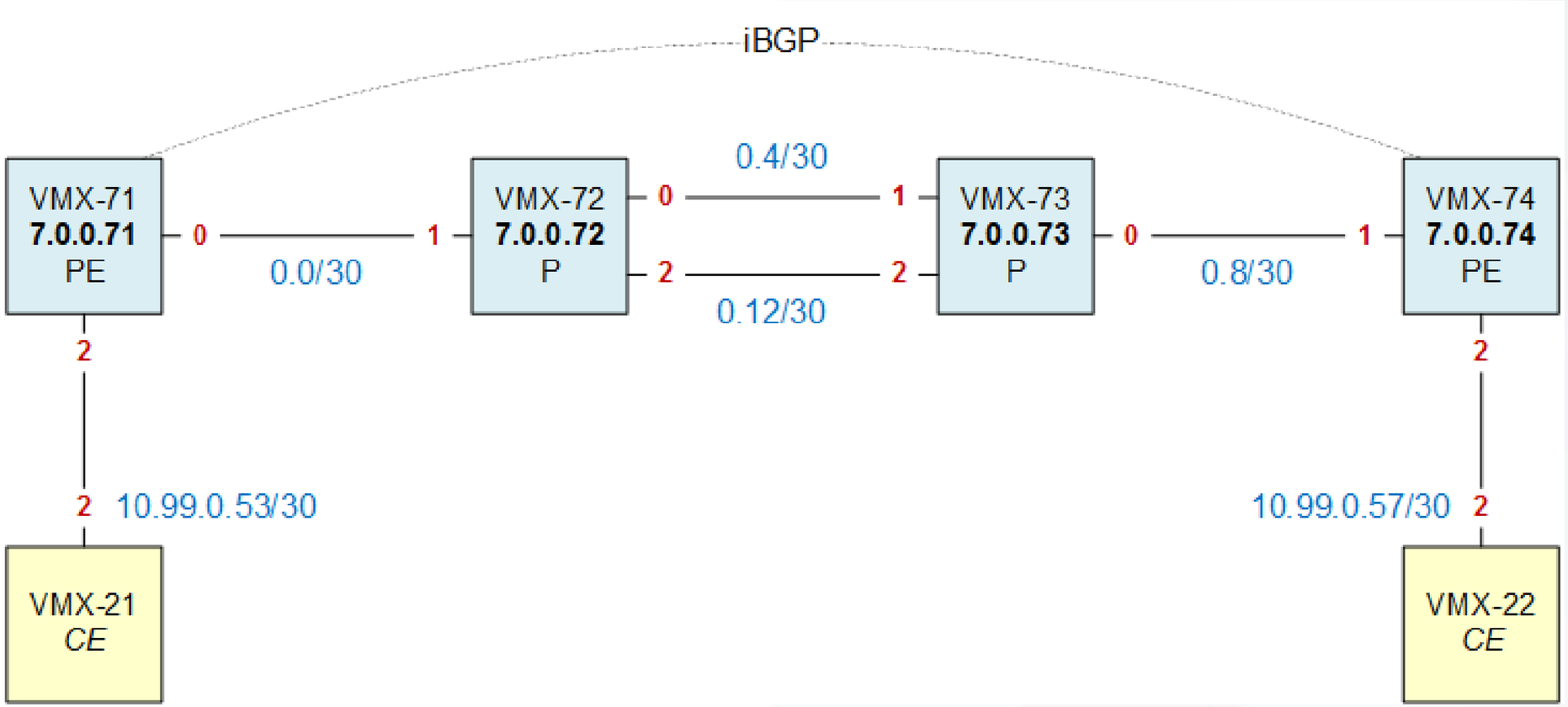
ЗА и ПРОТИВ

- + упрощение Control Plane (=меньше протоколов)
- + не нужна синхронизация между IGP и LDP
- + масштабирование: отсутствуют RSVP сессии на транзите
- + масштабирование: меньше меток/операций в FIB на транзите
- - пропадает статистика на транзите
- - нет резервирования полосы как в RSVP
- хорошо ложится на идеологию SDN: внешний контроллер!

Пример: миграция из LDP на SR

Виртуальная топология

- Классический набор: LDP для транспорта, BGP для сервиса



Связность (1)

```
warrior@vmx-71# run show route protocol bgp

inet.0: 19 destinations, 19 routes (19 active, 0 holddown, 0 hidden)
10.99.0.56/30      *[BGP/170] 00:12:03, localpref 100, from 7.0.0.74      // клиентский префикс
                      AS path: I, validation-state: unverified            // полученный от VMX-74 по iBGP
                    > to 10.99.0.2 via ge-0/0/0.0, Push 70

warrior@vmx-71# run show route 7.0.0.74
inet.0: 19 destinations, 19 routes (19 active, 0 holddown, 0 hidden)

7.0.0.74/32        *[IS-IS/18] 00:15:34, metric 30
                    > to 10.99.0.2 via ge-0/0/0.0

inet.3: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)

7.0.0.74/32        *[LDP/9] 00:06:33, metric 30
                    > to 10.99.0.2 via ge-0/0/0.0, Push 70                // LDP метка для лупбэка VMX-74
```

Связность (2)

```
warrior@vmx-21> traceroute 10.99.0.58
traceroute to 10.99.0.58 (10.99.0.58), 30 hops max, 40 byte packets
 1  10.99.0.54 (10.99.0.54)  2.463 ms  1.410 ms  3.604 ms                                  // vmx-71
 2  10.99.0.2 (10.99.0.2)  4.211 ms  3.584 ms  3.784 ms                                    // vmx-72
     MPLS Label=70 CoS=0 TTL=1 S=1
 3  10.99.0.6 (10.99.0.6)  4.378 ms 10.99.0.14 (10.99.0.14)  5.864 ms  4.283 ms          // vmx-73
     MPLS Label=84 CoS=0 TTL=1 S=1
 4  10.99.0.58 (10.99.0.58)  5.242 ms  3.165 ms  3.278 ms                                  // vmx-74

warrior@vmx-72# run show route table mpls.0 label 70
70                 *[LDP/9] 00:12:34, metric 1
                    > to 10.99.0.6 via ge-0/0/0.0, Swap 84              // VMX-72 делает своп 70->84
                      to 10.99.0.14 via ge-0/0/2.0, Swap 84             // кстати здесь ECMP (Equal Cost MultiPath)

warrior@vmx-72# run show route 7.0.0.74 table inet.3
7.0.0.74/32        *[LDP/9] 00:13:40, metric 20
                      to 10.99.0.6 via ge-0/0/0.0, Push 84              // 84 – это LDP метка для лупбэка VMX-74
                    > to 10.99.0.14 via ge-0/0/2.0, Push 84
```

Включаем Segment Routing

16.2R1

- Конфигурируем Node-SID на каждом P/PE
- Для наглядности, конфигурируем разную базу (SRGB)

```
warrior@vmx-71# show | compare
[edit protocols isis]
+    source-packet-routing {
+        srgb start-label 71000 index-range 100;
+        node-segment ipv4-index 71;
+    }
```

```
warrior@vmx-72# show | compare
[edit protocols isis]
+    source-packet-routing {
+        srgb start-label 72000 index-range 100;
+        node-segment ipv4-index 72;
+    }
```

```
warrior@vmx-73# show | compare
[edit protocols isis]
+    source-packet-routing {
+        srgb start-label 73000 index-range 100;
+        node-segment ipv4-index 73;
+    }
```

```
warrior@vmx-74# show | compare
[edit protocols isis]
+    source-packet-routing {
+        srgb start-label 74000 index-range 100;
+        node-segment ipv4-index 74;
+    }
```

IS-IS TLV с новой информацией (Node-SID)

```
warrior@vmx-71# run show isis database vmx-74.00-00 extensive
...
vmx-74.00-00 Sequence: 0x1c8, Checksum: 0x7cca, Lifetime: 1024 secs
  IPV4 Index: 74
  Node Segment Blocks Advertised:
    Start Index : 0, Size : 100, Label-Range: [ 74000, 74099 ]
   IS neighbor: vmx-73.00                         Metric:       10
     Two-way fragment: vmx-73.00-00, Two-way first fragment: vmx-73.00-00
...
    IP router id: 7.0.0.74
    Hostname: vmx-74
    Router Capability:  Router ID 7.0.0.74, Flags: 0x00
      SPRING Capability - Flags: 0xc0(I:1,V:1), Range: 100, SID-Label: 74000
    IS extended neighbor: vmx-73.00, Metric: default 10
      IP address: 10.99.0.10
      Neighbor's IP address: 10.99.0.9
      P2P IPV4 Adj-SID - Flags:0x30(F:0,B:0,V:1,L:1,S:0), Weight:0, Label: 86  // автоматически также создаются Adj-SID
    IP extended prefix: 7.0.0.74/32 metric 0 up                                // L:1 - флаг локальной значимости метки
      Node SID, Flags: 0x40(R:0,N:1,P:0,E:0,V:0,L:0), Algo: SPF(0), Value: 74
```

Что-нибудь изменилось? Пока ничего.

```
warrior@vmx-21> traceroute 10.99.0.58
traceroute to 10.99.0.58 (10.99.0.58), 30 hops max, 40 byte packets
 1  10.99.0.54 (10.99.0.54)  2.463 ms  1.410 ms  3.604 ms
 2  10.99.0.2 (10.99.0.2)  4.211 ms  3.584 ms  3.784 ms
     MPLS Label=70 CoS=0 TTL=1 S=1
 3  10.99.0.6 (10.99.0.6)  4.378 ms 10.99.0.14 (10.99.0.14)  5.864 ms  4.283 ms
     MPLS Label=84 CoS=0 TTL=1 S=1
 4  10.99.0.58 (10.99.0.58)  5.242 ms  3.165 ms  3.278 ms

warrior@vmx-71# run show route 10.99.0.58 table inet.0
10.99.0.56/30      *[BGP/170] 00:56:49, localpref 100, from 7.0.0.74
                      AS path: I, validation-state: unverified
                    > to 10.99.0.2 via ge-0/0/0.0, Push 70

warrior@vmx-71# run show route 7.0.0.74 table inet.3
7.0.0.74/32        *[LDP/9] 00:48:38, metric 30
                    > to 10.99.0.2 via ge-0/0/0.0, Push 70      // LDP метка
                    [L-ISIS/14] 00:12:59, metric 30
                    > to 10.99.0.2 via ge-0/0/0.0, Push 72074   // SR метка: 72000 (база vmx-72) + 74 (индекс vmx-74)
```

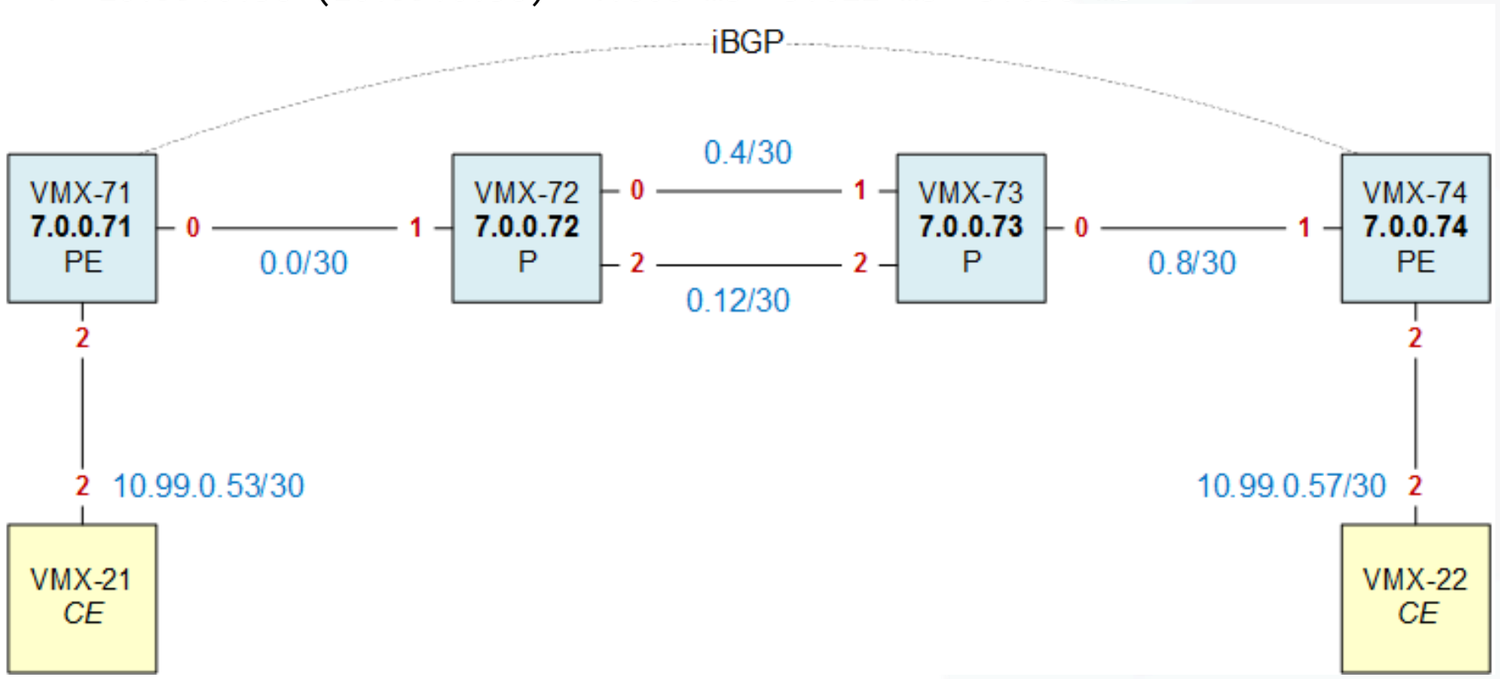
Меняем административную дистанцию для LDP

- Достаточно сделать только на ingress PE vmx-71 (!)

```
warrior@vmx-71# show | compare
[edit protocols ldp]
+    preference 15;

warrior@vmx-71# run show route 7.0.0.74 table inet.3

7.0.0.74/32        *[L-ISIS/14] 00:38:13, metric 30            // теперь Segment Routing более авторитетен, чем LDP
                    > to 10.99.0.2 via ge-0/0/0.0, Push 72074
                    [LDP/15] 00:00:01, metric 30
                    > to 10.99.0.2 via ge-0/0/0.0, Push 70

warrior@vmx-71# run show route 10.99.0.58 table inet.0

10.99.0.56/30      *[BGP/170] 01:24:35, localpref 100, from 7.0.0.74
                      AS path: I, validation-state: unverified
                    > to 10.99.0.2 via ge-0/0/0.0, Push 72074
```

Связность через Segment Routing

```
warrior@vmx-21> traceroute 10.99.0.58
traceroute to 10.99.0.58 (10.99.0.58), 30 hops max, 40 byte packets
 1  10.99.0.54 (10.99.0.54)  1.498 ms  1.246 ms  1.093 ms
 2  10.99.0.2 (10.99.0.2)  2.669 ms  3.086 ms  2.675 ms
     MPLS Label=72074 CoS=0 TTL=1 S=1                          // метка с vmx-71: база vmx-72 (72000) + индекс (74)
 3  10.99.0.14 (10.99.0.14)  2.837 ms  2.729 ms 10.99.0.6 (10.99.0.6)  3.159 ms
     MPLS Label=73074 CoS=0 TTL=1 S=1                          // метка с vmx-72: база vmx-73 (73000) + индекс (74)
 4  10.99.0.58 (10.99.0.58)  4.808 ms  3.022 ms  3.036 ms
```



Операции в FIB

```
warrior@vmx-71# run show route 7.0.0.74 table inet.3

7.0.0.74/32        *[L-ISIS/14] 00:58:21, metric 30
                    > to 10.99.0.2 via ge-0/0/0.0, Push 72074
                    [LDP/15] 00:20:09, metric 30
                    > to 10.99.0.2 via ge-0/0/0.0, Push 70

warrior@vmx-72# run show route table mpls.0 label 72074

72074              *[L-ISIS/14] 00:56:25, metric 20
                      to 10.99.0.6 via ge-0/0/0.0, Swap 73074
                    > to 10.99.0.14 via ge-0/0/2.0, Swap 73074

warrior@vmx-73# run show route table mpls.0 label 73074

73074              *[L-ISIS/14] 00:56:59, metric 10
                    > to 10.99.0.10 via ge-0/0/0.0, Pop          // vmx-73 выполняет PHP (снимает транспортную метку)
73074(S=0)         *[L-ISIS/14] 00:04:19, metric 10
                    > to 10.99.0.10 via ge-0/0/0.0, Pop
```

Одинаковый SRGB

- Одна и та же метка для форвардинга – легче решать проблемы

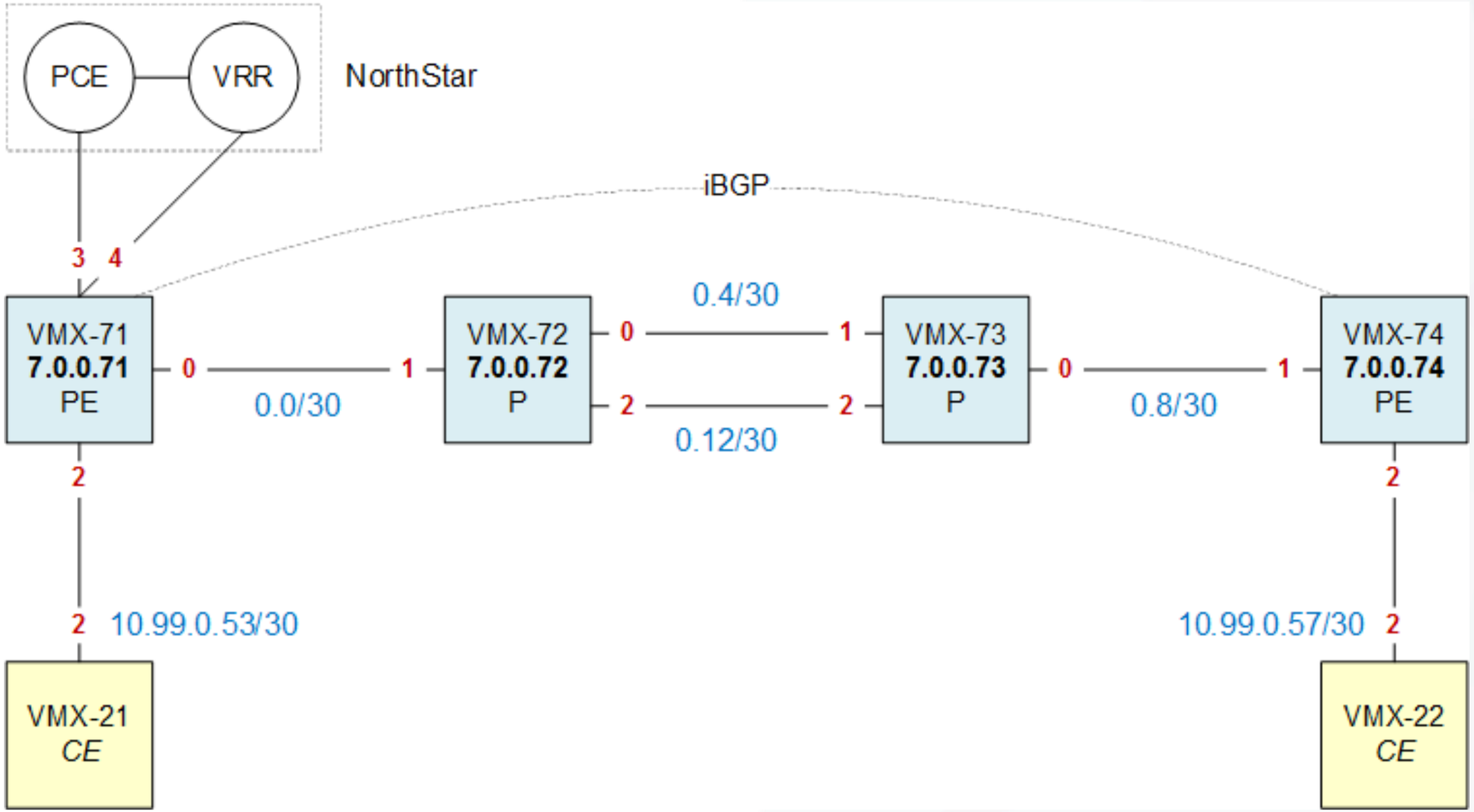
```
warrior@vmx-71# set protocols isis source-packet-routing srgb start-label 90000
warrior@vmx-72# set protocols isis source-packet-routing srgb start-label 90000
warrior@vmx-73# set protocols isis source-packet-routing srgb start-label 90000
warrior@vmx-74# set protocols isis source-packet-routing srgb start-label 90000

warrior@vmx-21> traceroute 10.99.0.58
traceroute to 10.99.0.58 (10.99.0.58), 30 hops max, 40 byte packets
 1  10.99.0.54 (10.99.0.54)  2.872 ms  108.387 ms  136.439 ms
 2  10.99.0.2 (10.99.0.2)  3.744 ms  3.335 ms  3.490 ms          // для форвардинга до vmx-74 все узлы используют
     MPLS Label=90074 CoS=0 TTL=1 S=1                           // одну и ту же метку 90074
 3  10.99.0.6 (10.99.0.6)  3.237 ms  3.450 ms  3.229 ms
     MPLS Label=90074 CoS=0 TTL=1 S=1
 4  10.99.0.58 (10.99.0.58)  26.672 ms  29.565 ms  30.039 ms
```

Пример: SR-TE из NorthStar

TE = Traffic Engineering

Виртуальная топология

- Добавляем NorthStar контроллер в домен IS-IS

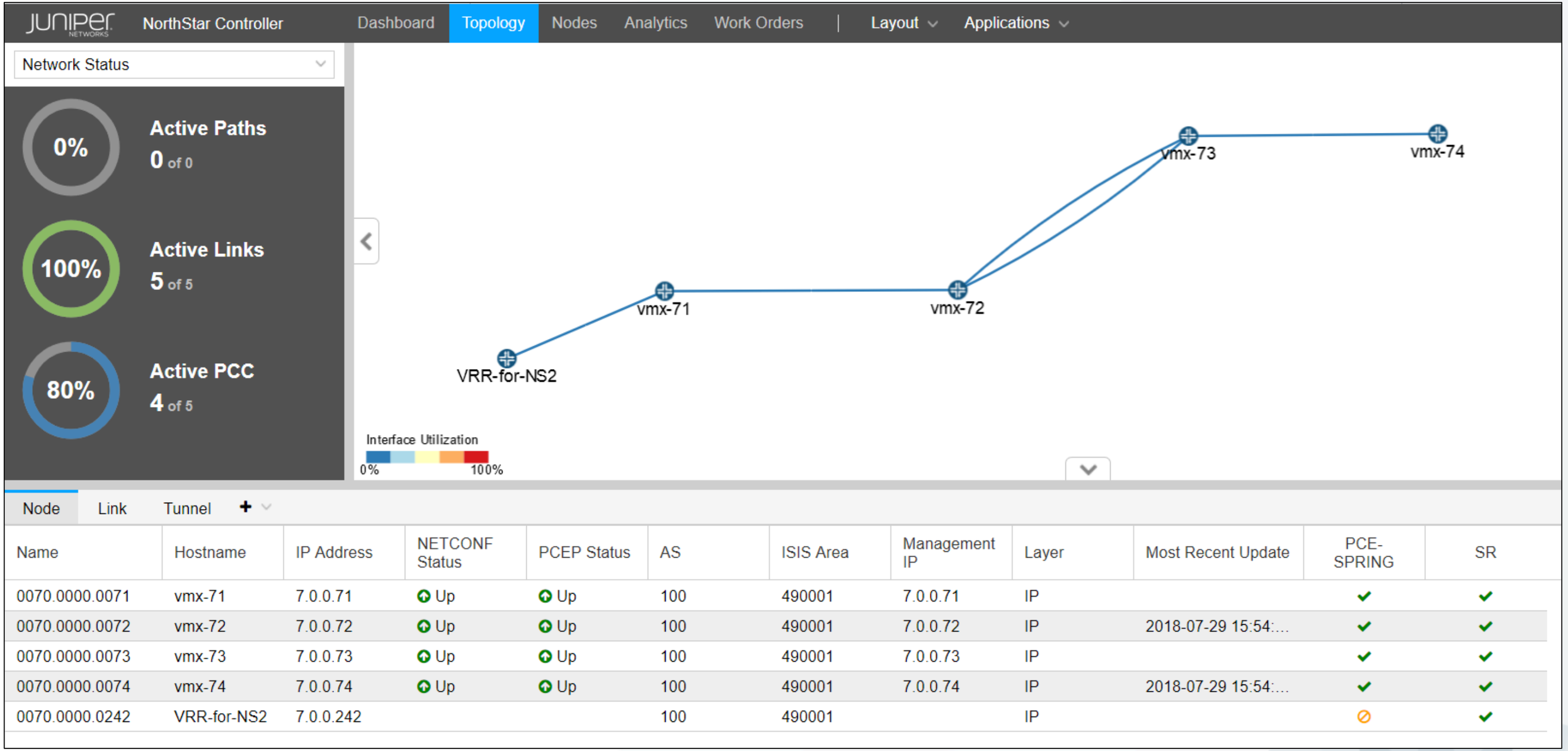


Добавляем NorthStar в конфигурацию каждого P/PE

```
warrior@vmx-71# show | compare
[edit protocols mpls]
+    lsp-external-controller pccd;
[edit protocols]
+   spring-traffic-engineering {
+       lsp-external-controller pccd;
+   }
+   pcep {
+       pce NS400 {                                  // PCE = Path Computation Element (=NorthStar)
+           local-address 7.0.0.71;
+           destination-ipv4-address 10.77.0.2;
+           destination-port 4189;
+           pce-type active stateful;
+           lsp-provisioning;
+           lsp-cleanup-timer 180;
+           spring-capability;
+       }
+   }
```

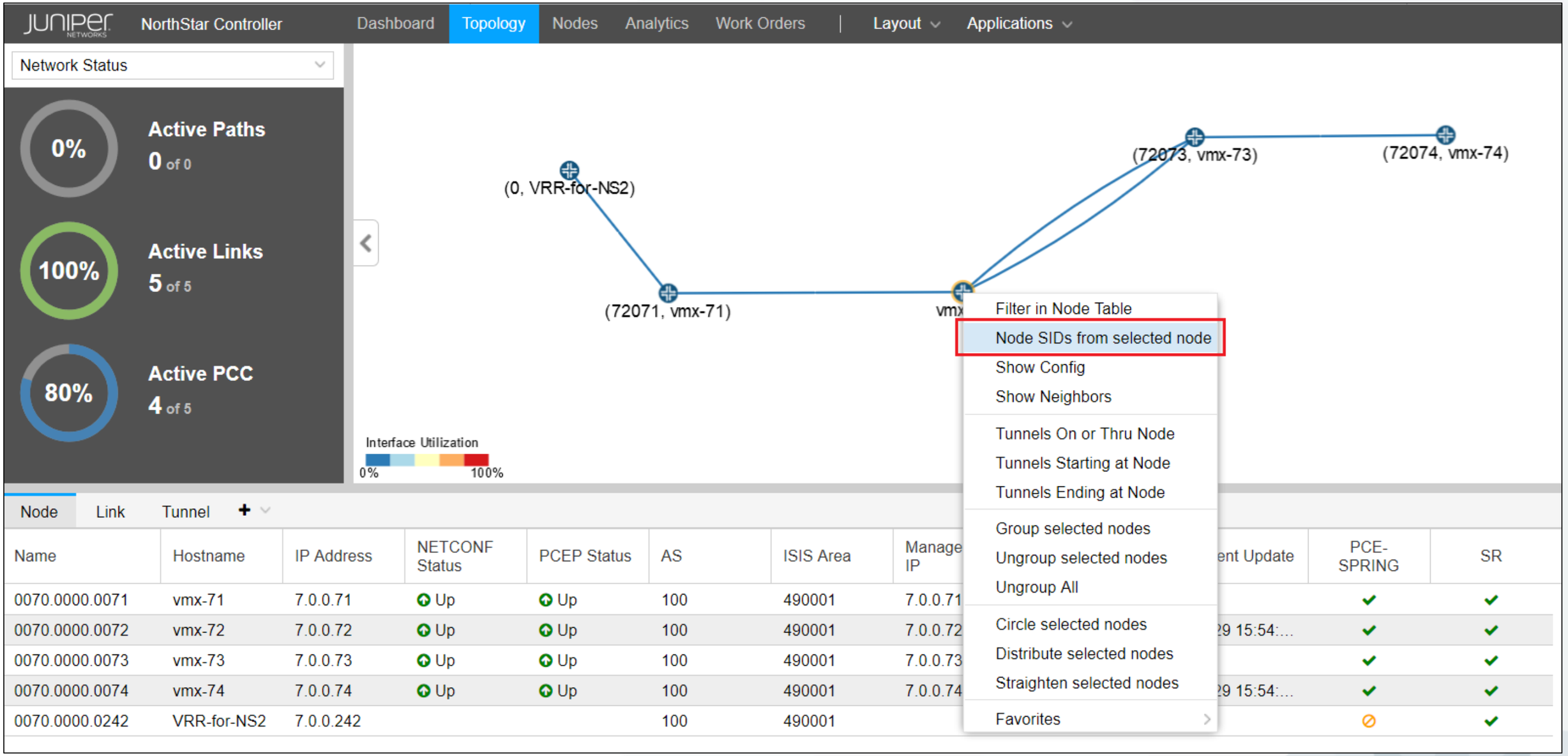
Проверка статуса PCE на узле сети

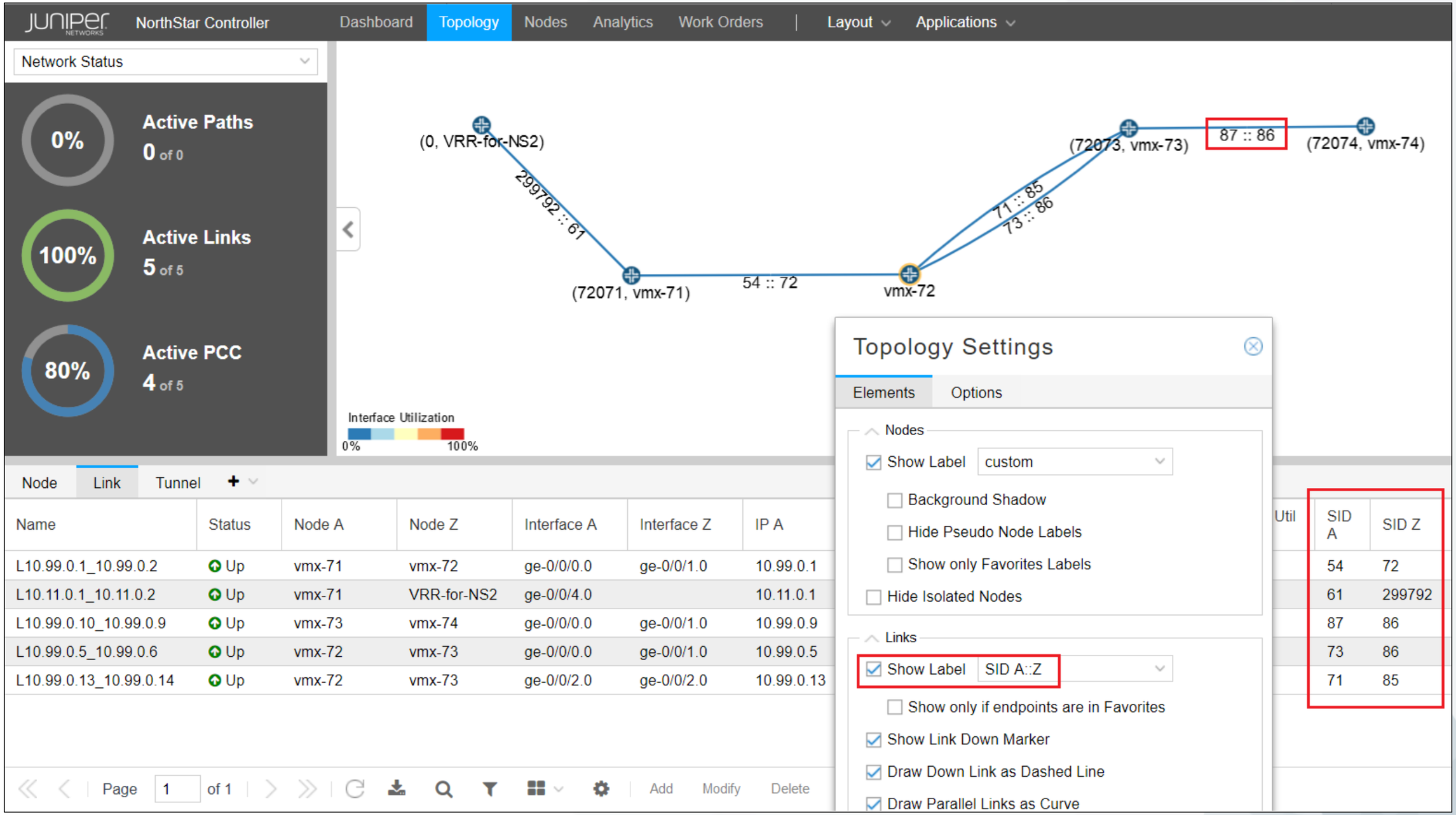
```
warrior@vmx-74# run show path-computation-client active-pce
PCE NS400
-----------------------------------------
General
    PCE IP address           : 10.77.0.2
    Local IP address         : 7.0.0.74
    Priority                 : 0
    PCE status               : PCE_STATE_UP                  // все в порядке
    Session type             : PCE_TYPE_STATEFULACTIVE
    LSP provisioning allowed : On
    P2MP LSP report allowed  : Off
    P2MP LSP update allowed  : Off
    P2MP LSP init allowed    : Off
    PCE-mastership           : main

warrior@vmx-74# run show spring-traffic-engineering lsp

Total displayed LSPs: 0 (Up: 0, Down: 0)
```



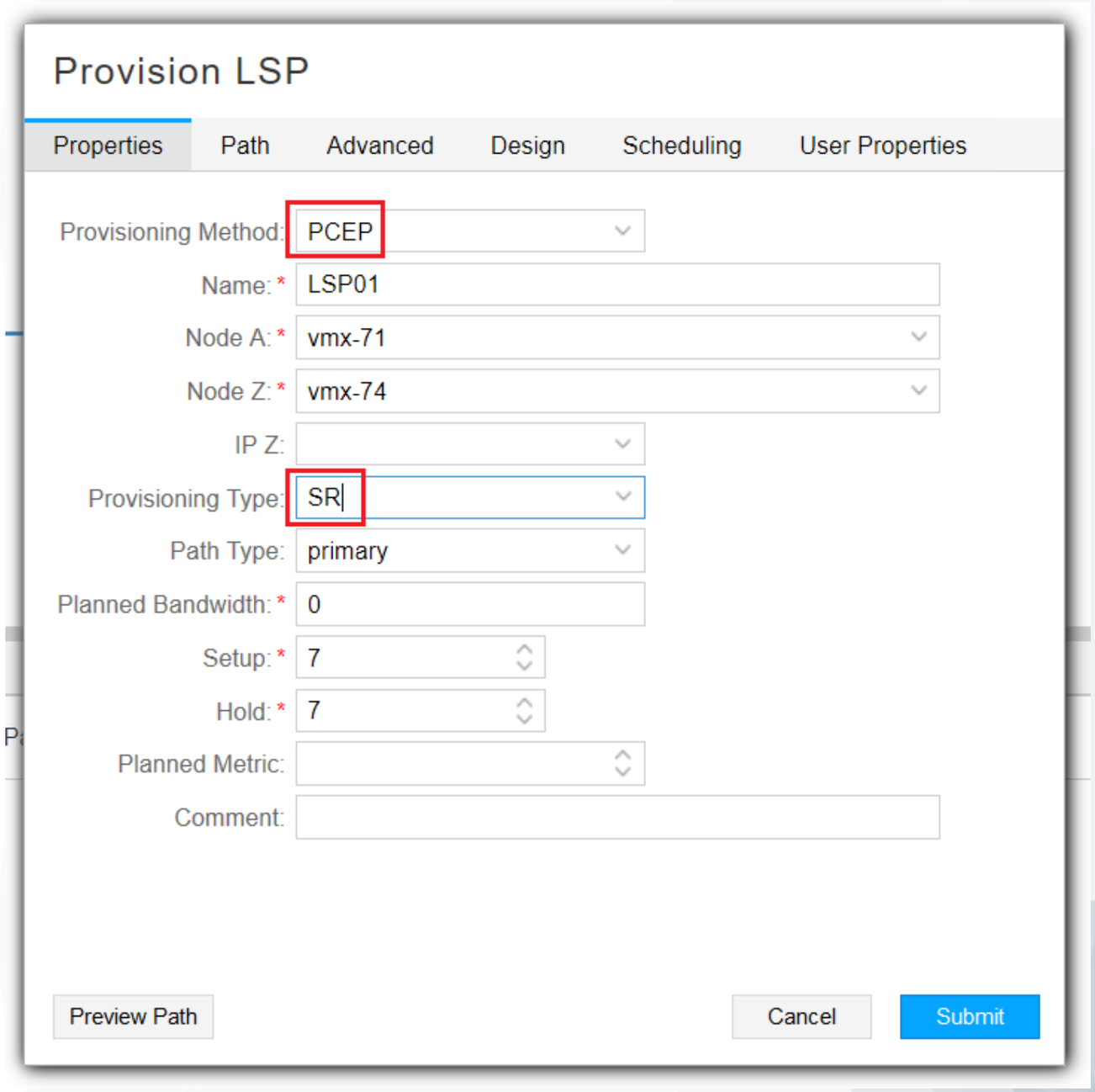
Node | Link | Tunnel

Name	Hostname	IP Address	NETCONF Status	PCEP Status	AS	ISIS Area	Management IP	Layer	Most Recent Update	PCE-SPRING	SR
0070.0000.0071	vmx-71	7.0.0.71	Up	Up	100	490001	7.0.0.71	IP		✔	✔
0070.0000.0072	vmx-72	7.0.0.72	Up	Up	100	490001	7.0.0.72	IP	2018-07-29 15:54:…	✔	✔
0070.0000.0073	vmx-73	7.0.0.73	Up	Up	100	490001	7.0.0.73	IP		✔	✔
0070.0000.0074	vmx-74	7.0.0.74	Up	Up	100	490001	7.0.0.74	IP	2018-07-29 15:54:…	✔	✔
0070.0000.0242	VRR-for-NS2	7.0.0.242			100	490001		IP		⊘	✔

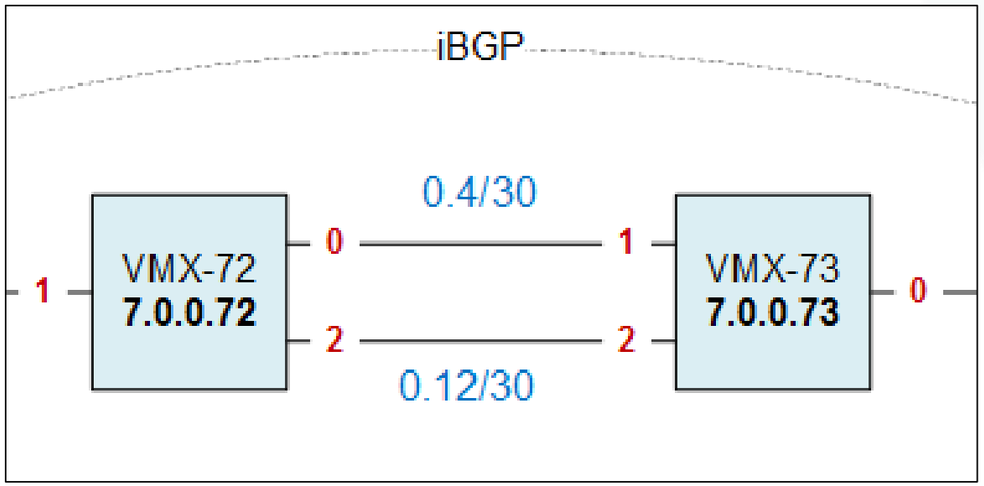


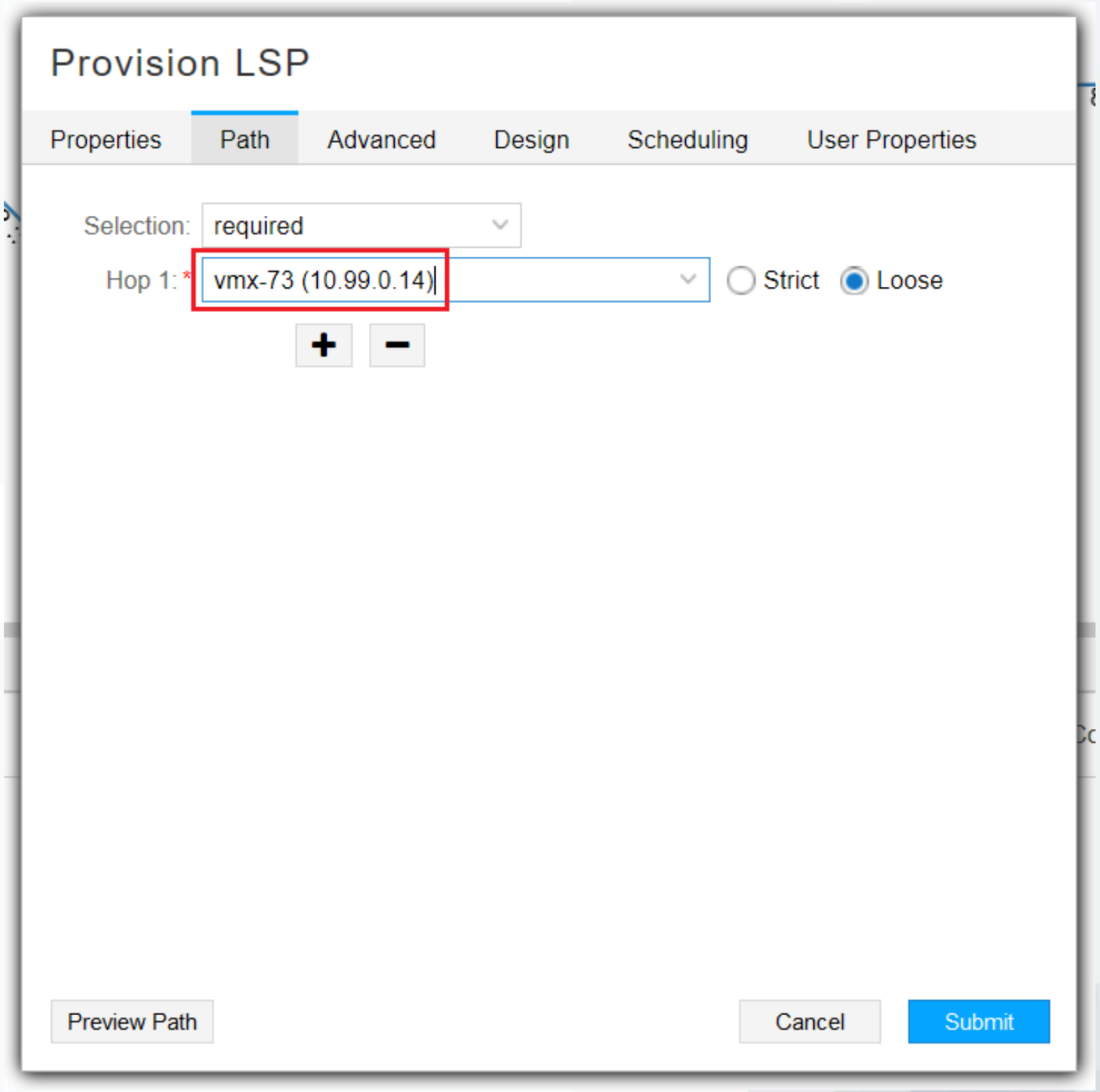


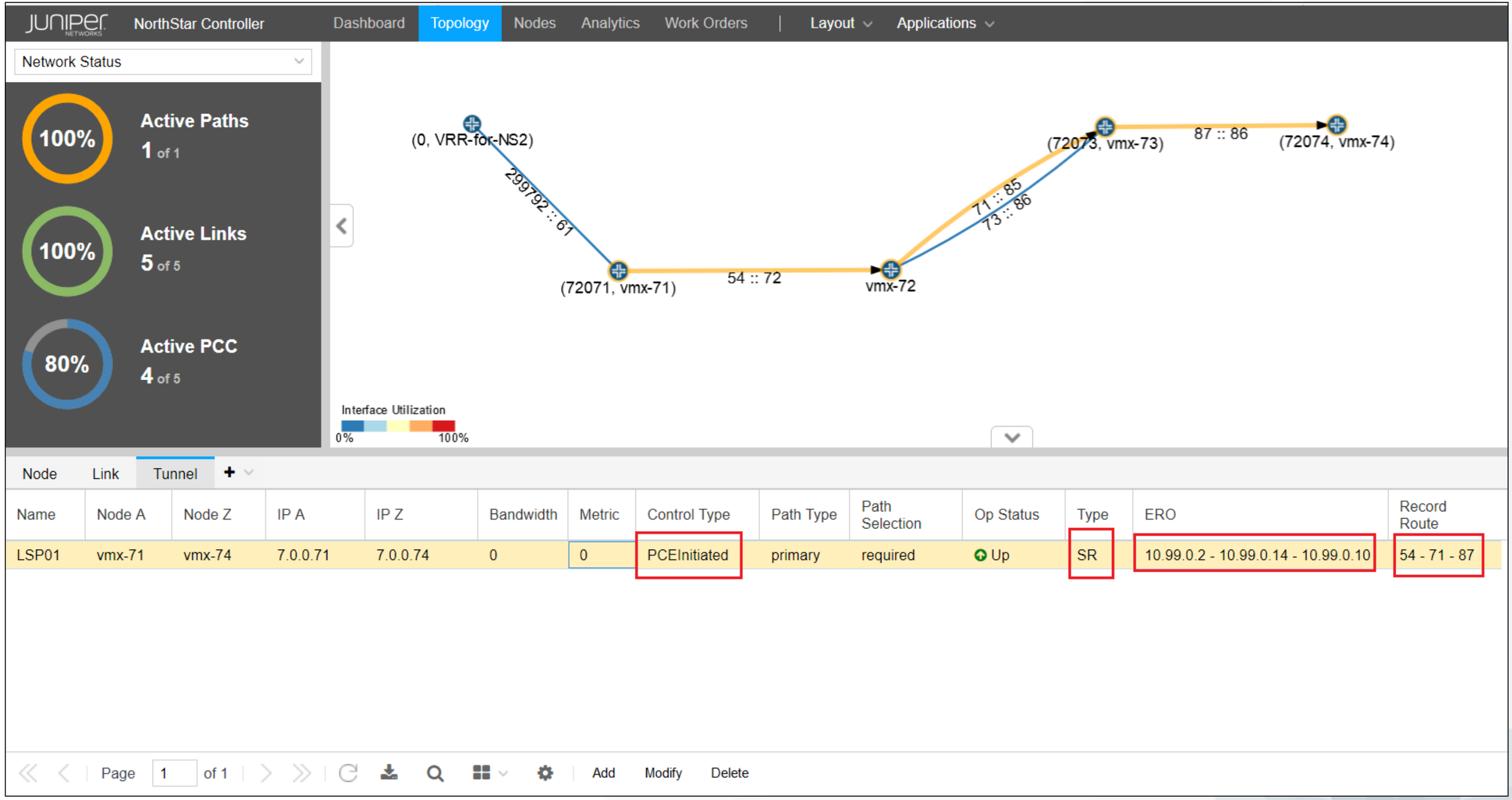
Прокладываем SR-TE LSP (1)



Прокладываем SR-TE LSP (2)







Новая связность (1)

```
warrior@vmx-71# run show route 10.99.0.58 table inet.0

10.99.0.56/30      *[BGP/170] 00:18:42, localpref 100, from 7.0.0.74
                      AS path: I, validation-state: unverified
                    > to 10.99.0.2 via ge-0/0/0.0, Push 87, Push 71(top)

warrior@vmx-71# run show route 7.0.0.74 table inet.3

7.0.0.74/32        *[SPRING-TE/8] 00:15:23, metric 1                    // SPRING Traffic Engineering
                    > to 10.99.0.2 via ge-0/0/0.0, Push 87, Push 71(top)
                    [L-ISIS/14] 01:30:18, metric 30
                    > to 10.99.0.2 via ge-0/0/0.0, Push 72074
```

Новая связность (2)

- Только ingress PE (vmx-71) и контроллер знают об этом LSP (!)

```
warrior@vmx-71# run show spring-traffic-engineering lsp detail
Name: LSP01
To: 7.0.0.74
State: Up, Outgoing interface: ge-0/0/0.0
SR-ERO hop count: 3
  Hop 1 (Strict):
    NAI: IPv4 Adjacency ID, 10.99.0.1 -> 10.99.0.2
    SID type: 20-bit label, Value: 54
  Hop 2 (Strict):
    NAI: IPv4 Adjacency ID, 10.99.0.13 -> 10.99.0.14
    SID type: 20-bit label, Value: 71
  Hop 3 (Strict):
    NAI: IPv4 Adjacency ID, 10.99.0.9 -> 10.99.0.10
    SID type: 20-bit label, Value: 87

Total displayed LSPs: 1 (Up: 1, Down: 0)
```

```
warrior@vmx-72# run show spring-traffic-engineering lsp detail

Total displayed LSPs: 0 (Up: 0, Down: 0)
```

```
warrior@vmx-73# run show spring-traffic-engineering lsp detail

Total displayed LSPs: 0 (Up: 0, Down: 0)
```

```
warrior@vmx-74# run show spring-traffic-engineering lsp detail

Total displayed LSPs: 0 (Up: 0, Down: 0)
```

Новая связность (3)

Было

```
warrior@vmx-21> traceroute 10.99.0.58
traceroute to 10.99.0.58 (10.99.0.58), 30 hops max, 40 byte packets
 1  10.99.0.54 (10.99.0.54)  2.022 ms  1.642 ms  1.362 ms
 2  10.99.0.2 (10.99.0.2)  4.970 ms  4.404 ms  3.451 ms
     MPLS Label=72074 CoS=0 TTL=1 S=1
 3  10.99.0.6 (10.99.0.6)  5.769 ms 10.99.0.14 (10.99.0.14)  3.483 ms 10.99.0.6 (10.99.0.6)  4.257 ms
     MPLS Label=73074 CoS=0 TTL=1 S=1
 4  10.99.0.58 (10.99.0.58)  4.904 ms  3.595 ms  3.905 ms
```

Стало

```
warrior@vmx-21> traceroute 10.99.0.58
traceroute to 10.99.0.58 (10.99.0.58), 30 hops max, 40 byte packets
 1  10.99.0.54 (10.99.0.54)  36.455 ms  55.820 ms  43.971 ms
 2  10.99.0.2 (10.99.0.2)  4.900 ms  57.481 ms  79.901 ms
     MPLS Label=71 CoS=0 TTL=1 S=0
     MPLS Label=87 CoS=0 TTL=1 S=1
 3  10.99.0.14 (10.99.0.14)  8.476 ms  5.306 ms  10.857 ms
     MPLS Label=87 CoS=0 TTL=1 S=1
 4  10.99.0.58 (10.99.0.58)  7.892 ms  13.221 ms  7.325 ms
```

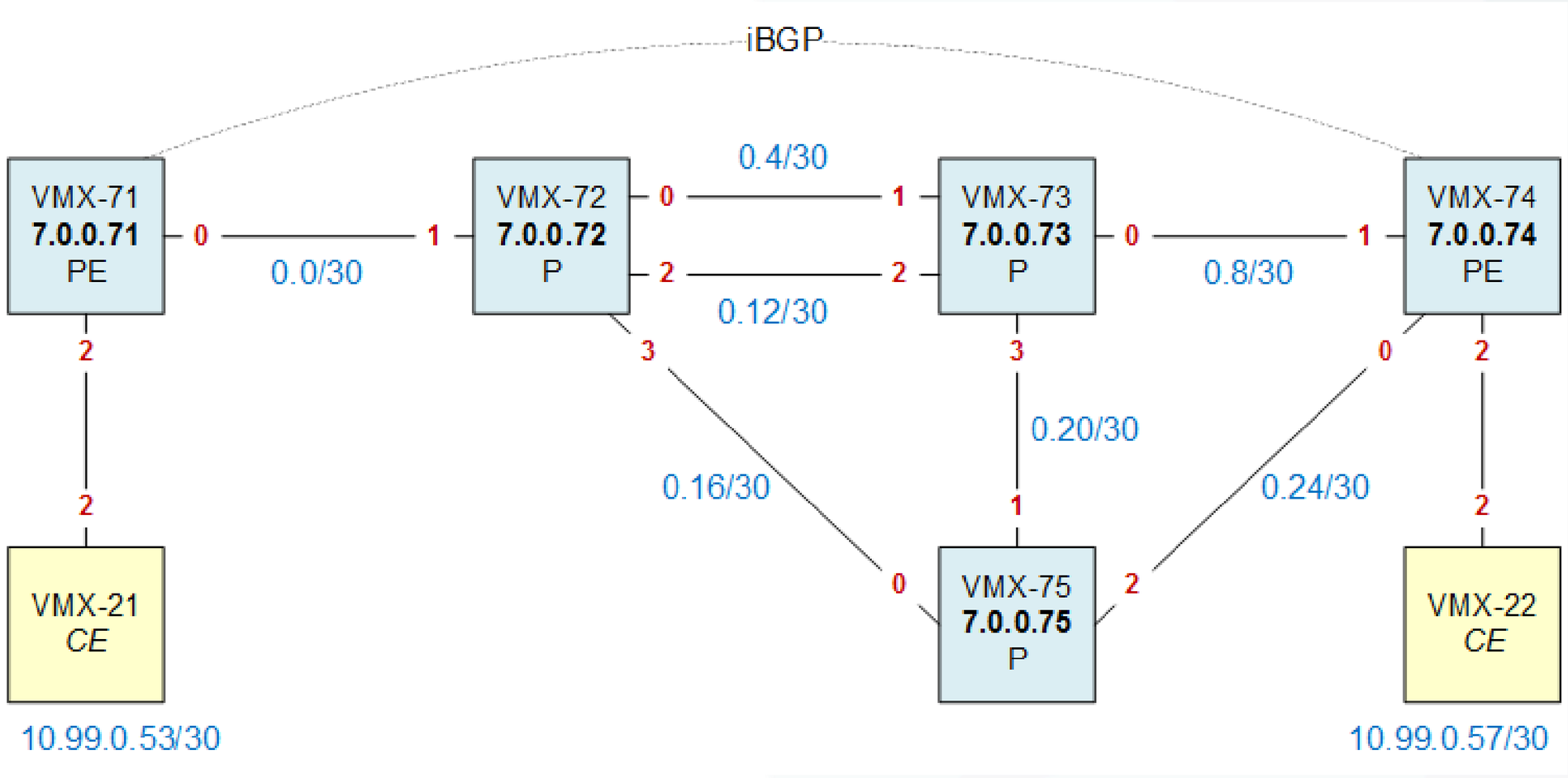
Операции в FIB

```
warrior@vmx-71# run show route 7.0.0.74 table inet.3

7.0.0.74/32        *[SPRING-TE/8] 00:34:23, metric 1
                    > to 10.99.0.2 via ge-0/0/0.0, Push 87, Push 71(top)
                    [L-ISIS/14] 01:49:18, metric 30
                    > to 10.99.0.2 via ge-0/0/0.0, Push 72074

warrior@vmx-72# run show route table mpls.0 label 71

71                 *[L-ISIS/14] 02:59:52, metric 0
                    > to 10.99.0.14 via ge-0/0/2.0, Pop          // сняли верхнюю метку (71), осталась нижняя (87)
71(S=0)            *[L-ISIS/14] 00:11:34, metric 0
                    > to 10.99.0.14 via ge-0/0/2.0, Pop

warrior@vmx-73# run show route table mpls.0 label 87

87                 *[L-ISIS/14] 03:00:34, metric 0
                    > to 10.99.0.10 via ge-0/0/0.0, Pop
87(S=0)            *[L-ISIS/14] 00:11:54, metric 0
                    > to 10.99.0.10 via ge-0/0/0.0, Pop
```

Пример: Anycast-SID

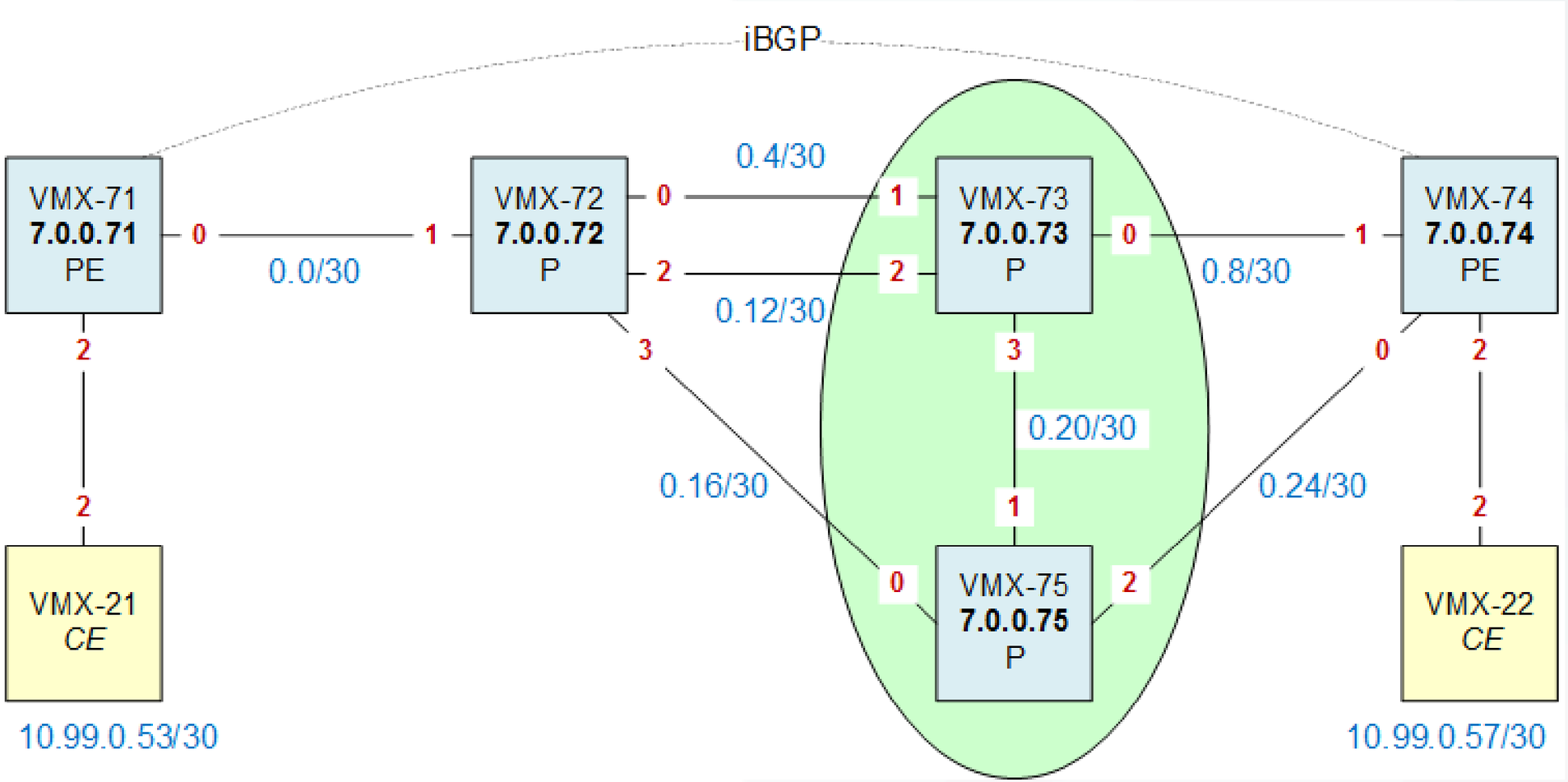
Виртуальная топология

- Добавим еще один роутер в нашу топологию: vmx-75



Anycast-SID

- Соберем vmx-73 и vmx-75 в одну anycast группу



Anycast-SID - конфигурация

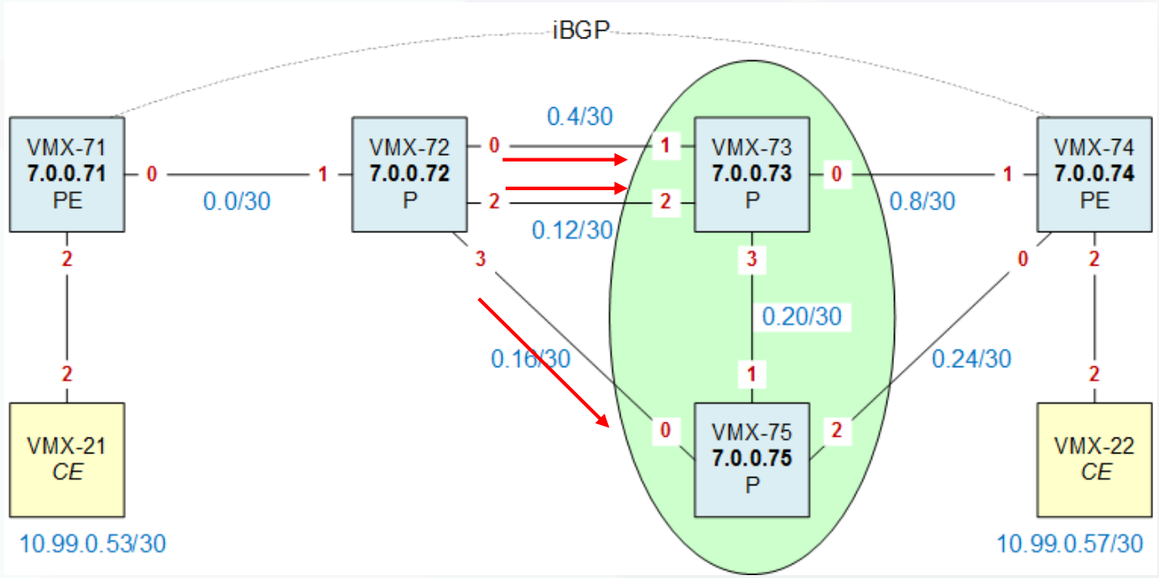
```
warrior@vmx-73# show | compare
[edit interfaces lo0 unit 0 family inet]
+       address 7.0.0.99/32;
[edit protocols isis]
+   export pl-isis-export;
[edit policy-options]
+   policy-statement pl-isis-export {
+       term 1 {
+           from {
+               protocol direct;
+               interface lo0.0;
+               route-filter 7.0.0.99/32 exact;
+           }
+           then {
+               prefix-segment index 99;
+               accept;
+           }
+       }
+   }
```

```
warrior@vmx-75# show | compare
[edit interfaces lo0 unit 0 family inet]
+       address 7.0.0.99/32;
[edit protocols isis]
+   export pl-isis-export;
[edit policy-options]
+   policy-statement pl-isis-export {
+       term 1 {
+           from {
+               protocol direct;
+               interface lo0.0;
+               route-filter 7.0.0.99/32 exact;
+           }
+           then {
+               prefix-segment index 99;
+               accept;
+           }
+       }
+   }
```

Anycast-SID в базе данных ISIS

```
warrior@vmx-71# run show isis database vmx-73.00-00 extensive | find TLVs
...
    IP extended prefix: 7.0.0.73/32 metric 0 up                          // известный нам Node-SID
      8 bytes of subtlvs
      Node SID, Flags: 0x40(R:0,N:1,P:0,E:0,V:0,L:0), Algo: SPF(0), Value: 73
    IP extended prefix: 7.0.0.99/32 metric 0 up                          // Prefix-SID, новый анонс
      8 bytes of subtlvs
      Prefix SID, Flags: 0x00(R:0,N:0,P:0,E:0,V:0,L:0), Algo: SPF(0), Value: 99
    IP address: 7.0.0.99

warrior@vmx-71# run show isis database vmx-75.00-00 extensive | find TLVs
...
    IP extended prefix: 7.0.0.99/32 metric 0 up                          // Prefix-SID, новый анонс
      8 bytes of subtlvs
      Prefix SID, Flags: 0x00(R:0,N:0,P:0,E:0,V:0,L:0), Algo: SPF(0), Value: 99
    IP extended prefix: 7.0.0.75/32 metric 0 up                          // известный нам Node-SID
      8 bytes of subtlvs
      Node SID, Flags: 0x40(R:0,N:1,P:0,E:0,V:0,L:0), Algo: SPF(0), Value: 75
    IP address: 7.0.0.99
```

Операции в FIB

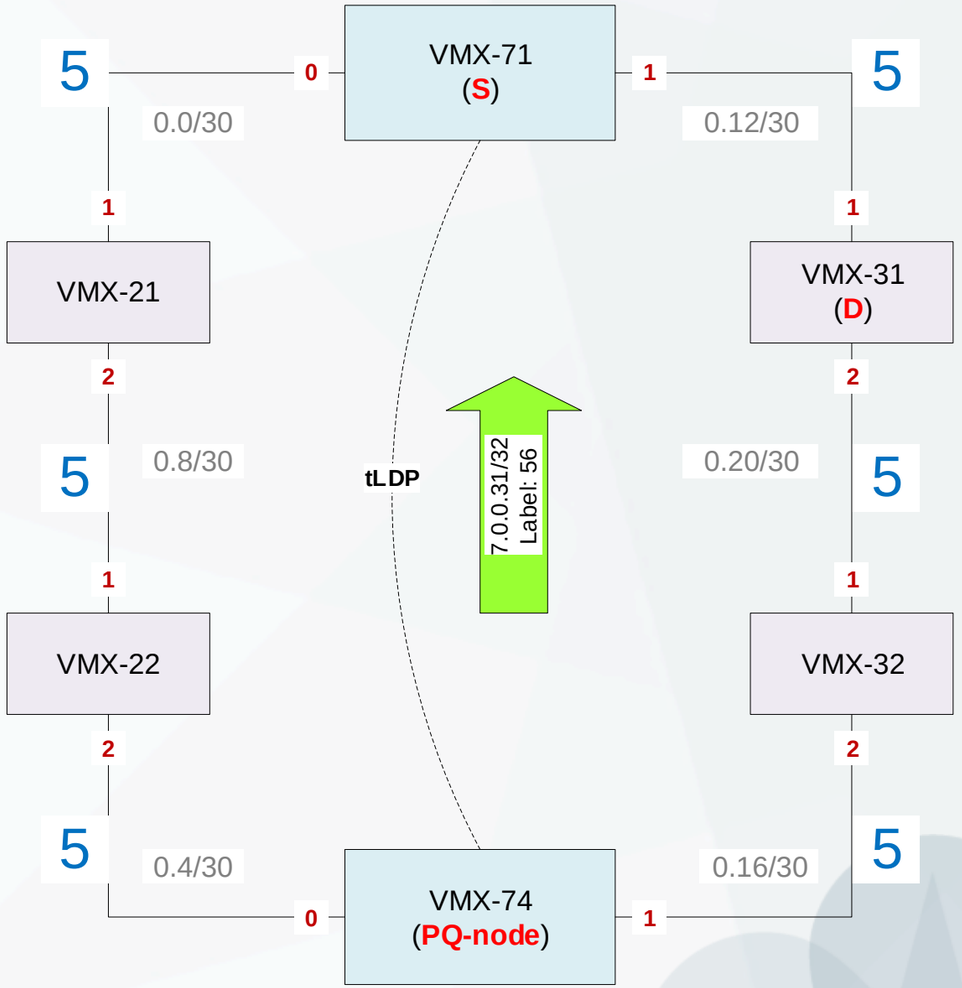



```
warrior@vmx-71# run show route table mpls.0 label 90099

90099              *[L-ISIS/14] 00:15:48, metric 20
                    > to 10.99.0.2 via ge-0/0/0.0, Swap 90099

warrior@vmx-72# run show route table mpls.0 label 90099

90099              *[L-ISIS/14] 00:15:23, metric 10
                      to 10.99.0.6 via ge-0/0/0.0, Pop                    // to VMX-73
                      to 10.99.0.14 via ge-0/0/2.0, Pop                   // to VMX-73
                    > to 10.99.0.18 via ge-0/0/3.0, Pop                   // to VMX-75

warrior@vmx-72# run show route forwarding-table label 90099               // ECMP балансировка
Routing table: default.mpls
Destination        Type RtRef Next hop           Type Index    NhRef Netif
90099              user     0                    ulst  1048578     2           // тип “unicast list”, т.е. “список”
                              10.99.0.6         Pop         606     4 ge-0/0/0.0
                              10.99.0.14        Pop         602     4 ge-0/0/2.0
                              10.99.0.18        Pop         604     4 ge-0/0/3.0
```

Пример: (R)LFA через SR

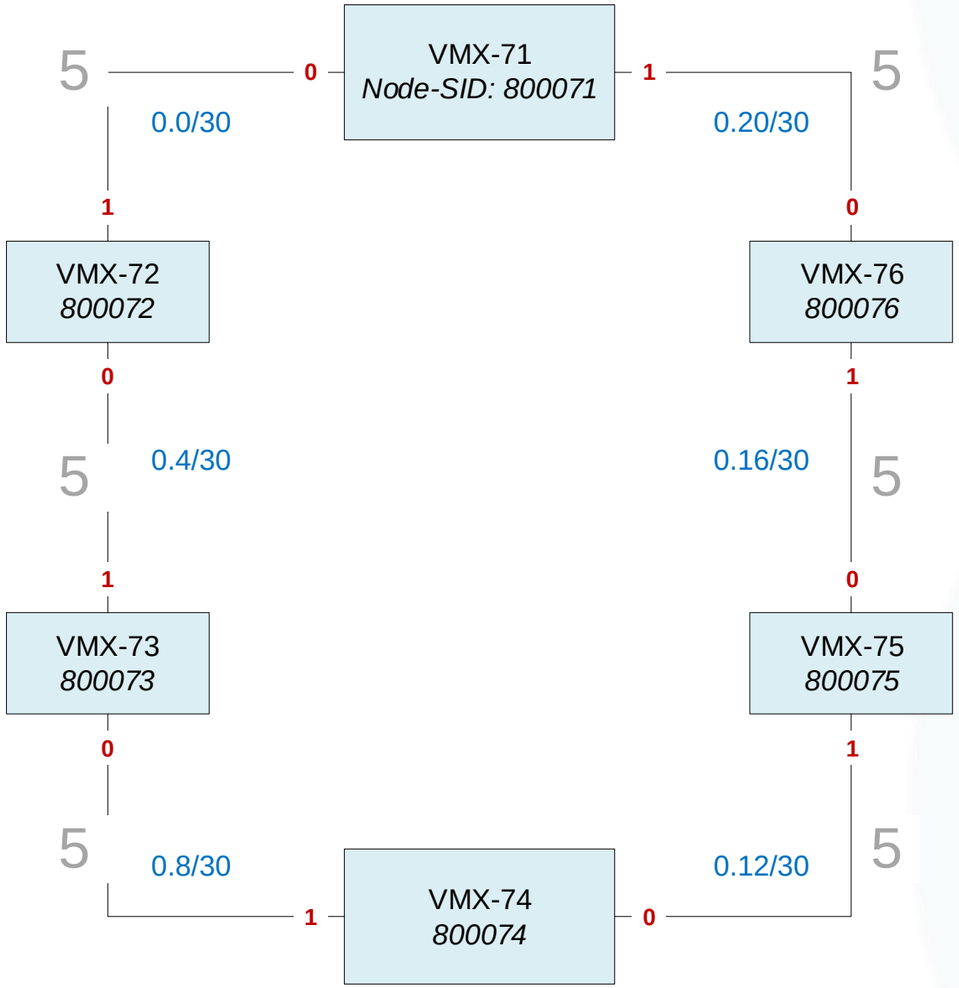
Классический (R)LFA для ISIS или OSPF

- Таргетированная LDP сессия до PQ-роутера
- Подробности в презентации “(R)LFA & TI-FRR”
- В большой сети потенциально имеем большое количество tLDP сессий (устанавливаемых автоматически)



Виртуальная топология (1)

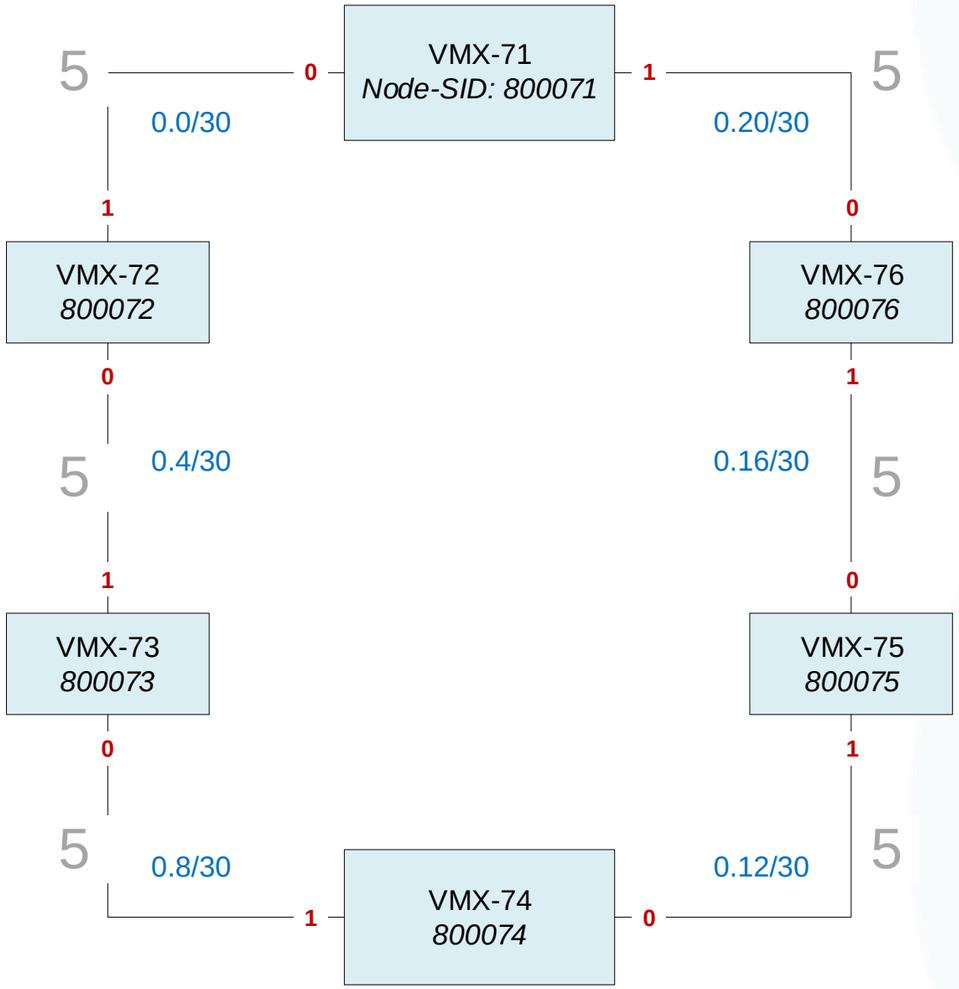
- Кольцо с включенным SR




```
warrior@vmx-71# show protocols isis
reference-bandwidth 10g;
source-packet-routing {
    srgb start-label 800000 index-range 100;
    node-segment ipv4-index 71;
}
level 1 disable;
level 2 wide-metrics-only;
interface ge-0/0/0.0 {
    point-to-point;
    level 2 metric 5;
}
interface ge-0/0/1.0 {
    point-to-point;
    level 2 metric 5;
}
interface lo0.0 {
    passive;
}

warrior@vmx-71# run show ldp session
LDP instance is not running
```

Виртуальная топология (2)

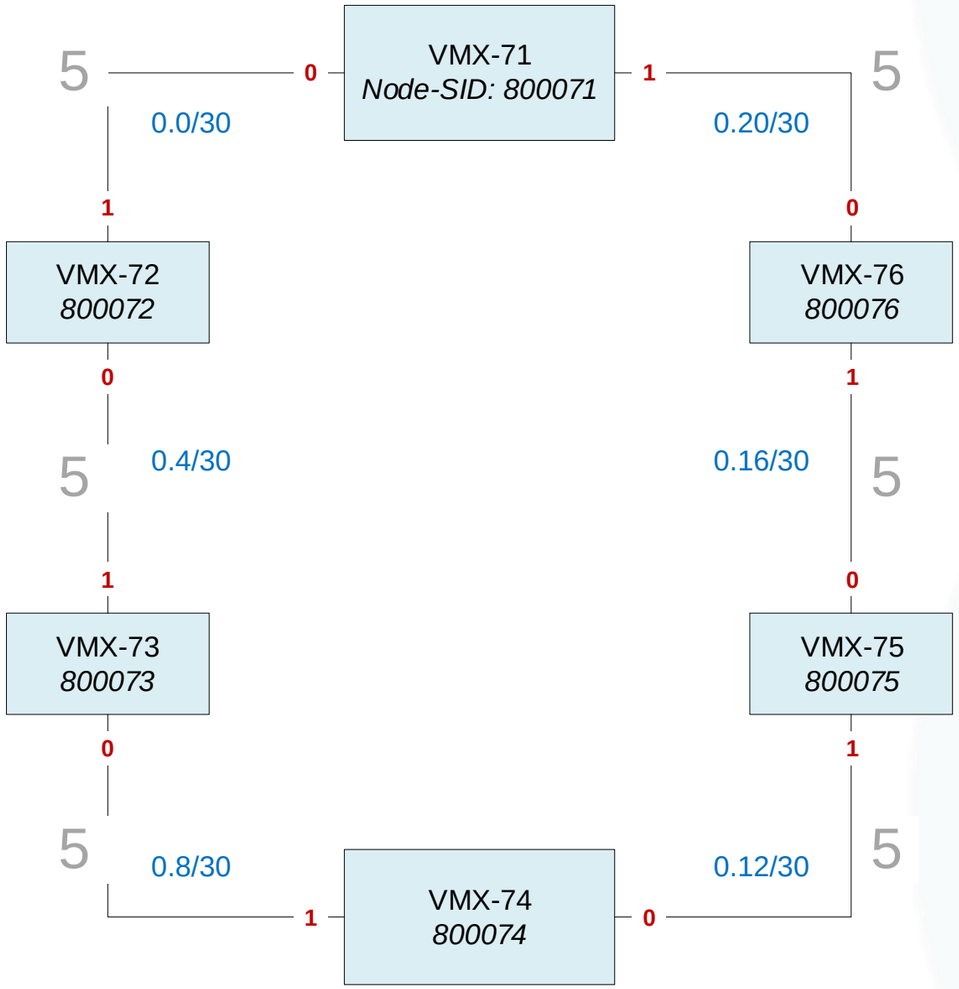
- LFA неэффективен, требуется (R)LFA




```
warrior@vmx-71# show | compare
[edit protocols isis interface ge-0/0/0.0]
+    link-protection;                              // включаем LFA
[edit protocols isis interface ge-0/0/1.0]
+    link-protection;                              // включаем LFA

warrior@vmx-71# run show isis backup coverage     // покрытие
Backup Coverage:
Topology        Level    Node     IPv4     IPv4-MPLS     IPv6     IPv6-MPLS
IPV4 Unicast        2  20.00%  11.11%       20.00%   0.00%          0.00%
```

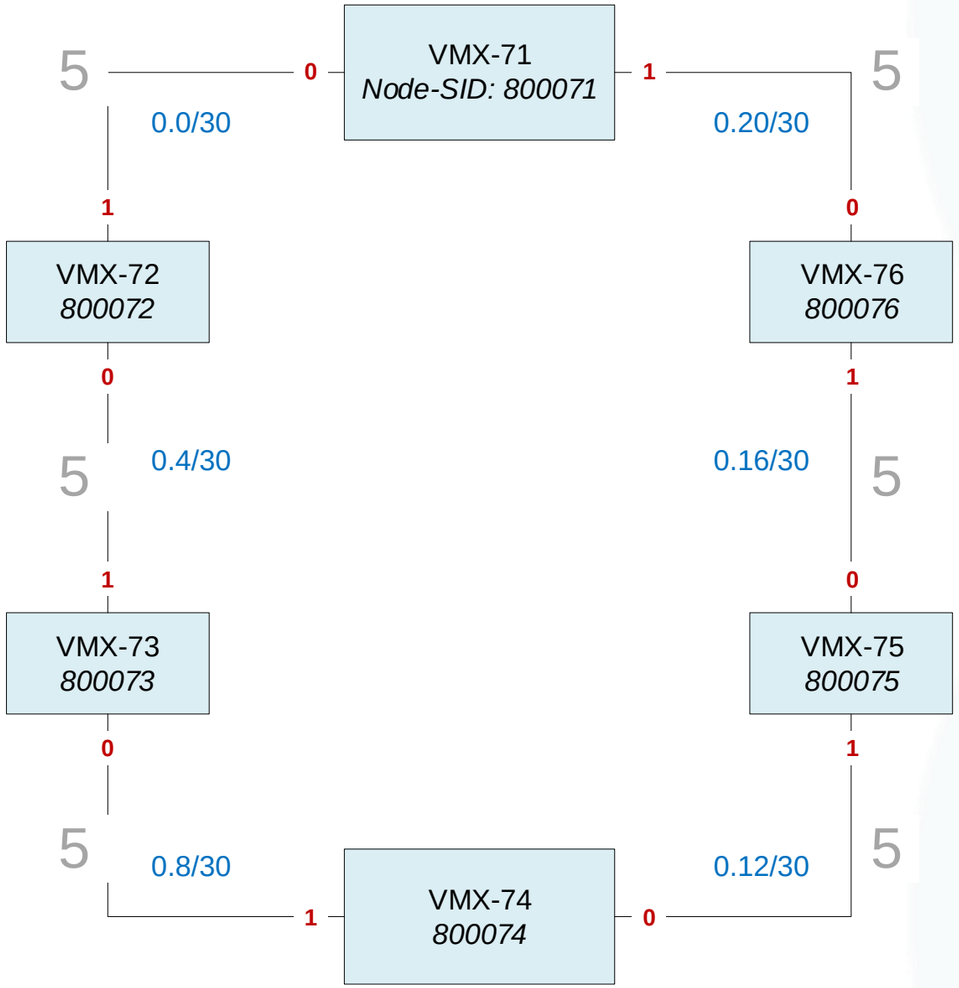
Виртуальная топология (3)

- Запасной маршрут есть только до VMX-74 (ECMP)




```
warrior@vmx-71# run show route protocol isis 7.0.0.0/24 table inet.3

7.0.0.72/32        *[L-ISIS/14] 00:23:12, metric 5
                    > to 10.99.0.2 via ge-0/0/0.0
7.0.0.73/32        *[L-ISIS/14] 00:23:12, metric 10
                    > to 10.99.0.2 via ge-0/0/0.0, Push 800073
7.0.0.74/32        *[L-ISIS/14] 00:36:05, metric 15
                       to 10.99.0.2 via ge-0/0/0.0, Push 800074
                    > to 10.99.0.21 via ge-0/0/1.0, Push 800074
7.0.0.75/32        *[L-ISIS/14] 00:23:12, metric 10
                    > to 10.99.0.21 via ge-0/0/1.0, Push 800075
7.0.0.76/32        *[L-ISIS/14] 00:23:12, metric 5
                    > to 10.99.0.21 via ge-0/0/1.0

// 5 соседей в домене, защищен только 1 = покрытие 20.00%
```

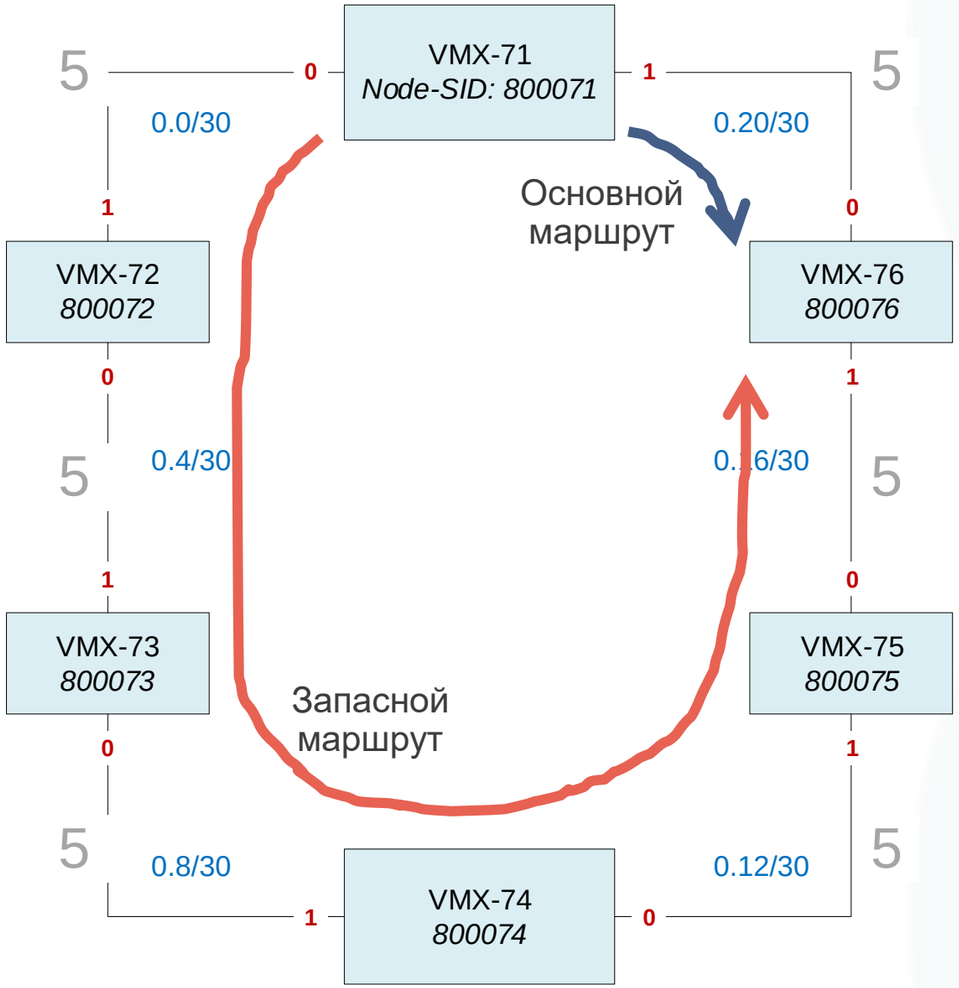
Включаем (R)LFA




```
warrior@vmx-71# show | compare
[edit protocols isis]
+    backup-spf-options {
+        remote-backup-calculation;     // включаем (R)LFA
+        use-source-packet-routing;     // не только для MPLS трафика
+                                       // (inet.3, inet6.3, mpls.0),
+                                       // но и для IPv4/IPv6 трафика
+                                       // (inet.0, inet6.0)
+    }
```

```
warrior@vmx-71# run show isis backup coverage
Backup Coverage:
Topology        Level    Node     IPv4     IPv4-MPLS     IPv6     IPv6-MPLS
IPV4 Unicast        2 100.00% 100.00%       100.00%    0.00%          0.00%
```

Результат




```
warrior@vmx-71# run show route 7.0.0.76                    // например

inet.0: 16 destinations, 16 routes (16 active, 0 holddown, 0 hidden)

7.0.0.76/32        *[IS-IS/18] 00:09:37, metric 5
                    > to 10.99.0.21 via ge-0/0/1.0
                      to 10.99.0.2 via ge-0/0/0.0, Push 800074

inet.3: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)

7.0.0.76/32        *[L-ISIS/14] 00:09:37, metric 5
                    > to 10.99.0.21 via ge-0/0/1.0
                      to 10.99.0.2 via ge-0/0/0.0, Push 800076,
                                                   Push 800074(top)

// 800074 = верхняя метка, доставить до VMX-74 (PQ-node !!!)
// 800076 = затем доставить до VMX-76

warrior@vmx-74# run show route table mpls.0 label 800076
800076             *[L-ISIS/14] 01:21:43, metric 10
                    > to 10.99.0.14 via ge-0/0/0.0, Swap 800076
```

Результат

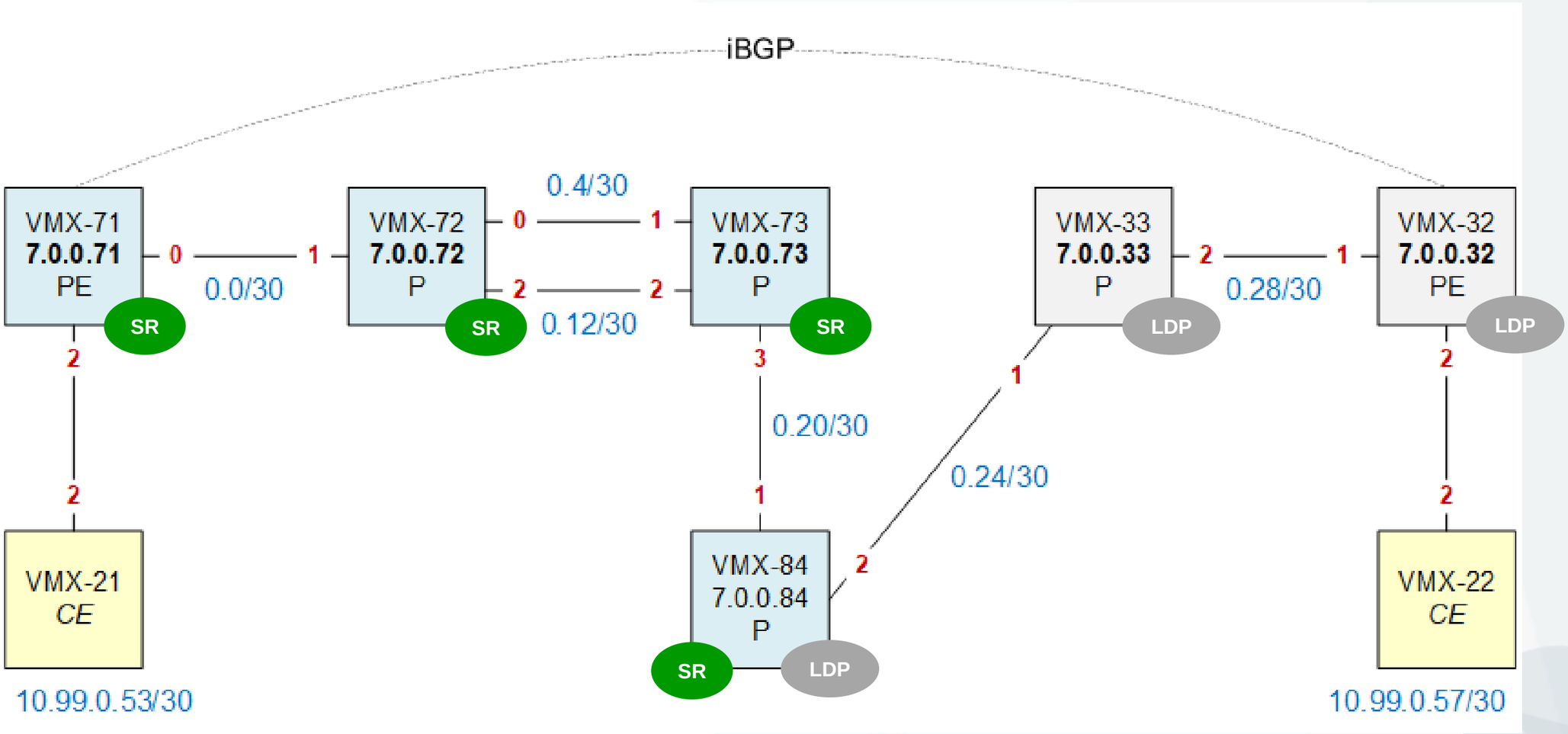
- 100% покрытие (для каждого есть запасной маршрут) …
- … но без LDP и автоматических tLDP сессий

```
warrior@vmx-71# run show route protocol isis 7.0.0.0/24 table inet.3
7.0.0.72/32        *[L-ISIS/14] 00:27:08, metric 5
                    > to 10.99.0.2 via ge-0/0/0.0
                      to 10.99.0.21 via ge-0/0/1.0, Push 800072, Push 800074(top)   // во всех случаях PQ-node это vmx-74
7.0.0.73/32        *[L-ISIS/14] 00:27:08, metric 10                                  // (в данной топологии) (для vmx-71)
                    > to 10.99.0.2 via ge-0/0/0.0, Push 800073
                      to 10.99.0.21 via ge-0/0/1.0, Push 800073, Push 800074(top)
7.0.0.74/32        *[L-ISIS/14] 00:41:05, metric 15
                      to 10.99.0.2 via ge-0/0/0.0, Push 800074
                    > to 10.99.0.21 via ge-0/0/1.0, Push 800074
7.0.0.75/32        *[L-ISIS/14] 00:27:08, metric 10
                    > to 10.99.0.21 via ge-0/0/1.0, Push 800075
                      to 10.99.0.2 via ge-0/0/0.0, Push 800075, Push 800074(top)
7.0.0.76/32        *[L-ISIS/14] 00:27:08, metric 5
                    > to 10.99.0.21 via ge-0/0/1.0
                      to 10.99.0.2 via ge-0/0/0.0, Push 800076, Push 800074(top)
```

Пример: сопряжение SR и LDP

Виртуальная топология

- Как связать (новый) SR домен и (старый) LDP домен?



Связность

```
warrior@vmx-71# run show bgp summary
Peer                     AS      InPkt     OutPkt    OutQ   Flaps Last Up/Dwn State|#Active/Received/Accepted/Damped...
7.0.0.32                100         22         19       0       2        7:47 1/1/1/0              0/0/0/0

warrior@vmx-71# run show route 7.0.0.32                                    // нет транспортной метки, чтобы достичь VMX-32
inet.0: 23 destinations, 23 routes (23 active, 0 holddown, 0 hidden)
7.0.0.32/32        *[IS-IS/18] 00:11:22, metric 50
                    > to 10.99.0.2 via ge-0/0/0.0

warrior@vmx-32# run show route 7.0.0.71                                    // нет транспортной метки, чтобы достичь VMX-71
inet.0: 21 destinations, 21 routes (21 active, 0 holddown, 0 hidden)
7.0.0.71/32        *[IS-IS/18] 00:43:04, metric 50
                    > to 10.99.0.29 via ge-0/0/1.0

jcluser@vmx-21# run traceroute 10.99.0.57                                  // связности нет
traceroute to 10.99.0.57 (10.99.0.57), 30 hops max, 52 byte packets
 1  10.99.0.54 (10.99.0.54)  1.491 ms  1.211 ms  0.964 ms
 2  * * *
 3  * *^C
```

Форвардинг из LDP в SR (1)

- Маршрутизаторы LDP домена видят только друг друга
- Где взять метки, чтобы достичь маршрутизаторы в SR домене?

```
jcluser@vmx-84# run show ldp database
Input label database, 7.0.0.84:0--7.0.0.33:0
Labels received: 3
  Label     Prefix
 299936      7.0.0.32/32
      3      7.0.0.33/32
 299952      7.0.0.84/32

Output label database, 7.0.0.84:0--7.0.0.33:0
Labels advertised: 3
  Label     Prefix
 300032      7.0.0.32/32
 300016      7.0.0.33/32
      3      7.0.0.84/32
```

```
jcluser@vmx-32# run show ldp database
Input label database, 7.0.0.32:0--7.0.0.33:0
Labels received: 3
  Label     Prefix
 299936      7.0.0.32/32
      3      7.0.0.33/32
 299952      7.0.0.84/32

Output label database, 7.0.0.32:0--7.0.0.33:0
Labels advertised: 3
  Label     Prefix
      3      7.0.0.32/32
 300096      7.0.0.33/32
 300112      7.0.0.84/32
```

Маппинг клиент

18.2R1

- VMX-84 назначает LDP метки для маршрутизаторов SR домена
- Операция в FIB: поменять LDP метку (*300080*) на SR метку (Node-SID, *800071*)

```
warrior@vmx-84# show | compare
[edit protocols ldp]
+    sr-mapping-client;              // включаем маппинг-клиент

warrior@vmx-84# run show route table mpls.0 label 300080

300080             *[LDP/9] 00:01:56, metric 1
                    > to 10.99.0.21 via ge-0/0/1.0, Swap 800071
```

```
jcluser@vmx-84# run show ldp database
Input label database, 7.0.0.84:0--7.0.0.33:0
Label      Prefix
 299936       7.0.0.32/32
      3       7.0.0.33/32
 300016       7.0.0.71/32
 300032       7.0.0.72/32
 300048       7.0.0.73/32
 299952       7.0.0.84/32

Output label database, 7.0.0.84:0--7.0.0.33:0
Label      Prefix
 300032       7.0.0.32/32
 300016       7.0.0.33/32
 300080       7.0.0.71/32
 300096       7.0.0.72/32
 300112       7.0.0.73/32
      3       7.0.0.84/32
```

Форвардинг из LDP в SR (2)

- теперь у VMX-32 есть MPLS метка, чтобы достичь VMX-71
- для VMX-32 это “обычная” LDP метка

```
jcluser@vmx-32# run show route 7.0.0.71

inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

7.0.0.71/32        *[IS-IS/18] 01:42:49, metric 50
                    > to 10.99.0.29 via ge-0/0/1.0

inet.3: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

7.0.0.71/32        *[LDP/9] 00:04:27, metric 50
                    > to 10.99.0.29 via ge-0/0/1.0, Push 300016
```

```
jcluser@vmx-32# run show ldp database
Input label database, 7.0.0.32:0--7.0.0.33:0
Label      Prefix
 299936       7.0.0.32/32
      3       7.0.0.33/32
 300016       7.0.0.71/32
 300032       7.0.0.72/32
 300048       7.0.0.73/32
 299952       7.0.0.84/32

Output label database, 7.0.0.32:0--7.0.0.33:0
Label      Prefix
      3       7.0.0.32/32
 300096       7.0.0.33/32
 300176       7.0.0.71/32
 300192       7.0.0.72/32
 300208       7.0.0.73/32
 300112       7.0.0.84/32
```

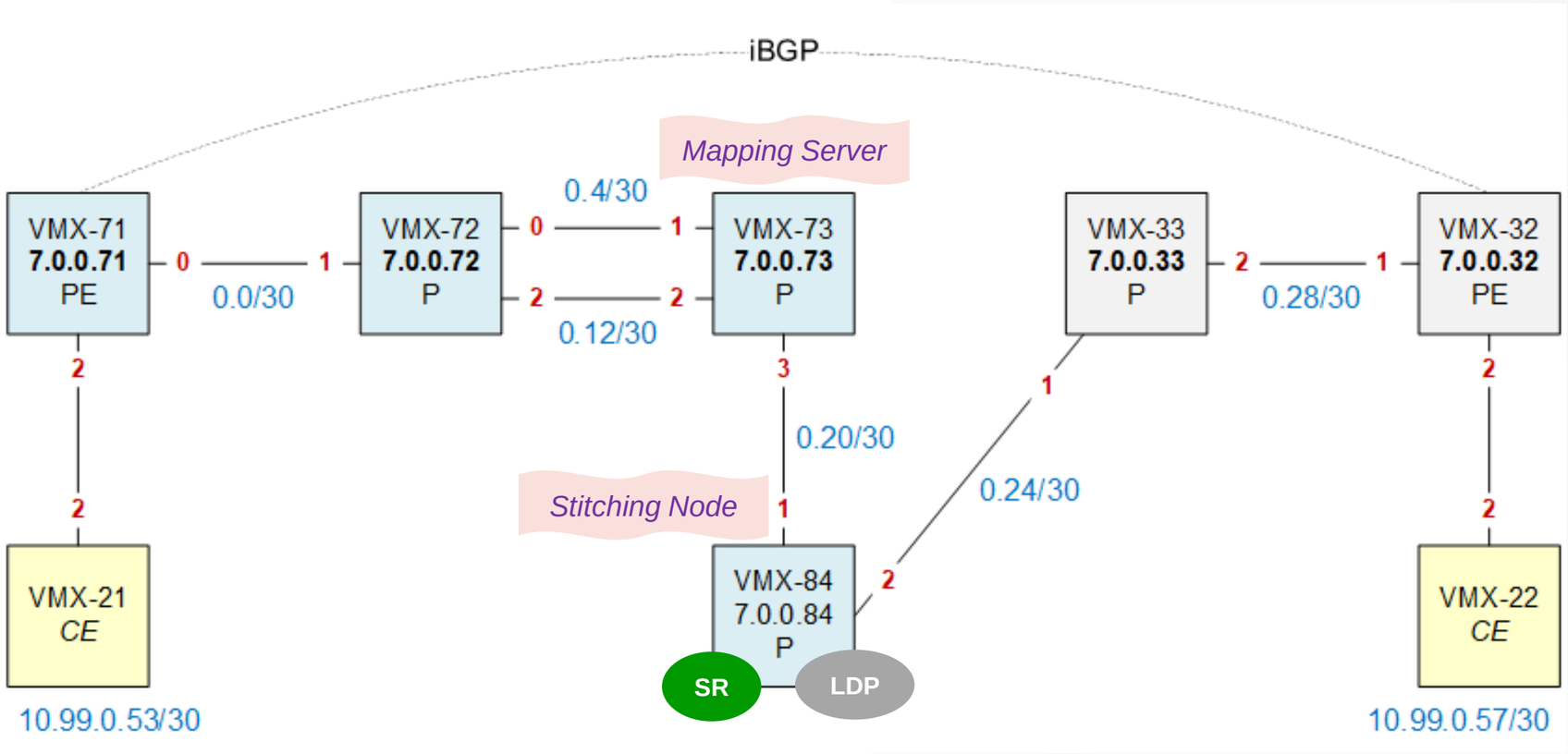
Форвардинг из SR в LDP

- В SR домене необходимы Node-SID метки для VMX-33, VMX-32
- VMX-33, VMX-32 не поддерживают SR. Кто сгенерирует метки?

```
jcluser@vmx-71# run show route table inet.3

inet.3: 3 destinations, 3 routes (3 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

7.0.0.72/32        *[L-ISIS/14] 00:20:28, metric 10
                    > to 10.99.0.2 via ge-0/0/0.0
7.0.0.73/32        *[L-ISIS/14] 00:20:17, metric 20
                    > to 10.99.0.2 via ge-0/0/0.0, Push 800073       // Node-SID для vmx-73
7.0.0.84/32        *[L-ISIS/14] 00:19:24, metric 30
                    > to 10.99.0.2 via ge-0/0/0.0, Push 800084       // Node-SID для vmx-84
```

Маппинг сервер

- Маппинг сервер генерирует SR метки для LDP домена
- Это может быть любой маршрутизатор SR домена, кроме stitching node VMX-84 (текущее ограничение в Junos)



Маппинг сервер – конфигурация и результат

?.?

- Теперь у vmx-73 есть метка, чтобы достичь vmx-32

```
warrior@vmx-73# show | compare
[edit]
+  routing-options {
+      source-packet-routing {
+          mapping-server-entry MPS1 {
+              prefix-segment 7.0.0.32/32 index 32;
+              prefix-segment 7.0.0.33/32 index 32;
+          }
+      }
+  }
[edit protocols isis source-packet-routing]
+    mapping-server MPS1;

jcluser@vmx-73# commit
commit complete
```

```
jcluser@vmx-71# run show route table inet.3

inet.3: 5 destinations, 5 routes (5 active, 0 holddown, 0
hidden)

7.0.0.32/32        *[L-ISIS/14] 00:01:34, metric 50
                    > to 10.99.0.2 via ge-0/0/0.0, Push 800032
7.0.0.33/32        *[L-ISIS/14] 00:01:34, metric 40
                    > to 10.99.0.2 via ge-0/0/0.0, Push 800033
7.0.0.72/32        *[L-ISIS/14] 00:27:34, metric 10
                    > to 10.99.0.2 via ge-0/0/0.0
7.0.0.73/32        *[L-ISIS/14] 00:27:23, metric 20
                    > to 10.99.0.2 via ge-0/0/0.0, Push 800073
7.0.0.84/32        *[L-ISIS/14] 00:26:30, metric 30
                    > to 10.99.0.2 via ge-0/0/0.0, Push 800084
```

Форвардинг из SR в LDP – последний штрих

```
jcluser@vmx-84# run show route table mpls.0 label 800032              // 800032 – SR метка для vmx-32

jcluser@vmx-84# show | compare
[edit protocols isis source-packet-routing]
+    ldp-stitching;                                                   // связываем SR метки и LDP метки

jcluser@vmx-84# commit
commit complete

jcluser@vmx-84# run show route table mpls.0 label 800032
mpls.0: 24 destinations, 24 routes (24 active, 0 holddown, 0 hidden)

800032             *[L-ISIS/14] 00:00:15, metric 20                  // 800032 – SR метка для vmx-32
                    > to 10.99.0.25 via ge-0/0/2.0, Swap 299936      // 299936 – LDP метка для vmx-32

jcluser@vmx-84# run show route 7.0.0.32 table inet.3
inet.3: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)

7.0.0.32/32        *[LDP/9] 00:56:27, metric 20
                    > to 10.99.0.25 via ge-0/0/2.0, Push 299936      // 299936 – LDP метка для vmx-32
```

Связность VMX-21 → VMX-22

```
jcluser@vmx-21# run ping 10.99.0.57
PING 10.99.0.57 (10.99.0.57): 56 data bytes
64 bytes from 10.99.0.57: icmp_seq=0 ttl=58 time=6.083 ms
^C
--- 10.99.0.57 ping statistics ---
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max/stddev = 6.083/6.083/6.083/0.000 ms

jcluser@vmx-21# run traceroute 10.99.0.57
traceroute to 10.99.0.57 (10.99.0.57), 30 hops max, 52 byte packets
 1  10.99.0.54 (10.99.0.54)  1.208 ms  0.990 ms  1.090 ms
 2  10.99.0.2 (10.99.0.2)  4.342 ms  4.249 ms  3.903 ms
     MPLS Label=800032 CoS=0 TTL=1 S=1
 3  10.99.0.14 (10.99.0.14)  4.157 ms  3.912 ms 10.99.0.6 (10.99.0.6)  3.696 ms
     MPLS Label=800032 CoS=0 TTL=1 S=1
 4  10.99.0.22 (10.99.0.22)  5.020 ms  4.694 ms  4.303 ms
     MPLS Label=800032 CoS=0 TTL=1 S=1
 5  * * *                                                    // vmx-33, Penultimate Hop, (см. слайд 65)
 6  10.99.0.30 (10.99.0.30)  48.872 ms  3.970 ms  3.971 ms
 7  10.99.0.57 (10.99.0.57)  5.385 ms  4.646 ms  4.547 ms
```

Связность VMX-22 → VMX-21

```
jcluser@vmx-22# run ping 10.99.0.53
PING 10.99.0.53 (10.99.0.53): 56 data bytes
64 bytes from 10.99.0.53: icmp_seq=0 ttl=58 time=5.060 ms
^C
--- 10.99.0.53 ping statistics ---
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max/stddev = 5.060/5.060/5.060/0.000 ms

jcluser@vmx-22# run traceroute 10.99.0.53
traceroute to 10.99.0.53 (10.99.0.53), 30 hops max, 52 byte packets
 1  10.99.0.58 (10.99.0.58)  98.957 ms  78.907 ms  26.958 ms
 2  10.99.0.29 (10.99.0.29)  4.582 ms  3.864 ms  4.299 ms
     MPLS Label=300016 CoS=0 TTL=1 S=1
 3  10.99.0.26 (10.99.0.26)  6.459 ms  4.433 ms  4.754 ms
     MPLS Label=300080 CoS=0 TTL=1 S=1
 4  10.99.0.21 (10.99.0.21)  4.552 ms  4.363 ms  4.048 ms
     MPLS Label=800071 CoS=0 TTL=1 S=1
 5  * * *                                                    // vmx-72, Penultimate Hop, (см. слайд 65)
 6  10.99.0.1 (10.99.0.1)  4.608 ms  4.166 ms  4.344 ms
 7  10.99.0.53 (10.99.0.53)  4.603 ms  5.165 ms  4.637 ms
```

Про звездочки в traceroute на PH роутере (1)

```
// дефолтный маршрут в inet.0 (для всех VMX в данном стенде)
// PHP пытается отправить “time exceeded” через inet.0, в сторону источника
routing-options {
    static {
        route 0.0.0.0/0 next-hop 100.123.0.1;
    }
}
```

```
jcluser@vmx-71# show | compare
[edit protocols isis source-packet-routing]
+    explicit-null;

jcluser@vmx-72# show | compare
[edit protocols isis source-packet-routing]
+    explicit-null;

jcluser@vmx-73# show | compare
[edit protocols isis source-packet-routing]
+    explicit-null;
```

```
jcluser@vmx-84# show | compare
[edit protocols isis source-packet-routing]
+    explicit-null;
[edit protocols ldp]
!    active: explicit-null;

jcluser@vmx-33# show | compare
[edit protocols ldp]
!    active: explicit-null;

jcluser@vmx-32# show | compare
[edit protocols ldp]
!    active: explicit-null;
```

Про звездочки в traceroute на PH роутере (2)

- Полноценный трейс с включенным explicit-null

```
jcluser@vmx-21# run traceroute 10.99.0.57
traceroute to 10.99.0.57 (10.99.0.57), 30 hops max …
 1  10.99.0.54 (10.99.0.54)  1.305 ms  1.095 ms  0.817 ms
 2  10.99.0.2 (10.99.0.2)  4.891 ms  5.150 ms  4.526 ms
     MPLS Label=800032 CoS=0 TTL=1 S=1
 3  10.99.0.6 (10.99.0.6)  4.176 ms 10.99.0.14 (10.99.0.14)
4.384 ms 10.99.0.6 (10.99.0.6)  4.097 ms
     MPLS Label=800032 CoS=0 TTL=1 S=1
 4  10.99.0.22 (10.99.0.22)  4.557 ms  4.638 ms  4.413 ms
     MPLS Label=800032 CoS=0 TTL=1 S=1
 5  10.99.0.25 (10.99.0.25)  4.300 ms  4.155 ms  4.400 ms
     MPLS Label=300064 CoS=0 TTL=1 S=1
 6  10.99.0.30 (10.99.0.30)  4.589 ms  4.039 ms  3.937 ms
 7  10.99.0.57 (10.99.0.57)  5.527 ms  4.720 ms  5.114 ms
```

```
jcluser@vmx-22# run traceroute 10.99.0.53
traceroute to 10.99.0.53 (10.99.0.53), 30 hops max …
 1  10.99.0.58 (10.99.0.58)  1.278 ms  0.963 ms  1.073 ms
 2  10.99.0.29 (10.99.0.29)  4.985 ms  4.440 ms  3.842 ms
     MPLS Label=300096 CoS=0 TTL=1 S=1
 3  10.99.0.26 (10.99.0.26)  4.778 ms  5.908 ms  39.831 ms
     MPLS Label=300240 CoS=0 TTL=1 S=1
 4  10.99.0.21 (10.99.0.21)  6.228 ms  4.050 ms  4.203 ms
     MPLS Label=800071 CoS=0 TTL=1 S=1
 5  10.99.0.13 (10.99.0.13)  4.384 ms  4.236 ms 10.99.0.5
(10.99.0.5)  3.896 ms
     MPLS Label=800071 CoS=0 TTL=1 S=1
 6  10.99.0.1 (10.99.0.1)  4.771 ms  4.070 ms  4.261 ms
 7  10.99.0.53 (10.99.0.53)  5.034 ms  4.725 ms  4.812 ms
```

Разное

Список (некоторых) RFC

- RFC 7855: Problem Statement and Requirements
- draft-ietf-spring-segment-routing
- draft-ietf-spring-segment-routing-MPLS
- draft-ietf-isis-segment-routing-extensions
- draft-ietf-ospf-segment-routing-extensions

Реализация в Junos

- Understanding Source Packet Routing in Networking (SPRING)

  https://www.juniper.net/documentation/en_US/junos/topics/concept/source-packet-routing.html

- CLI Explorer

  https://apps.juniper.net/cli-explorer/

Спасибо!

matvey@juniper.net