

Построение гигабитного доступа с 10G-uplink-портами

С начала 2020 года, по причине массового перехода граждан на удаленную работу, нагрузка на сети связи операторов значительно выросла. Так [по данным крупнейшего провайдера "Ростелеком"](#), рост потребления трафика жителями Свердловской области, по сравнению с предыдущим годом, составил примерно 25%. В тоже время, по данным [speedtest.net](#), средняя скорость фиксированного ШПД за прошедший год выросла на 35% (с 56 до 76 мегабит в секунду), что означает активный переход абонентов на более скоростные тарифы.

Соответственно, если ранее, по причине постепенного роста, вопрос модернизации сетей можно было откладывать в долгий ящик, то текущий колоссальный рост - почти на треть, делает данный вопрос крайне острым. Гигабитных скоростей на каналах от уровня агрегации до уровня доступа уже может не хватать, особенно при предоставлении клиентам высокоскоростных тарифов и использовании кольцевых топологий. А это значит - самое время рассмотреть возможность перехода на решение гигабитного доступа с uplink-портами 10 гигабит в секунду.

Компания SNR идет в ногу со временем и уже предлагает вам рассмотреть комплексное решение по построению гигабитного доступа с 10G uplink-портами и 10G-агрегации с uplink-портами 40G. Начнем с гигабитного доступа.

В августе 2020 года поступила на склад наша новая модель линейки коммутаторов SNR уровня доступа - [SNR-S2989G-24TX](#).



Портовая емкость новинки, в принципе, достаточно типовая, но, де-факто, является стандартом для построения "честного" гигабитного доступа, при котором полоса пропускания аплинк-порта не может быть полностью занята одним пользователем с гигабитным тарифом. Это 24 медных порта Gigabit Ethernet и 4 SFP+ аплинк порта, которые, к слову, могут работать как на скорости 1Gbps, так и на скорости 10Gbps.

Построен коммутатор на чипсете Marvell на основе софта серии [S2995G](#), уже отлаженного и доработанного под требования наших постоянных клиентов, и, кроме того, использует стандартные для всех коммутаторов SNR Cisco-like CLI и MIB. Поэтому для тех, кто уже знаком с коммутаторами SNR, и с серией S2995G в частности, интеграция в сеть новой модели не составит проблем. Для удобства управления модель также оснащена выделенными management и консольными портами, а также USB-портом для подключения внешних накопителей. Рекомендуемая розничная цена на SNR-S2989G-24TX на текущий момент составляет всего \$299.

Характеристики	SNR-S2989G-24TX
Интерфейсы	24xGE RJ45; 4x10GE SFP+
ASIC	Marvell Prestera DX
CPU	Dual-Core ARM
Матрица коммутации	128Gbps
Таблица MAC	16K
Пакетный буфер	1,5MB

Важно отметить, что, в отличие от старших моделей коммутаторов SNR, построенных на чипсетах Marvell, SNR-S2989G-24TX является L2-коммутатором уровня доступа, L3-функционал отсутствует. Но при этом необходимый операторский L2-функционал, любые сервисные модели и топологии поддерживаются в полной мере:

L2 4K VLAN, LACP 802.3ad, STP/RSTP/MSTP, ERPS G.8032, ERPS+CFM, VLAN translation, Port-based / Selective QinQ

L2 Multicast IGMP Snooping, MLD Snooping, MVR

Security L2-L4 port/VLAN-based ACL, User-defined ACL, Port Security, IP-MAC-Port Binding, Broadcast/Multicast/Unicast Storm Control

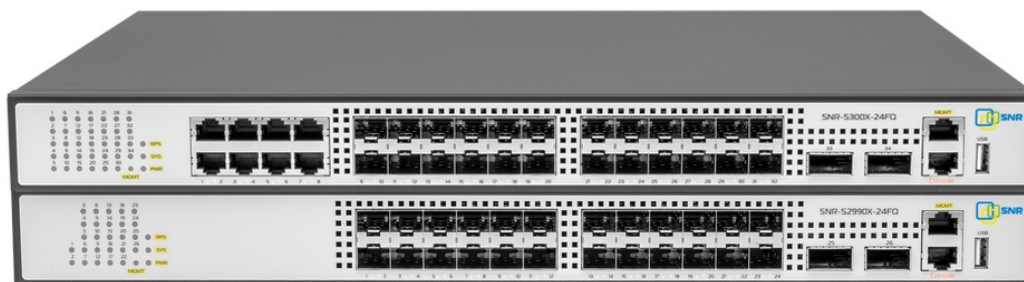
QoS 8 очередей на порт, SP/WDRR/SP+WDRR, маркировка/перемаркировка 802.1p, DSCP

* подробнее с характеристиками новинки можно ознакомиться в [техническом описании](#).

Кроме того, совсем недавно линейка пополнилась еще одной моделью - [SNR-S2989G-24TX-RPS](#) с RPS-портом 12В, позволяющим с небольшими затратами зарезервировать электропитание узла связи и обеспечить необходимое время автономной работы.

Перейдем к уровню агрегации.

В качестве коммутаторов агрегации мы предлагаем использовать коммутаторы [SNR-S2990X-24FQ](#) и [SNR-S300X-24FQ](#).



Так как обзор данных моделей [был совсем недавно представлен на нашем сайте](#), подробно останавливаться на них мы не будем. Лишь напомним, что SNR-S2990X-24FQ - L2-коммутатор, предназначенный для агрегации 10G-каналов операторов связи, коммутации в ЦОД, а SNR-S300X-24FQ - полноценный L3-коммутатор, ориентированный на ядро корпоративной сети или агрегацию/ядро оператора связи. Обе модели оснащены 10G downlink и 40G uplink-интерфейсами, а значит идеально подходят для решения нашей задачи.

Характеристики	SNR-S2990X-24FQ	SNR-S300X-24FQ
Интерфейсы	24x1/10GE SFP+, 2x40GbE QSFP+	8x10/100/1000BaseT, 24x1/10GE SFP+, 2x40GbE QSFP+
Матрица коммутации	640 Gbps	656 Gbps
Таблица MAC	32K	32K
ACL правил	1K	3K
Пакетный буфер	4 MB	4 MB
IPv4/IPv6 маршруты	-	16K
Таблица ARP	-	16K

Хотелось бы отметить, что рассмотренные нами коммутаторы доступа и агрегации в полной мере поддерживают любые сервисные модели и технологии подключения, например IPoE, PPPoE, а также могут применяться во всех стандартных сетевых топологиях. Поговорим о последних чуть более подробно.

Основные топологии, наиболее часто используемые операторами связи при построении СПД - всем известные звезда и кольцо. Как мы с вами уже обсуждали ранее в [нашем вебинаре](#), использование топологии кольцо при uplink-портах 10G на коммутаторах доступа более оправдано. Во-первых, для кольца из нескольких коммутаторов, пропускной способности в 20Gbps (два плеча кольца) в большинстве случаев должно быть достаточно. А во-вторых, очевидна экономия 10G-портов на агрегации, которые существенно дороже 1G-портов.

Убедимся в этом на практике с помощью расчетов, сравним стоимость подключения 5 коммутаторов доступа по топологии звезда и 5 коммутаторов доступа в одном кольце. В качестве коммутатора агрегации будем рассматривать SNR-S2990X-24FQ.

Рекомендуемая розничная цена коммутатора SNR-S2990X-24FQ - \$2500. Коммутатор оснащен 24 SFP+-портами, а значит стоимость одного порта SFP+ составит ~\$104. В качестве модулей SFP+ будем использовать [SNR-SFP+W37\(73\)-20](#) с ценой ~\$35.

Для построения топологии звезда нам потребуется 10 SFP+ модулей и 5 портов SFP+ на агрегации, а для построения кольца - 12 модулей, но всего лишь 2 порта на агрегации. Посмотрим, что мы имеем в итоге:

	Звезда		Кольцо	
	количество	стоимость	количество	стоимость
SFP+	10	\$350	12	\$420
Порт агрегации	5	\$520	2	\$208
Итого		\$870		\$628
Итого на узел		\$174		\$125

Как мы и предполагали, использование топологии кольцо при uplink-портах 10G на коммутаторах доступа оказалось более оправдано, экономия при 5 коммутаторах в кольце составила около 40%. Также, стоит отметить, что коммутатор SNR-S2990X-24FQ из нашего обзора имеет 2 порта QSFP+, которые, с помощью [MPO модуля](#) и [MPO патчкорда](#), можно разделить на 8 портов SFP+. Это сделает стоимость порта SFP+ ниже, а экономию чуть более существенной.

Наконец, перейдем к практической части нашего обзора.

Настройка ERPS и CFM на коммутаторах SNR-S2989G-24TX и SNR-S300X-24FQ

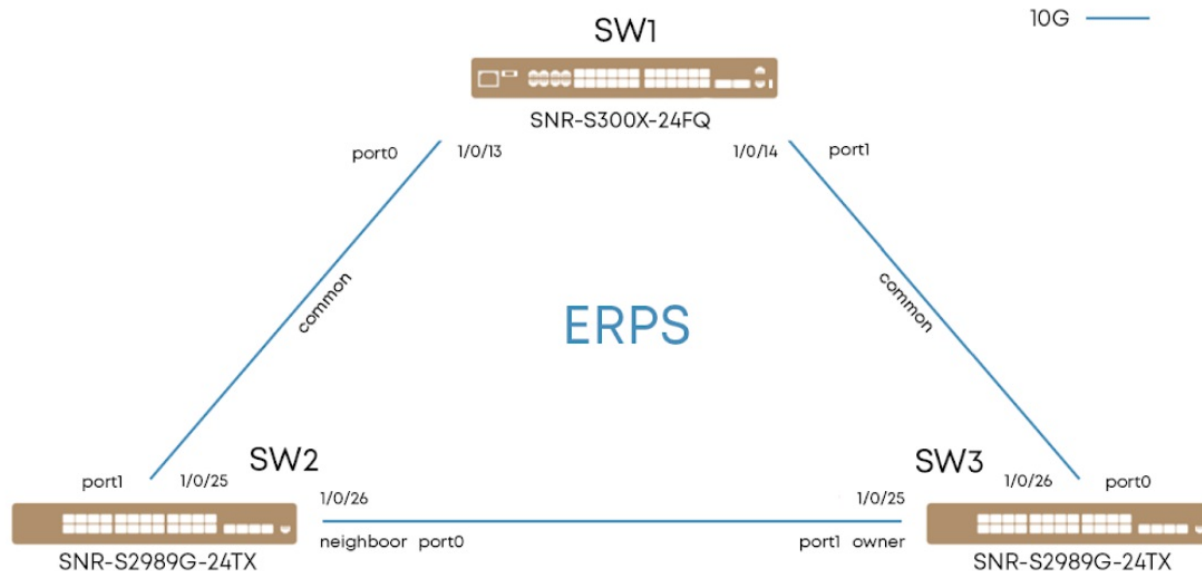
Учитывая уже доказанную нами выше выгоду кольцевой топологии, в практической части обзора мы решили рассмотреть конфигурацию и использование протокола ERPS на коммутаторах SNR-S300X-24FQ и SNR-S2989G-24TX, а также частный случай использования ERPS и CFM.

Для начала немного общей информации о ERPS.

[ERPS](#) (Ethernet Ring Protection Switching) - протокол, позволяющий осуществлять резервирование канала на втором уровне модели OSI путем физического создания петель и их логической блокировки.

Для каждого порта в кольце ERPS выбирается 1 из 3 возможных ролей: RPL Owner, RPL Neighbour или Common. Именно линк RPL Owner - RPL Neighbor при нормальных условиях выполняет блокировку петли и ее разблокировку в случае разрыва кольца. При этом, обнаружить разрыв кольца можно двумя способами. Первый - это стандартное падение физического интерфейса (линк падает - кольцо перестраивается). Второй - это обнаружение разрыва кольца с помощью CFM-протокола. Данный протокол позволяет настроить мониторинг между двумя коммутаторами на уровне L2 с помощью CCM-пакетов, которые отправляются с заданным таймаутом. Если в течение указанного таймаута не был получен CCM-пакет - включается режим защиты ERPS и кольцо перестраивается. Чаще всего CFM используется, если в кольце установлены подряд несколько коммутаторов, не поддерживающих ERPS, либо если один из линков предоставляется сторонним оператором СПД.

В первом примере рассмотрим простую схему из трех коммутаторов. К L3-коммутатору агрегации SNR-S300X-24FQ двумя 10G-линками подключены коммутаторы доступа SNR-S2989G-24TX, которые также соединены между собой оптическим линком 10G. Для того, чтобы равномерно разделить нагрузку по пропускной способности кольца на оба плеча, при отсутствии аварий линк между SNR-S2989G-24TX будет логически разорван с помощью ERPS.



Конфигурация коммутаторов в кольце в таком случае будет выглядеть следующим образом:

SW3 (Коммутатор с Owner RPL-портом всегда необходимо настраивать первым)

```

!
spanning-tree mst configuration #Создаем MST-инстанцию, которую будем защищать с помощью протокола ERPS.
instance 0 vlan 2-99;101-199;201-4094
instance 1 vlan 1;100;200 #Так как по умолчанию на коммутаторах SNR в trunk-портах добавлена VLAN1 как native, ее также необходимо добавить в эту инстанцию.
exit
!
port-scan-mode interrupt #Для ускорения сходимости кольца при разрыве/восстановлении линка изменим метод отслеживания состояния портов на ожидание
соответствующих прерываний (по умолчанию порты периодически опрашиваются).
!
vlan 1;100;200 #Создаем клиентскую VLAN (200) и VLAN, в которой будет передаваться служебный трафик ERPS (control VLAN 100).
!
erps-ring ring1 #Создаем кольцо и экземпляр ERPS.
erps-instance 1
guard-timer 10
wtr-timer 1 #Для удобства тестирования настраиваем значение WTR Timer в 1 минуту, а значение Guard Timer в 10 миллисекунд.
rpl port1 owner #Так как на данном коммутаторе имеется порт с ролью RPL Owner, то в режиме настройки инстанции необходимо указать эту информацию.
protected-instance 1 #Указываем MST-инстанцию, которую мы создали ранее.
control-vlan 100 #Указываем VLAN, в которой будут ходить служебные сообщения ERPS.
exit
!
Interface Ethernet1/0/25
switchport mode trunk
switchport trunk allowed vlan 100;200 #Настраиваем порты 1/0/25 и 1/0/26 на прохождение VLAN 100 и 200.
erps-ring ring1 port1 #Добавляем данные порты в текущую топологию ERPS.
!
Interface Ethernet1/0/26
switchport mode trunk
switchport trunk allowed vlan 100;200
erps-ring ring1 port0
!

```

SW2

Основные настройки идентичны SW3, за исключением того, что port0 является RPL Neighbor:

```

!
spanning-tree mst configuration
instance 0 vlan 2-99;101-199;201-4094
instance 1 vlan 1;100;200
exit
!
port-scan-mode interrupt
!
vlan 1;100;200
!
erps-ring ring1
erps-instance 1
guard-timer 10
wtr-timer 1
rpl port0 neighbour
protected-instance 1
control-vlan 100
exit
!
Interface Ethernet1/0/25
switchport mode trunk
switchport trunk allowed vlan 100;200
erps-ring ring1 port1
!
Interface Ethernet1/0/26
switchport mode trunk
switchport trunk allowed vlan 100;200
erps-ring ring1 port0
!

```

SW1

Настройки идентичны SW2 и SW3, но роль портов не настраивается (по умолчанию Common):

```
!
spanning-tree mst configuration
instance 0 vlan 2-99:101-199:201-4094
instance 1 vlan 1;100;200
exit
!
port-scan-mode interrupt
!
vlan 1;100;200
!
erps-ring ring1
erps-instance 1
guard-timer 10
wtr-timer 1
protected-instance 1
control-vlan 100
exit
!
Interface Ethernet1/0/13
switchport mode trunk
switchport trunk allowed vlan 100;200
erps-ring ring1 port0
!
Interface Ethernet1/0/14
switchport mode trunk
switchport trunk allowed vlan 100;200
erps-ring ring1 port1
!
```

Проверим настройки ERPS и убедимся, что кольцо перешло в режим IDLE, при котором отсутствуют аварии и логически заблокирован порт RPL Owner:

SW1

```
SW1# show erps ring ring1
```

```
R: RPL Owner
N: RPL Neighbour
C: Common Node
```

```
-----
Version: ITU-T G.8032v2
R-APS ring topology: major-ring
R-APS Virtual-Channel: with
Port0: Ethernet1/0/13
Failure-detect type: physical-link
Port1: Ethernet1/0/14
Failure-detect type: physical-link
```

Instance ID	Control Vlan	Protected Instance	WTR_Timer (min)	Guard_Timer (10ms)	Holdoff_Timer (second)	Port0	Port1
1	100	1	1	10	0	C	C

```
SW1#show erps status
```

```
ERPS ring: ring1 instance: 1 status:
Active: 1
Node State: IDLE
Time last topology change:Sep 15 08:48:00 2020
```

Port	Interface	Port-Status	Signal-Status	R-APS-Nodeid	BPR
Port0	Ethernet1/0/13	forwarding	Non-failed	f8-f0-82-79-ef-34	1
Port1	Ethernet1/0/14	forwarding	Non-failed	f8-f0-82-79-ef-34	1

SW2

```
SW2#show erps ring ring1
```

```
R: RPL Owner
N: RPL Neighbour
C: Common Node
```

```
-----
Version: ITU-T G.8032v2
R-APS ring topology: major-ring
R-APS Virtual-Channel: with
Port0: Ethernet1/0/26
Failure-detect type: physical-link
Port1: Ethernet1/0/25
Failure-detect type: physical-link
```

Instance ID	Control Vlan	Protected Instance	WTR_Timer (min)	Guard_Timer (10ms)	Holdoff_Timer (second)	Port0	Port1
1	100	1	1	10	0	N	C

```
SW2#show erps status
```

```
ERPS ring: ring1 instance: 1 status:
Active: 1
Node State: IDLE
Time last topology change:Sep 15 08:48:05 2020
```

Port	Interface	Port-Status	Signal-Status	R-APS-Nodeid	BPR
Port0	Ethernet1/0/26	forwarding	Non-failed	f8-f0-82-79-ef-34	1
Port1	Ethernet1/0/25	forwarding	Non-failed	f8-f0-82-79-ef-34	1

SW3

```
SW3#show erps ring ring1
```

```

R: RPL Owner
N: RPL Neighbour
C: Common Node
-----
Version: ITU-T G.8032v2
R-APS ring topology: major-ring
R-APS Virtual-Channel: with
Port0: Ethernet1/0/26
Failure-detect type: physical-link
Port1: Ethernet1/0/25
Failure-detect type: physical-link

```

Instance ID	Control Vlan	Protected Instance	WTR_Timer (min)	Guard_Timer (10ms)	Holdoff_Timer (second)	Port0	Port1
1	100	1	1	10	0	C	R

SW3#show erps status

```

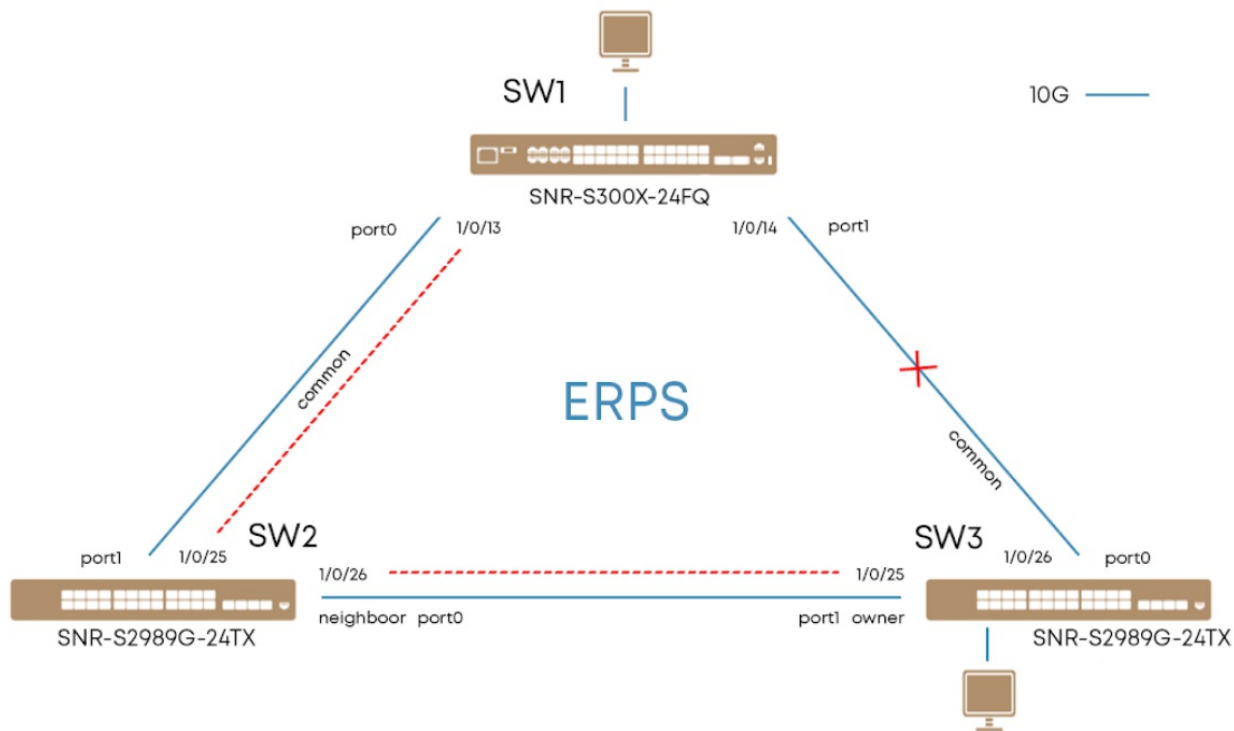
ERPS ring: ring1 instance: 1 status:
Active: 1
Node State: IDLE
Time last topology change: Sep 15 08:48:10 2020

```

Port	Interface	Port-Status	Signal-Status	R-APS-NodeId	BPR
Port0	Ethernet1/0/26	forwarding	Non-failed	00-00-00-00-00-00	0
Port1	Ethernet1/0/25	blocked	Non-failed	00-00-00-00-00-00	0

Как мы убедились, с настройками все в порядке, кольцо перешло в свое штатное состояние - IDLE.

Проведем несколько тестов, чтобы узнать среднее время сходимости топологии при аварийном падении одного из линков. Для этого запустим ICMP-тест с таймаутом в 1 миллисекунду между двумя устройствами, подключенными к SW1 и SW3. Соответственно, при падении линка SW1 - SW3 трафик будет ходить по маршруту SW1 - SW2 - SW3, а при восстановлении линка и истечении WTR Timer вернется на основной маршрут.



При падении линка, ERPS логирует события и переводит кольцо в состояние PENDING. Рассмотрим на примере SW3:

SW3 авария

```

SW3#show erps status
ERPS ring: ring1 instance: 1 status:
Active: 1
Node State: PROTECTION
Time last topology change: Sep 15 08:48:10 2020

```

Port	Interface	Port-Status	Signal-Status	R-APS-NodeId	BPR
Port0	Ethernet1/0/26	blocked	Failed	00-00-00-00-00-00	0
Port1	Ethernet1/0/25	forwarding	Non-failed	00-00-00-00-00-00	0

```

212 SW3 %Sep 15 08:57:34:758 2020 MODULE_PORT/5/:%LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet1/0/26, changed state to DOWN
211 SW3 %Sep 15 08:57:34:746 2020 MODULE_L2_ERPS/2/:Ethernet1/0/26 on ring ring1 instance 1 is set to block
210 SW3 %Sep 15 08:57:34:745 2020 MODULE_L2_ERPS/2/:Signal failure detected on f8-f0-82-79-ef-34 in ERPS ring:ring1, instance:1, port PORT0
209 SW3 %Sep 15 08:57:34:733 2020 MODULE_L2_ERPS/2/:Ethernet1/0/25 on ring ring1 instance 1 is set to forward

```

Как мы видим, при падении линка, SW3 разблокировал порт RPL Owner и заблокировал порт Common (1/0/26) по причине 'signal failure detected'. При восстановлении работоспособности линка происходит обратный процесс:

SW3 восстановление

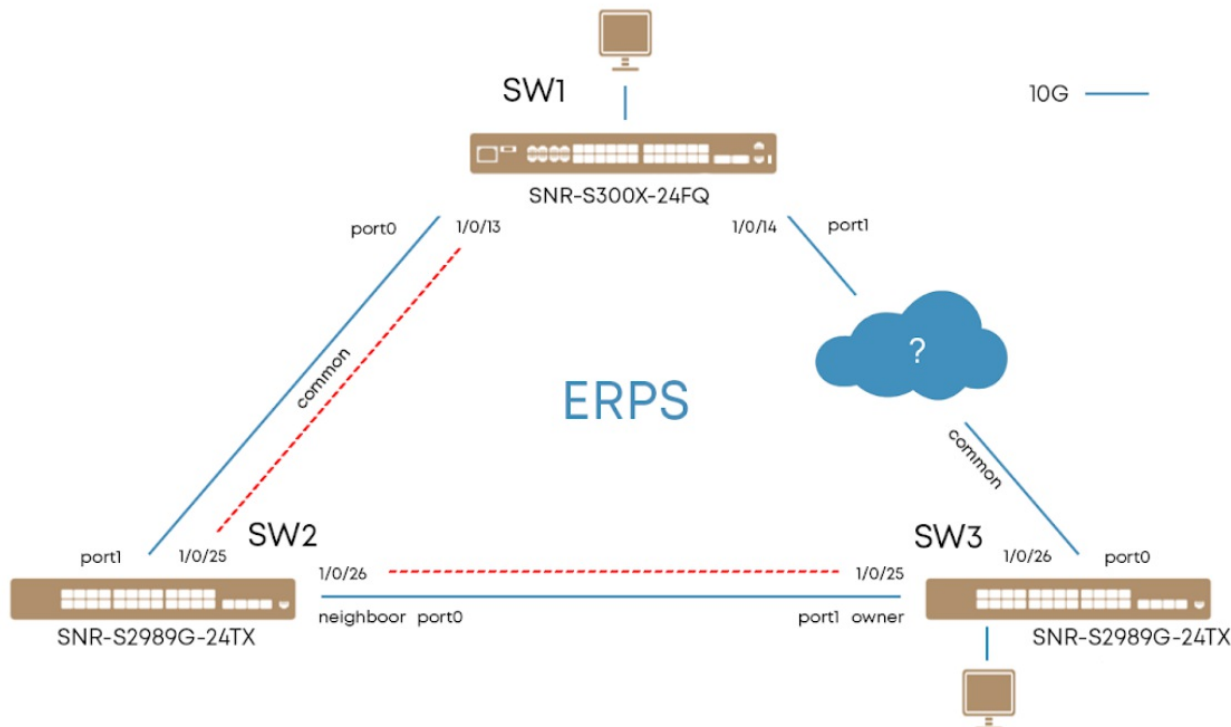
```

217 SW3 %Sep 15 08:38:59:437 2020 MODULE_L2_ERPS/2/:Ethernet1/0/25 on ring ring1 instance 1 is set to block
216 SW3 %Sep 15 08:37:59:194 2020 MODULE_L2_ERPS/2/:Ethernet1/0/26 on ring ring1 instance 1 is set to forward

```

215 SW3 %Sep 15 08:37:59:807 2020 MODULE_PORT/5:LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet1/0/26, changed state to UP
 214 SW3 %Sep 15 08:37:40:798 2020 MODULE_L2_ERPS/2:Signal failure cleared on f8-f0-82-79-ef-34 in ERPS ring:ring1, instance:1, port PORT0

Во втором примере немного усложним схему и добавим между SW1 и SW3 СПД стороннего оператора. В этом случае стандартного детектирования падения линка может не хватить - при проблемах на сети стороннего оператора физические интерфейсы на SW1 и SW3 будут активны, но связность при этом будет отсутствовать. Как мы уже знаем, решить эту проблему поможет протокол CFM, который настраивается на пограничных коммутаторах (SW1 и SW3) и с заданной периодичностью отправляет CCM-пакеты. В случае, если коммутатор не получает CCM-пакеты в течение заданного таймера - включается механизм защиты ERPS и топология перестраивается.



Рассмотрим настройку связки ERPS и CFM на SW1 и SW3:

SW1

```
!
ethernet cfm global #Включаем CFM глобально на коммутаторе.
!
ethernet cfm domain test level 2 #Создаем домен test уровня оператора.
service test pvlan 100 direction down #Создаем сервис с указанием Control VLAN и направления, в котором работает конечная точка.
mep mepid 100;200 #Указываем конечные точки данного сервиса.
continuity-check enable #Включаем отправку пакетов CCM.
continuity-check interval 3 #Настраиваем интервал отправки CCM-пакетов. Для уменьшения времени сходимости выберем 10pps (3).
continuity-check receive mep 100 #Указываем mepid получаемых CCM-пакетов, то есть mepid второго коммутатора.
exit
!
erps-ring ring1
port1 failure-detect physical-link-or-cc domain test service test mep 200 mep 100 #Включаем возможность обнаружения падения канала как с помощью падения физического порта, так и с помощью CFM.
!
Interface Ethernet1/0/14
ethernet cfm mep 200 domain test service test #Настраиваем порт 1/0/14 как конечную точку нашего CFM-сервиса.
!
```

SW3

Настройки на SW3 будут идентичны SW1 за исключением mepid:

```
!
ethernet cfm global
!
ethernet cfm domain test level 2
service test pvlan 100 direction down
mep mepid 100;200
continuity-check enable
continuity-check interval 3
continuity-check receive mep 200
exit
!
erps-ring ring1
port0 failure-detect physical-link-or-cc domain test service test mep 100 mep 200
!
Interface Ethernet1/0/26
ethernet cfm mep 100 domain test service test
!
```

Аналогично первому примеру убедимся, что с настройками все в порядке и проведем повторные тесты для выяснения среднего времени сходимости связки ERPS+CFM. В этом тесте в качестве СПД оператора мы используем два коммутатора (соответственно, без ERPS), между которыми и разрываем физический линк, имитируя аварию на сети.

При падении линка ERPS все также логирует события и переводит кольцо в состояние PENDING, но кроме этого, логируются и события CFM. Рассмотрим на примере SW3:

SW3 авария

4 SW3 %Sep 15 10:26:52:095 2020 MODULE_L2_CFM/4:/CFM:A CFM_ALARM_RMEP_CCM of Interface Ethernet1/0/26 is detected.
3 SW3 %Sep 15 10:26:49:298 2020 MODULE_L2_ERPS/2:/Ethernet1/0/26 on ring ring1 instance 1 is set to block
2 SW3 %Sep 15 10:26:49:298 2020 MODULE_L2_ERPS/2:/Signal failure detected on f8-f0-82-79-ef-34 in ERPS ring:ring1, instance:1, port PORT0
1 SW3 %Sep 15 10:26:49:297 2020 MODULE_L2_ERPS/2:/Ethernet1/0/25 on ring ring1 instance 1 is set to forward

Как мы видим, процесс практически идентичен, за исключением того, что вместо падения физического интерфейса о проблемах с линком нам сообщает CFM ALARM. При восстановлении работоспособности линка CFM сообщает, что проблема исчезла (сообщение появляется с настраиваемой задержкой):

SW3 восстановление

8 SW3 %Sep 15 10:28:02:259 2020 MODULE_L2_CFM/5:/CFM:The CC alarmed defects of Interface Ethernet1/0/26 disappear.
7 SW3 %Sep 15 10:27:52:934 2020 MODULE_L2_ERPS/2:/Ethernet1/0/25 on ring ring1 instance 1 is set to block
6 SW3 %Sep 15 10:26:57:129 2020 MODULE_L2_ERPS/2:/Ethernet1/0/26 on ring ring1 instance 1 is set to forward
5 SW3 %Sep 15 10:26:52:159 2020 MODULE_L2_ERPS/2:/Signal failure cleared on f8-f0-82-79-ef-34 in ERPS ring:ring1, instance:1, port PORT0

Подведем итоги.

После пяти проведенных тестов в стандартной топологии (первый пример) мы получили среднее время перестроения топологии при аварии - до 5 ms, при этом во время восстановления линка, влияния на сервис не удалось зафиксировать вообще. При использовании же связки ERPS + CFM среднее время перестроения топологии при аварии составило до 20 ms, при восстановлении линка - до 3 ms. Увеличение задержки при использовании CFM связано с периодичностью отправки CCM-сообщений. Поэтому мы рекомендуем использовать при настройке наименьший interval - 3, при котором в одну секунду коммутатором будет отправляться 10 CCM-сообщений.

Как мы видим, в обоих случаях время восстановления сервиса минимально. Да, конечно, тесты проводились, можно сказать, в практически идеальных условиях и минимальной нагрузке на Control Plane, тем не менее данные несравнимы, например, с протоколом STP, при котором время восстановления сервиса при перестроении топологии значительно выше.

Напоминаем, что по всем вопросам, в том числе предоставления скидок, вы можете обратиться [к вашему менеджеру](#).

Не забывайте подписываться на [наш Telegram-канал](#), в котором мы активно публикуем новости и анонсы, связанные с коммутаторами SNR!